

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *National Instruments PXIe-6592* at our website: *Click HERE*

NI High-Speed Serial Instruments User Manual

PXIe-6591R
PXIe-6592R
PXIe-7902

June 2017
374574F-01

NATIONAL INSTRUMENTS

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Worldwide Technical Support and Product Information

ni.com

Worldwide Offices

Visit ni.com/niglobal to access the branch office websites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

National Instruments Corporate Headquarters

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

For further support information, refer to the *NI Services* appendix. To comment on National Instruments documentation, refer to the National Instruments website at ni.com/info and enter the Info Code feedback.

© 2014-2017 National Instruments. All rights reserved.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Legal Information

Limited Warranty

This document is provided ‘as is’ and is subject to being changed, without notice, in future editions. For the latest version, refer to `ni.com/manuals`. NI reviews this document carefully for technical accuracy; however, NI MAKES NO EXPRESS OR IMPLIED WARRANTIES AS TO THE ACCURACY OF THE INFORMATION CONTAINED HEREIN AND SHALL NOT BE LIABLE FOR ANY ERRORS.

NI warrants that its hardware products will be free of defects in materials and workmanship that cause the product to fail to substantially conform to the applicable NI published specifications for one (1) year from the date of invoice.

For a period of ninety (90) days from the date of invoice, NI warrants that (i) its software products will perform substantially in accordance with the applicable documentation provided with the software and (ii) the software media will be free from defects in materials and workmanship.

If NI receives notice of a defect or non-conformance during the applicable warranty period, NI will, in its discretion: (i) repair or replace the affected product, or (ii) refund the fees paid for the affected product. Repaired or replaced Hardware will be warranted for the remainder of the original warranty period or ninety (90) days, whichever is longer. If NI elects to repair or replace the product, NI may use new or refurbished parts or products that are equivalent to new in performance and reliability and are at least functionally equivalent to the original part or product.

You must obtain an RMA number from NI before returning any product to NI. NI reserves the right to charge a fee for examining and testing Hardware not covered by the Limited Warranty.

This Limited Warranty does not apply if the defect of the product resulted from improper or inadequate maintenance, installation, repair, or calibration (performed by a party other than NI); unauthorized modification; improper environment; use of an improper hardware or software key; improper use or operation outside of the specification for the product; improper voltages; accident, abuse, or neglect; or a hazard such as lightning, flood, or other act of nature.

THE REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND THE CUSTOMER’S SOLE REMEDIES, AND SHALL APPLY EVEN IF SUCH REMEDIES FAIL OF THEIR ESSENTIAL PURPOSE.

EXCEPT AS EXPRESSLY SET FORTH HEREIN, PRODUCTS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND AND NI DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, WITH RESPECT TO THE PRODUCTS, INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, AND ANY WARRANTIES THAT MAY ARISE FROM USAGE OF TRADE OR COURSE OF DEALING. NI DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE OF OR THE RESULTS OF THE USE OF THE PRODUCTS IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. NI DOES NOT WARRANT THAT THE OPERATION OF THE PRODUCTS WILL BE UNINTERRUPTED OR ERROR FREE.

In the event that you and NI have a separate signed written agreement with warranty terms covering the products, then the warranty terms in the separate agreement shall control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

National Instruments respects the intellectual property of others, and we ask our users to do the same. NI software is protected by copyright and other intellectual property laws. Where NI software may be used to reproduce software or other materials belonging to others, you may use NI software only to reproduce materials that you may reproduce in accordance with the terms of any applicable license or other legal restriction.

End-User License Agreements and Third-Party Legal Notices

You can find end-user license agreements (EULAs) and third-party legal notices in the following locations:

- Notices are located in the `<National Instruments>\_Legal Information` and `<National Instruments>` directories.
- EULAs are located in the `<National Instruments>\Shared\MDF\Legal\license` directory.
- Review `<National Instruments>\_Legal Information.txt` for information on including legal information in installers built with NI products.

U.S. Government Restricted Rights

If you are an agency, department, or other entity of the United States Government (“Government”), the use, duplication, reproduction, release, modification, disclosure or transfer of the technical data included in this manual is governed by the Restricted Rights provisions under Federal Acquisition Regulation 52.227-14 for civilian agencies and Defense Federal Acquisition Regulation Supplement Section 252.227-7014 and 252.227-7015 for military agencies.

Trademarks

Refer to the *NI Trademarks and Logo Guidelines* at `ni.com/trademarks` for more information on National Instruments trademarks.

ARM, Keil, and µVision are trademarks or registered of ARM Ltd or its subsidiaries.

LEGO, the LEGO logo, WEDO, and MINDSTORMS are trademarks of the LEGO Group.

TETRIX by Pitsco is a trademark of Pitsco, Inc.

FIELDBUS FOUNDATION™ and FOUNDATION™ are trademarks of the Fieldbus Foundation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

EtherCAT® is a registered trademark of and licensed by Beckhoff Automation GmbH.

CANopen® is a registered Community Trademark of CAN in Automation e.V.

DeviceNet™ and EtherNet/IP™ are trademarks of ODVA.

Go!, SensorDAQ, and Vernier are registered trademarks of Vernier Software & Technology. Vernier Software & Technology and `vernier.com` are trademarks or trade dress.

Xilinx is the registered trademark of Xilinx, Inc.

Taptite and Trilobular are registered trademarks of Research Engineering & Manufacturing Inc.

FireWire® is the registered trademark of Apple Inc.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Handle Graphics®, MATLAB®, Real-Time Workshop®, Simulink®, Stateflow®, and xPC TargetBox® are registered trademarks, and TargetBox™ and Target Language Compiler™ are trademarks of The MathWorks, Inc.

Tektronix®, Tek, and Tektronix, Enabling Technology are registered trademarks of Tektronix, Inc.

The Bluetooth® word mark is a registered trademark owned by the Bluetooth SIG, Inc.

The ExpressCard™ word mark and logos are owned by PCMCIA and any use of such marks by National Instruments is under license.

The mark LabWindows is used under a license from Microsoft Corporation. Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Other product and company names mentioned herein are trademarks or trade names of their respective companies.

Members of the National Instruments Alliance Partner Program are business entities independent from National Instruments and have no agency, partnership, or joint-venture relationship with National Instruments.

Patents

For patents covering National Instruments products/technology, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your media, or the *National Instruments Patent Notice* at `ni.com/patents`.

Export Compliance Information

Refer to the *Export Compliance Information* at `ni.com/legal/export-compliance` for the National Instruments global trade compliance policy and how to obtain relevant HTS codes, ECCNs, and other import/export data.

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

YOU ARE ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY AND RELIABILITY OF THE PRODUCTS WHENEVER THE PRODUCTS ARE INCORPORATED IN YOUR SYSTEM OR APPLICATION, INCLUDING THE APPROPRIATE DESIGN, PROCESS, AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

PRODUCTS ARE NOT DESIGNED, MANUFACTURED, OR TESTED FOR USE IN LIFE OR SAFETY CRITICAL SYSTEMS, HAZARDOUS ENVIRONMENTS OR ANY OTHER ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE, INCLUDING IN THE OPERATION OF NUCLEAR FACILITIES; AIRCRAFT NAVIGATION; AIR TRAFFIC CONTROL SYSTEMS; LIFE SAVING OR LIFE SUSTAINING SYSTEMS OR SUCH OTHER MEDICAL DEVICES; OR ANY OTHER APPLICATION IN WHICH THE FAILURE OF THE PRODUCT OR SERVICE COULD LEAD TO DEATH, PERSONAL INJURY, SEVERE PROPERTY DAMAGE OR ENVIRONMENTAL HARM (COLLECTIVELY, “HIGH-RISK USES”). FURTHER, PRUDENT STEPS MUST BE TAKEN TO PROTECT AGAINST FAILURES, INCLUDING PROVIDING BACK-UP AND SHUT-DOWN MECHANISMS. NI EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS OF THE PRODUCTS OR SERVICES FOR HIGH-RISK USES.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Compliance

Electromagnetic Compatibility Information

This hardware has been tested and found to comply with the applicable regulatory requirements and limits for electromagnetic compatibility (EMC) as indicated in the hardware's Declaration of Conformity (DoC)[1]. These requirements and limits are designed to provide reasonable protection against harmful interference when the hardware is operated in the intended electromagnetic environment. In special cases, for example when either highly sensitive or noisy hardware is being used in close proximity, additional mitigation measures may have to be employed to minimize the potential for electromagnetic interference.

While this hardware is compliant with the applicable regulatory EMC requirements, there is no guarantee that interference will not occur in a particular installation. To minimize the potential for the hardware to cause interference to radio and television reception or to experience unacceptable performance degradation, install and use this hardware in strict accordance with the instructions in the hardware documentation and the DoC[1].

If this hardware does cause interference with licensed radio communications services or other nearby electronics, which can be determined by turning the hardware off and on, you are encouraged to try to correct the interference by one or more of the following measures:

- Reorient the antenna of the receiver (the device suffering interference).
- Relocate the transmitter (the device generating interference) with respect to the receiver.
- Plug the transmitter into a different outlet so that the transmitter and the receiver are on different branch circuits.

Some hardware may require the use of a metal, shielded enclosure (windowless version) to meet the EMC requirements for special EMC environments such as, for marine use or in heavy industrial areas. Refer to the hardware's user documentation and the DoC[1] for product installation requirements.

When the hardware is connected to a test object or to test leads, the system may become more sensitive to disturbances or may cause interference in the local electromagnetic environment.

Operation of this hardware in a residential area is likely to cause harmful interference. Users are required to correct the interference at their own expense or cease operation of the hardware.

Changes or modifications not expressly approved by National Instruments could void the user's right to operate the hardware under the local regulatory rules.

[1] The Declaration of Conformity (DoC) contains important EMC compliance information and instructions for the user or installer. To obtain the DoC for this product, visit `ni.com/certification`, search by model number or product line, and click the appropriate link in the Certification column.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Contents

About This Manual

Related Documentation 11
Xilinx Documentation 14
Additional Resources 14

Chapter 1
Before You Begin

Development Requirements 1-1
Xilinx Licensing Information 1-2
Installation Instructions 1-2

Chapter 2
PXIe-6591R Hardware Architecture

PXIe-6591R Module Overview 2-2
Clocking Architecture 2-3
PXIe-6591R Clocking 2-3

Chapter 3
Connecting and Interfacing with the PXIe-6591R

Front Panel 3-1
Recommended Mating Cables and Connectors 3-2
Transceiver Lane and Quad Mapping 3-2
Signal Routing 3-3
Socketed CLIP Interface 3-3
PXIe-6591R Socketed CLIP 3-4

Chapter 4
PXIe-6592R Hardware Architecture

PXIe-6592R Module Overview 4-2
Clocking Architecture 4-3
PXIe-6592R Clocking 4-3

Chapter 5
Connecting and Interfacing with the PXIe-6592R

Front Panel 5-1
Recommended Mating Cables and Connectors 5-2
Transceiver Lane and Quad Mapping 5-2
Signal Routing 5-3
Socketed CLIP Interface 5-3
PXIe-6592R Socketed CLIP 5-4

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Chapter 6
PXIe-7902 Hardware Architecture

PXIe-7902 Module Overview 6-2
Clocking Architecture 6-3
PXIe-7902 Clocking 6-3

Chapter 7
Connecting and Interfacing with the PXIe-7902

Front Panel 7-1
Recommended Mating Cables and Connectors 7-2
Transceiver Lane and Quad Mapping 7-2
Signal Routing 7-3
Socketed CLIP Interface 7-3
PXIe-7902 Socketed CLIP 7-4

Chapter 8
Developing Applications for the High-Speed Serial Device

Development Flow 8-1
Socketed CLIP Development 8-1
Accessing the Xilinx Vivado Tools 8-2
Exporting to Vivado 8-3
Generating an IP Core from the Xilinx Vivado IP Catalog 8-5
Modifying Third-Party IP Core Logic 8-5
Building a Netlist from the IP Core 8-6
Writing a VHDL Wrapper Around the Protocol IP Core 8-8
Constraints and Hierarchy 8-9
Documenting Your IP 8-10
Improving Performance in Larger Designs through Enable Chain Removal 8-10
Developing with LabVIEW FPGA 8-11
Configuring the High-Speed Serial Device LabVIEW FPGA Targets 8-11
Using Existing VHDL IP inside CLIP or IPIN 8-15
Adding High-Speed Serial Device Target I/O 8-15
Using the NI Common Instrument Design Libraries 8-16
Using niInstr Instruction Framework 8-16
Using niInstr Streaming 8-18
Using niInstr CLIP Adapters 8-18
Using niInstr Data Trigger 8-18
Using niInstr Basic Elements 8-18
Using niInstr Eye Scan 8-18
Connecting AXI4-Lite and AXI4-Stream Interfaces to the Host 8-18
Connecting Signals to Enable Eye Scan 8-19
Compiling LabVIEW FPGA VIs 8-21

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

LabVIEW and System Integration 8-21
Download, Reset, and Run Side Effects in the LabVIEW FPGA Host Interface 8-21
DMA Streaming 8-22
Peer-to-Peer Streaming 8-23
Maximizing Peer-to-Peer Streaming Throughput 8-23
PXI Triggers 8-23
Configuring Trigger Pulses 8-23
Reserving PXI Triggers 8-24
Reserving Trigger Lines in MAX 8-24
Reserving Trigger Lines in the LabVIEW FPGA Host VI 8-24
Reserving Trigger Lines 8-24
Releasing Trigger Lines 8-25
Monitoring Power and Temperature 8-25
Soft Shutdown 8-26
Power/Thermal Protection and Shutdown 8-26
Debugging Clocks Using Frequency Counters 8-26
Debugging Link Connections Using Eye Scan 8-27
Rectangular Eye Scan 8-27
N Point Eye Scan 8-28
Eye Scan State Model 8-29

Appendix A
Troubleshooting

Appendix B
Xilinx Documentation References

Appendix C
NI Services

Glossary

Figures

Figure 2-1. PXIe-6591R System Architecture Elements 2-2
Figure 2-2. PXIe-6591R Clocking Diagram 2-4

Figure 3-1. PXIe-6591R Front Panel Connectors and Pinouts 3-1
Figure 3-2. PXIe-6591R Signal Routing 3-3
Figure 3-3. PXIe-6591R Socketed CLIP Diagram 3-4

Figure 4-1. PXIe-6592R System Architecture Elements 4-2
Figure 4-2. PXIe-6592R Clocking Diagram 4-4

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Figure 5-1. PXIe-6592R Front Panel Connectors and Pinout 5-1
Figure 5-2. PXIe-6592R Signal Routing 5-3
Figure 5-3. PXIe-6592R Socketed CLIP Diagram 5-4

Figure 6-1. PXIe-7902 System Architecture Elements 6-2
Figure 6-2. PXIe-7902 Clocking Diagram 6-4

Figure 7-1. PXIe-7902 Front Panel Connectors and Pinout 7-1
Figure 7-2. PXIe-7902 Signal Routing 7-3
Figure 7-3. PXIe-7902 Socketed CLIP Diagram 7-4

Figure 8-1. High-Speed Serial Development Process 8-1
Figure 8-2. EDK Environment Error Message 8-2
Figure 8-3. Top-Level CLIP VHDL and Shared Logic 8-4
Figure 8-4. Instruction Framework Overview 8-14
Figure 8-5. Create AXI4 Resources VI Block Diagram 8-17
Figure 8-6. Connecting CLIP Resources to the Instruction Framework 8-18
Figure 8-7. Create AXI4-Lite Resources.vi 8-18
Figure 8-8. Read Module Temperature 8-23
Figure 8-9. Read Module Power 8-24
Figure 8-10. Rectangular Eye Scan 8-25
Figure 8-11. N Point Eye Scan 8-26
Figure 8-12. Eye Scan State Model 8-27

Tables

Table 1. Documentation Locations and Descriptions xi
Table 2. Xilinx Documentation xiv

Table 1-1. Fundamentals Resources 1-1

Table 2-1. PXIe-6591R Key Features 2-2
Table 2-2. PXIe-6591R Reference Clocks 2-3

Table 3-1. PXIe-6591R Front Panel Connectors 3-2
Table 3-2. Transceiver Lane and Quad Mapping 3-2
Table 3-3. Clock Signal and Quad Mapping 3-3
Table 3-4. PXIe-6591R CLIP Signals 3-4

Table 4-1. PXIe-6592R Key Features 4-2
Table 4-2. NI 6591R Reference Clocks 4-3
Table 4-3. PXIe-6592R Reference Clocks 4-3
Table 4-4. NI 6591R Reference Clocks 4-4

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-1. PXIe-6592R Front Panel Connectors 5-2
Table 5-2. Transceiver Lane and Quad Mapping 5-2
Table 5-3. Clock Signal and Quad Mapping 5-3
Table 5-4. PXIe-6592R CLIP Signals 5-4

Table 6-1. PXIe-7902 Key Features 6-2
Table 6-2. PXIe-7902 Reference Clocks 6-3

Table 7-1. PXIe-7902 Front Panel Connectors 7-2
Table 7-2. Transceiver Lane and Quad Mapping 7-2
Table 7-3. Clock Signal and Quad Mapping 7-3
Table 7-4. PXIe-7902 Socketed CLIP Signals 7-4

Table 8-1. PXIe-6591R Clocking and Routing Dependencies 8-11
Table 8-2. NI 6592R Clocking and Routing Dependencies 8-12

Table B-1. Xilinx 7-Series FPGA Documentation B-1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

About This Manual

The *NI High-Speed Serial Instruments User Manual* describes how to develop applications for use with the PXIe-6591R, PXIe-6592R, and PXIe-7902 high-speed serial FPGA targets.

The PXIe-6591R, PXIe-6592R, and PXIe-7902 are designed for use in the following applications:

- Real-time data transmission, reception, and validation
- Functional test for semiconductor production
- Device interfacing
- High performance embedded systems

This manual provides detailed information about the electrical and mechanical requirements of component-level IP (CLIP) and LabVIEW FPGA design.

Chapters 2 and 3 contain information about the PXIe-6591R hardware architecture and functionality, respectively. Chapters 4 and 5 contain information about the PXIe-6592R hardware architecture and functionality, respectively. Chapters 6 and 7 contain information about the PXIe-7902 hardware architecture and functionality, respectively. Chapter 8 contains information about how to develop applications for all high-speed serial devices.

Related Documentation

The following documents contain information that you may find helpful as you read this manual.

Table 1. Documentation Locations and Descriptions

Document	Location	Description
NI PXIe-6591R Specifications	Available from the Start menu and at `ni.com/manuals`.	Contains specifications for your PXIe-6591R module.
NI PXIe-6592R Specifications	Available from the Start menu and at `ni.com/manuals`.	Contains specifications for your PXIe-6592R module.
PXIe-7902 Specifications	Available from the Start menu and at `ni.com/manuals`.	Contains specifications for your PXIe-7902 module.
PXIe-6591R Getting Started Guide	Available from the Start menu and at `ni.com/manuals`.	Contains installation instructions for your system.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 1. Documentation Locations and Descriptions (Continued)

Document	Location	Description
PXIe-6592R Getting Started Guide	Available from the Start menu and at `ni.com/manuals`.	Contains installation instructions for your system.
PXIe-7902 Getting Started Guide	Available from the Start menu and at `ni.com/manuals`.	Contains installation instructions for your system.
NI High-Speed Serial Instruments Help	Available from the Start menu and at `ni.com/manuals`.	Contains information about how to add FPGA I/O to your project, and how to configure your high-speed serial device with VIs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 1. Documentation Locations and Descriptions (Continued)

Document		Location	Description
LabVIEW FPGA documentation	*NI LabVIEW High-Performance FPGA Developer's Guide*	Available at `ni.com/tutorial`.	Summarizes the most effective techniques for optimizing throughput, latency, and FPGA resources when using the LabVIEW FPGA Module and the NI RIO hardware platform.
	FPGA Module Help	This document is a book within the *LabVIEW Help*. Access this document by navigating to **Start»All Programs»National Instruments» LabVIEW» LabVIEW Help**, or by searching for *FPGA Module Help* at `ni.com/manuals`. Browse to the **FPGA Module** book in the **Contents** tab for information about using the LabVIEW FPGA Module.	With the LabVIEW FPGA Module and LabVIEW, you can create VIs that run on National Instruments FPGA targets. The *Getting Started with the LabVIEW FPGA* book provides links to the top resources that you can use to get started with LabVIEW FPGA. The *Integrating Third-Party IP (FPGA Module)* book contains information about adding custom HDL code to your LabVIEW project.
	LabVIEW FPGA Module Release and Upgrade Notes	Available at `ni.com/manuals`. You can also view this document by selecting **Start»All Programs»National Instruments» LabVIEW» LabVIEW Manuals**.	Contains information about installing the LabVIEW FPGA Module, describes new features, and provides upgrade information**.**

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Xilinx Documentation

Xilinx FPGA documentation provides information required for the successful development of your high-speed serial device. The following table provides a list of specific Xilinx documentation resources.

All Xilinx documentation can be found at `www.xilinx.com`.

Table 2. Xilinx Documentation

Document	Document Part Number	Description
7 Series FPGAs Overview	DS180	Outlines the features and product selection of the Xilinx 7 series FPGAs: Artix-7, Kintex-7, and Virtex-7 devices.
Kintex-7 FPGAs Data Sheet: DC and AC Switching Characteristics	DS182	Contains the DC and AC switching characteristic specifications for the Kintex-7 FPGAs.
Vivado Design Suite: Release Notes, Installation, and Licensing	UG973	Provides an overview of the new release of the Vivado Design Suite, including information on new and changed features, installation requirements for the software, and licensing information.
High-Speed Serial I/O Made Simple: A Designer's Guide, with FPGA Applications	—	Recommended for users new to high-speed serial.
7 Series FPGAs GTX/GTH Transceivers User Guide	UG476	Technical reference describing the 7 series FPGAs GTX/GTH transceivers.
Vivado Design Suite User Guide: Using Constraints	UG903	Describes using Xilinx Design Constraints (XDC) in Vivado tools.

Additional Resources

The software-designed instruments webpage, located at `ni.com/software-designed-instruments`, contains product information, white papers, and videos to help you develop applications.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1 Before You Begin

This section contains information you need before developing applications using the high-speed serial devices.

Development Requirements

Successful system design with the high-speed serial devices may require knowledge in the following areas, depending on your application.

- High-speed serial fundamentals
- VHDL code design
- NI LabVIEW and LabVIEW FPGA programming

If you are unfamiliar with any of these concepts, refer to the following table for a list of resources for learning the fundamentals required for development with high-speed serial devices.

Table 1-1. Fundamentals Resources

Concept	Resources
High-speed serial fundamentals	*High-Speed Serial I/O Made Simple: A Designers' Guide, with FPGA Applications*, available at `xilinx.com`.
VHDL code design	Some VHDL training or experience is required before working with the high-speed serial devices. Do not attempt to develop Component-Level IP (CLIP) without VHDL knowledge. Refer to *An Introduction to NI High-Speed Serial Instruments* for information about programming options.
NI LabVIEW and LabVIEW FPGA	NI LabVIEW and LabVIEW FPGA training are available at `ni.com/training`. You can also refer to the *NI LabVIEW High-Performance FPGA Developer's Guide*, available at `ni.com/tutorial`.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Xilinx Licensing Information

Refer to the *Xilinx Documentation* section in *About This Manual* for a list of Xilinx documentation that contains important Xilinx licensing information.

Installation Instructions

Refer to the getting started guide for your device (refer to the *Related Documentation* section of this document) for instructions about how to install LabVIEW, LabVIEW FPGA, the NI LabVIEW Instrument Design Libraries for High Speed Serial Instruments software, and your high-speed serial device.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2

PXIe-6591R Hardware Architecture

The following chapter contains information about the PXIe-6591R clocking architecture.

The PXIe-6591R is a high-speed serial interfacing module. The PXIe-6591R hardware architecture allows you to fully customize your serial digital protocol application. The high-speed serial interface uses Xilinx GTX transceiver technology; you can reuse existing protocol IP that works with Xilinx GTX transceivers, or you can develop your own protocol IP. If you develop your own protocol IP, the IP must be developed for a Xilinx Kintex-7 GTX transceiver.

Note The PXIe-6591R hardware does not require calibration.

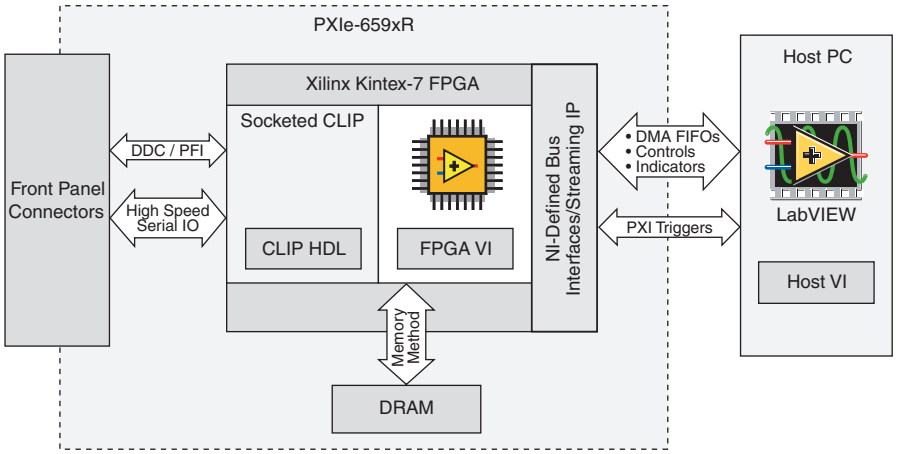
PXIe-6591R-based projects consist of the following key components:

- PXIe-6591R front panel connectors for data, clocking, and triggering external to the module
- Socketed CLIP for HDL IP and interface definition from the FPGA VI to the PXIe-6591R front panel
- Xilinx Kintex-7 FPGA
- Dynamic random access memory (DRAM)
- NI-defined bus interface from the FPGA to the host PC

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following figure illustrates the key components of the PXIe-6591R architecture.

Figure 2-1. PXIe-6591R System Architecture Elements



PXIe-6591R Module Overview

The PXIe-6591R module includes the following key features. Refer to the *PXIe-6591R Specifications* for more details.

Table 2-1. PXIe-6591R Key Features

Line rate	500 Mb/s to 8 Gb/s and 9.8 Gb/s to 12.5 Gb/s
Multi-gigabit transceiver lanes	8 (4 per port)
Front Panel Connectors	• Two Mini-SAS HD connectors (Port 0 and Port 1) for high-speed serial I/O • One SMA (CLK IN/OUT) connector for external clock routing • One VHDCI (Digital Data & Control) for 20 general-purpose input/output lines
FPGA	Kintex-7 410T FFG900 package
FPGA speed grade	-3 (XC7K410T-3FFG900)
DRAM	• 2 GB onboard DRAM • 166 MHz clock frequency • Bit-width: 512-bit
Backplane connection	Gen 2x8 PXI Express, PCIe 2.0 compliant

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Clocking Architecture

The PXIe-6591R module includes dedicated clocking hardware to provide a flexible clocking solution for generating the high-speed serial transceiver reference clocks (MGT_RefClk). Use the **Clocking and IO** properties page in the LabVIEW project to configure the clock settings for your module. Refer to the *Configuring the High-Speed Serial Device LabVIEW FPGA Targets* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for more detailed information about configuring clocking in your LabVIEW project.

PXIe-6591R Clocking

The PXIe-6591R clocking architecture includes the following MGT Reference Clocks:

- MGT_RefClk0
- MGT_RefClk1

MGT_RefClk0 and MGT_RefClk1 are separate clocks, but are derived though simple integer division of a common, higher frequency PLL clock. Refer to the following table for information about the clocks' supported frequency ranges and available sources.

Table 2-2. PXIe-6591R Reference Clocks

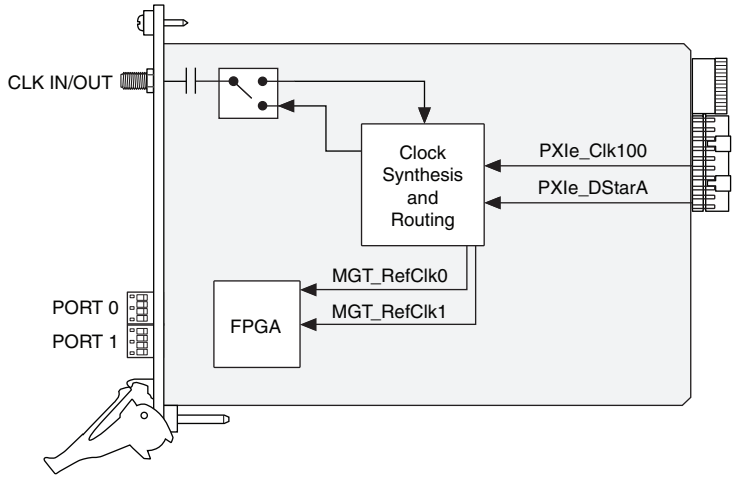
Clock Name	Frequency Range	Available Sources
MGT_RefClk0	60 MHz to 700 MHz	Backplane: PXIe_Clk100 and PXIe_DStarA Front panel: CLK IN/OUT
MGT_RefClk1		

Refer to the *Configuring the High-Speed Serial Device LabVIEW FPGA Targets* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for more information about how to configure Reference Clocks for your device.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following figure illustrates the clocking circuitry on the PXIe-6591R.

Figure 2-2. PXIe-6591R Clocking Diagram



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3

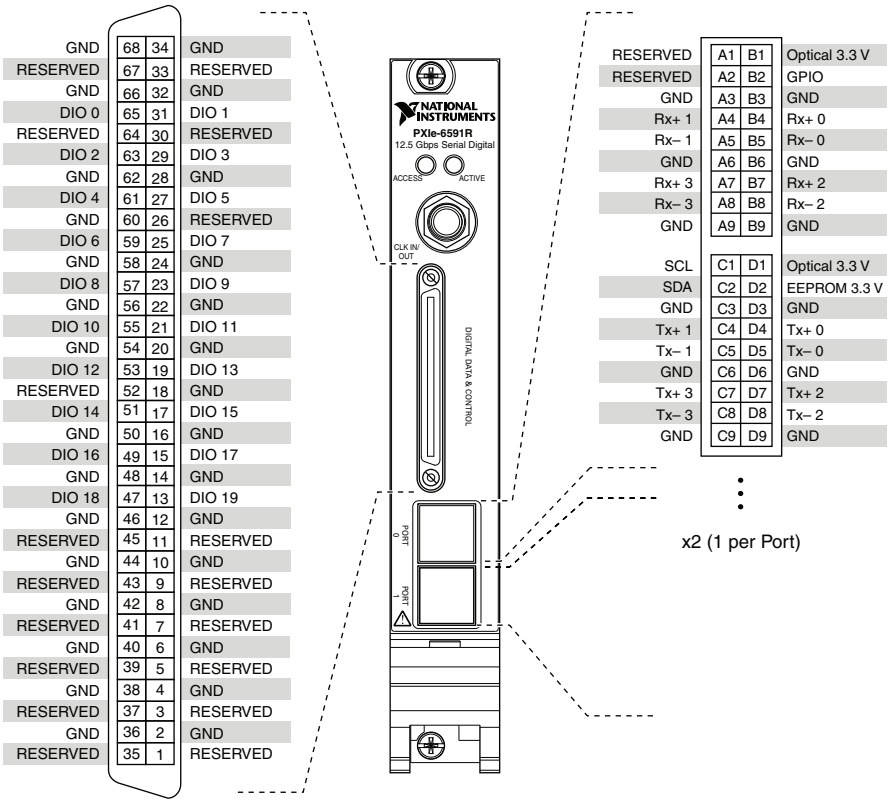
Connecting and Interfacing with the PXIe-6591R

This chapter contains information about the PXIe-6591R module and its functionality, including front panel diagrams, connectors, and pinouts.

Front Panel

The following figure shows the pinouts for the PXIe-6591R front panel connectors.

Figure 3-1. PXIe-6591R Front Panel Connectors and Pinouts



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Refer to the following table for a list of the PXIe-6591R front panel connectors and their descriptions.

Table 3-1. PXIe-6591R Front Panel Connectors

Connector	Type	Description
CLK IN/OUT	SMA	Reference Clock input and exported clock output
Digital Data & Control (DDC)	VHDCI	General purpose I/O
Port 0	Mini-SAS HD External	High-speed serial interfacing ports
Port 1		

Recommended Mating Cables and Connectors

Refer to the PXIe-6591R product listing page for a list of mating cables to use with your PXIe-6591R.

Transceiver Lane and Quad Mapping

If your application requires multiple lanes, refer to Table 3-2 and Table 3-3 for information about transceiver and RefClk selection when using the Xilinx tools to generate protocol IP.

Table 3-2. Transceiver Lane and Quad Mapping

Connector	Lane	Quad Location	Physical Resource
PORT 0	0	Quad 3 (Q3)	GTX_X0Y13
	1		GTX_X0Y15
	2		GTX_X0Y14
	3		GTX_X0Y12
PORT 1	0	Quad 2 (Q2)	GTX_X0Y10
	1		GTX_X0Y11
	2		GTX_X0Y9
	3		GTX_X0Y8

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-3. Clock Signal and Quad Mapping

Clock Signal	Quad Location	Physical Resource
MGT_RefClk0	Quad 3 (Q3)	REFCLK1_Q3
MGT_RefClk1	Quad 2 (Q2)	REFCLK0_Q2

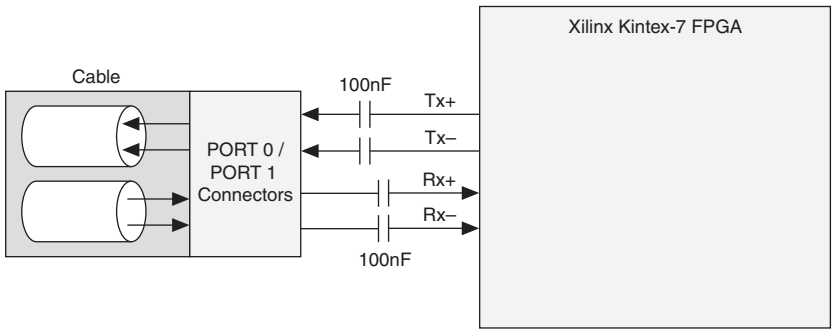
For more information about lane and channel bonding caveats, refer to the *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) at `xilinx.com`.

Note The MGT_RefClk quad location and physical resource identify the physical resource that clock signals use to enter the FPGA, but they still may act as a Reference Clock for adjacent quads. Refer to the *Reference Clock Selection and Distribution* section of *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) for more information about cases when using single or multiple Reference Clocks for single or multiple transceivers.

Signal Routing

The PXIe-6591R high-speed serial differential signals are routed directly from the Kintex-7 FPGA pins to the PORT 0 and PORT 1 connector pins using a 100 nF AC-coupling capacitor, as shown in the following figure.

Figure 3-2. PXIe-6591R Signal Routing



Socketed CLIP Interface

Socketed CLIP allows you to insert HDL IP into an FPGA target, enabling VHDL code to communicate directly with an FPGA VI. Socketed CLIP also allows the CLIP to communicate directly with circuitry external to the FPGA.

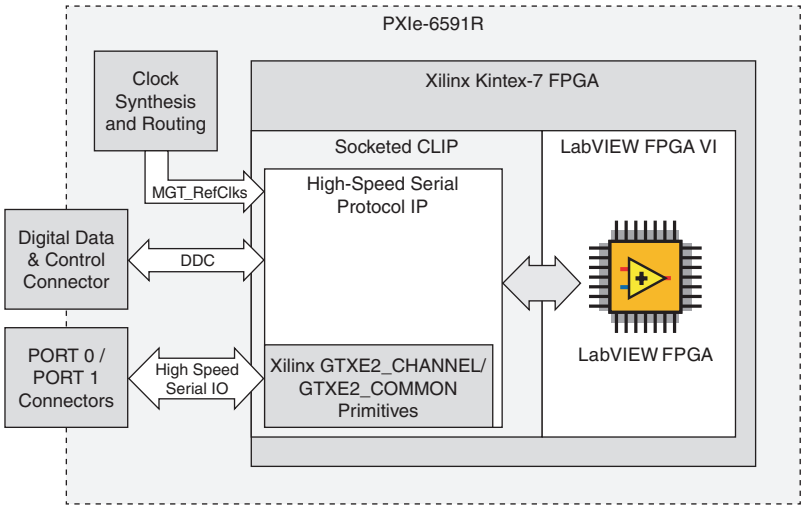
The following sections provide information about how to configure your device for use with socketed CLIP.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

PXIe-6591R Socketed CLIP

Refer to the following diagram for an overview of the PXIe-6591R socketed CLIP interface.

Figure 3-3. PXIe-6591R Socketed CLIP Diagram



Refer to the following table for a list of the PXIe-6591R socketed CLIP signals.

Table 3-4. PXIe-6591R CLIP Signals

Port	Direction	Clock Domain	Description
MGT_RefClk0_p	In (pad)	N/A	Differential input clock that you must connect to an IBUFDS_GTE2 input buffer primitive when this input clock is used in your design.
MGT_RefClk0_n	In (pad)	N/A	
MGT_RefClk1_p	In (pad)	N/A	
MGT_RefClk1_n	In (pad)	N/A	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
MGT_RefClks_ ExtPllLocked	In	Async	Indicates the state of the PLL within the clocking logic that provides the Reference Clock to the FPGA MGTs (**MGT_ RefClk*x*** signals). Use this signal with **MGT_ RefClks_Valid** to gate and/or reset the clocking signals into any CLIP that depends on the **MGT_RefClk*x*** signals.
MGT_RefClks_Valid	In	Async	Indicates if the selected clock input to the clocking logic is valid and the PLL within the clocking logic has locked. Use this signal to gate and/or reset the clocking signals into any CLIP that depends on the **MGT_RefClk*x*** signals. On the rising edge of **MGT_ RefClks_Valid**, you may need to reset or relock state machines and/or internal PLLs sensitive to **MGT_RefClk*x*** signals.
DebugClks(3:0)	Out	Clock	Debug ports to aid in debugging the clocking connections in the CLIP. These ports connect to frequency counters that can monitor the frequency of any clock that you connect to these ports. Refer to the *Debugging Clocks Using Frequency Counters* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for details about how to use these signals.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
ExportedUser ReferenceClk	Out	Clock	Reserved for future use.
LED_ActiveRed	Out	Async	The front panel Active indicator's red LED turns on when this signal is driven high. The CLIP's access to this LED may be temporarily overridden to show error conditions, temperature faults, and power faults. This signal is conditioned with the pulse stretcher to guarantee a minimum assertion time of 100 ms to comply with PXI guidelines and to facilitate visual perception. You can drive this signal asynchronously if you provide a 50 ns minimum assertion time. You can also drive this signal synchronously for a minimum 1 cycle of SocketClk40.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
LED_ActiveGreen	Out	Async	The front panel Active indicator's green LED turns on when this signal is driven high. The CLIP's access to this LED may be temporarily overridden to show error conditions, temperature faults, and power faults. This signal is conditioned with the pulse stretcher to guarantee a minimum assertion time of 100 ms to comply with PXI guidelines and to facilitate visual perception. You can drive this signal asynchronously if you provide a 50 ns minimum assertion time. You can also drive this signal synchronously for a minimum 1 cycle of SocketClk40.
SocketClk40	In	Clock	A 40 MHz clock that runs continuously regardless of connectivity. This signal is connected to the **40 MHz Onboard Clock** signal, which is the default top-level clock for the LabVIEW FPGA VI.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
sFrontEndConfiguration Done	In	SocketClk40	Asserts high and stays high when the power-on self-configuration (POSC) state machine is finished with configuration. After the **aResetSl** signal transitions from high to low, indicating that the CLIP logic should come out of reset, a POSC reconfiguration occurs unconditionally. The required clocking signals are not valid until after this signal asserts high.
sFrontEndConfiguration Prepare	In	SocketClk40	Reserved for future use. NI recommends assigning this signal to **sFrontEnd ConfigurationReady**.
sFrontEndConfiguration Ready	Out	SocketClk40	Reserved for future use. NI recommends assigning **sFrontEndConfiguration Prepare** to this signal.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
aResetSl	In	Async	This signal is not required. This signal is an asynchronous reset signal from the LabVIEW FPGA environment. If you create an input signal to your CLIP and assign it as **Reset** in the CLIP wizard, that signal is driven as an asynchronous reset signal. Reset all CLIP state machines and logic whenever this signal is logic high. This signal is driven high when you call the LabVIEW FPGA **Reset** invoke method. Call **Run** on the FPGA VI to deassert this signal. Do not use CLIP inputs from the LabVIEW FPGA VI in the CLIP until **aResetSl** is deasserted.
Port<0..1>_RX_p(3:0)	In (pad)	N/A	Dedicated MGT receive signals for Port <0..1>.
Port<0..1>_RX_n(3:0)	In (pad)	N/A	
Port<0..1>_TX_p(3:0)	Out (pad)	N/A	Dedicated MGT transmit signals for Port <0..1>.
Port<0..1>_TX_n(3:0)	Out (pad)	N/A	
Port<0..1>_SCL	In/Out	Async	Bidirectional serial clock signal for the two-wire communication interface on the Port <0..1> connector. Valid values: `0` and `Z` (open drain). This signal is also called MODDEF1. This signal has a 5 kΩ pull up to +3.3V.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
Port<0..1>_SDA	In/Out	Async	Bidirectional serial data signal for the two-wire communication interface on the Port <0..1> connector. Valid values: 0 and Z (open drain). This signal is also called MODDEF2. This signal has a 5 kΩ pull up to +3.3V.
Port<0..1>_GPIO_In	In	Async	Active low presence detect signal from pin B2 on the cable connector. You must tie **GPIO_OutEnable_n** to "1" in order to allow this functionality. This input is driven low by the high-speed connector while it is inserted into the module.
Port<0..1>_GPIO_Out	Out	Async	This signal is unused.
Port<0..1>_GPIO_ OutEnable_n	Out	Async	You must tie this signal to "1" to disable output and allow the B2 pin to function as a presence detect signal.
sPort<0..1>_ EnablePower	Out	SocketClk40	Enables or disables the power supply to the cable on Port <0..1>. This signal is active high.
sPort<0..1>_Power Good	In	SocketClk40	Indicates that the power supply to the cable for Port <0..1> is enabled. This signal may deassert if an over-power condition is detected.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 3-4. PXIe-6591R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
DDC_GPIO_In(19:0)	In	Async	These signals are GPIO inputs within the DDC VHDCI connector. These signals, along with **DDC_GPIO_Out(19:0)** and **DDC_GPIO_OutEnable_n(19:0)**, allow control and monitoring of the DIO(19:0) connections on the DDC_ VHDCI connector.
DDC_GPIO_Out(19:0)	Out	Async	These signals are GPIO outputs within the DDC VHDCI connector. These signals, along with **DDC_GPIO_In(19:0)** and **DDC_GPIO_OutEnable_n(19:0)**, allow control and monitoring of the DIO(19:0) connections on the DDC_ VHDCI connector.
DDC_GPIO_OutEnable_n(19:0)	Out	Async	These signals enable GPIO_Out within the DDC VHDCI connector. Drive these signals low to enable output. These signals, along with **DDC_GPIO_In(19:0)** and **DDC_GPIO_Out(19:0)**, allow control and monitoring of the DIO(19:0) connections on the DDC_VHDCI connector.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4

PXIe-6592R Hardware Architecture

The following chapter contains information about the PXIe-6592R clocking architecture.

The PXIe-6592R is a high-speed serial interfacing module. The PXIe-6592R hardware architecture allows you to fully customize your serial digital protocol application. The high-speed serial interface uses Xilinx GTX transceiver technology; you can reuse existing protocol IP that works with Xilinx GTX transceivers, or you can develop your own protocol IP. If you develop your own protocol IP, the IP must be developed for a Xilinx Kintex-7 GTX transceiver.



Note The PXIe-6592R hardware does not require calibration.

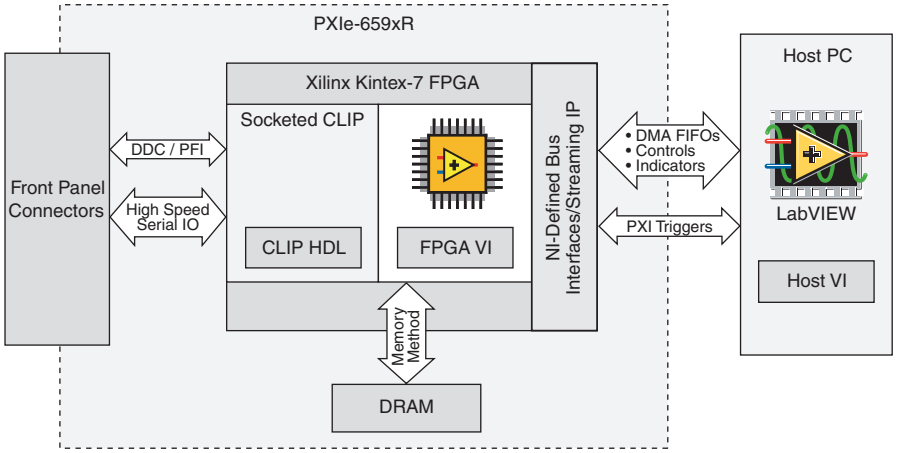
PXIe-6592R-based projects consist of the following key components:

- PXIe-6592R front panel connectors for data, clocking, and triggering external to the module
- Socketed CLIP for HDL IP and interface definition from the FPGA VI to the PXIe-6592R front panel
- Xilinx Kintex-7 FPGA
- Dynamic random access memory (DRAM)
- NI-defined bus interface from the FPGA to the host PC

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following figure illustrates the key components of the PXIe-6592R architecture.

Figure 4-1. PXIe-6592R System Architecture Elements



PXIe-6592R Module Overview

The PXIe-6592R modules include the following key features. Refer to the *PXIe-6592R Specifications* for more details.

Table 4-1. PXIe-6592R Key Features

Line rate	500 Mb/s to 8 Gb/s and 9.8 Gb/s to 10.3125 Gb/s serial
Multi-gigabit transceiver lanes	4 (1 per port)
Front Panel Connectors	• Four SFP+ connectors (Port 0, Port 1, Port 2, and Port 3) for high-speed serial • Four SMB connectors (PFI 0/CLK IN/OUT, PFI 1/CLK OUT, PFI 2/CLK OUT, and PFI 3/CLK OUT) for external triggering and clock input/output
FPGA	Kintex-7 410T FFG900 package
FPGA speed grade	-2 (XC7K410T-2FFG900)
DRAM	• 2 GB onboard DRAM • 166 MHz clock frequency • Bit-width: 512-bit
Backplane connection	Gen 2x8 PXI Express, PCIe 2.0 compliant

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Clocking Architecture

The PXIe-6592R modules include dedicated clocking hardware to provide a flexible clocking solution for generating the high-speed serial transceiver reference clocks (MGT_RefClk). Use the **Clocking and IO** properties page in the LabVIEW project to configure the clock settings for your module.

Table 4-2. PXIe-6592R Reference Clocks

Clock Name	Frequency Range	Available Sources
MGT_RefClk0	60 MHz to 700 MHz	Backplane: PXIe_Clk100 and PXIe_DStarA Front panel: CLK IN/OUT
MGT_RefClk1		

Refer to the *Configuring the High-Speed Serial Device LabVIEW FPGA Targets* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for more detailed information about configuring clocking in your LabVIEW project.

PXIe-6592R Clocking

The PXIe-6592R clocking architecture includes the following MGT Reference Clocks:

- MGT_RefClk0
- MGT_RefClk1
- MGT_RefClk2

MGT_RefClk0 and MGT_RefClk1 are separate clocks, but are derived though simple integer division of a common, higher frequency PLL clock. MGT_RefClk2 is an independent, free-running 156.25 MHz clock. Refer to the following table for information about the clocks' supported frequency ranges and available sources.

Table 4-3. PXIe-6592R Reference Clocks

Clock Name	Frequency Range	Available Sources
MGT_RefClk0	60 MHz to 670 MHz	Backplane: PXIe_Clk100 and PXIe_DStarA Front panel: PFI 0/CLK IN/OUT Other: 10 MHz Onboard Clock
MGT_RefClk1		
MGT_RefClk2	156.25 MHz	—

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 4-4. PXIe-6592R Reference Clocks

Clock Name	Frequency Range	Available Sources
MGT_RefClk0	60 MHz to 700 MHz	Backplane: PXIe_Clk100 and PXIe_DStarA Front panel: CLK IN/OUT
MGT_RefClk1		

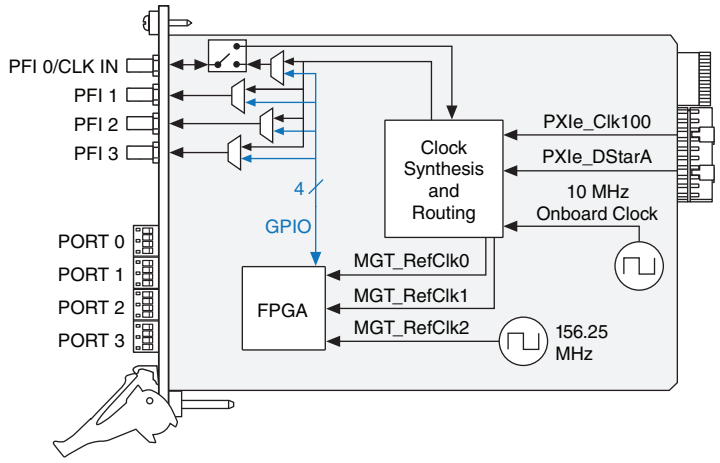
Refer to the *Configuring the High-Speed Serial Device LabVIEW FPGA Targets* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for more information about how to configure Reference Clocks for your device.

You can also configure any of the four front panel PFI 0, PFI 1, PFI 2, or PFI 3 connectors to export a clock from the module.

Note If you configure the front panel PFI connectors to export a clock from the module, all of the PFI connectors must be the same frequency.

The following figure illustrates the clocking circuitry on the PXIe-6592R.

Figure 4-2. PXIe-6592R Clocking Diagram



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5

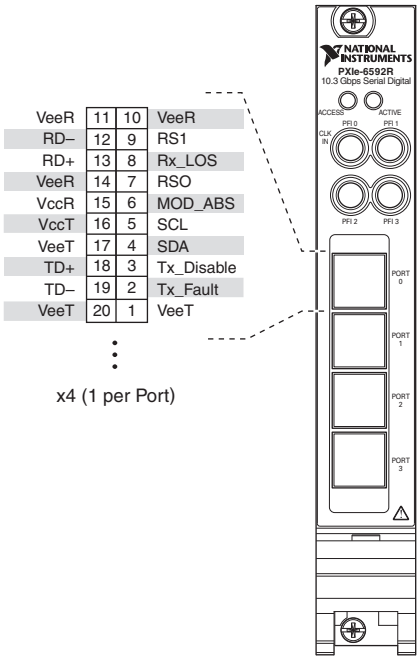
Connecting and Interfacing with the PXIe-6592R

This chapter contains information about the PXIe-6592R module and its functionality, including front panel diagrams, connectors, and pinouts.

Front Panel

The following figure shows the pinouts for the PXIe-6592R front panel connectors.

Figure 5-1. PXIe-6592R Front Panel Connectors and Pinout



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Refer to the following table for a list of the PXIe-6592R front panel connectors and their descriptions.

Table 5-1. PXIe-6592R Front Panel Connectors

Connector	Type	Description
PFI 0/CLK IN/OUT	SMB	Reference Clock input, exported clock output, and general-purpose I/O.
PFI 1/CLK OUT		
PFI 2/CLK OUT		
PFI 3/CLK OUT		
Port 0	SFP+	High-speed serial interfacing ports
Port 1		
Port 2		
Port 3		

Recommended Mating Cables and Connectors

Refer to the PXIe-6592R product listing page for a list of mating cables to use with your PXIe-6592R.

Transceiver Lane and Quad Mapping

If your application requires multiple lanes, refer to Table 5-2 and Table 5-3 for information about transceiver and RefClk selection when using the Xilinx tools to generate protocol IP.

Table 5-2. Transceiver Lane and Quad Mapping

Connector	Lane	Quad Location	Physical Resource
PORT 0	0	Quad 3 (Q3)	GTX_X0Y15
PORT 1	0		GTX_X0Y13
PORT 2	0	Quad 2 (Q2)	GTX_X0Y10
PORT 3	0		GTX_X0Y8

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-3. Clock Signal and Quad Mapping

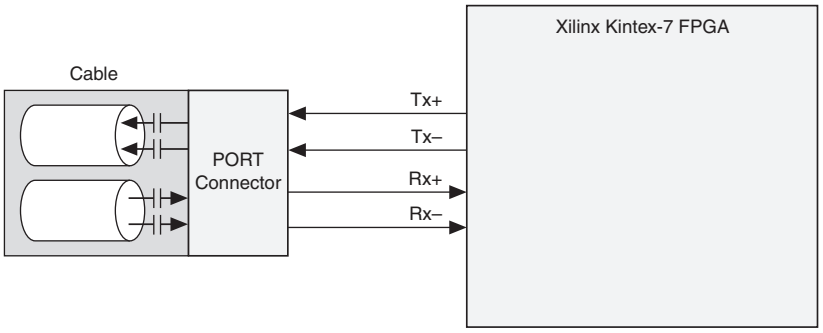
Clock Signal	Quad Location	Physical Resource
MGT_RefClk0	Quad 3 (Q3)	REFCLK1_Q3
MGT_RefClk1	Quad 2 (Q2)	REFCLK0_Q2
MGT_RefClk2	Quad 3 (Q3)	REFCLK0_Q3

For more information about lane and channel bonding caveats, including cases where you need to use single or multiple reference clocks for single or multiple transceivers, refer to *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) at `xilinx.com`.

Signal Routing

The PXIe-6592R high-speed serial differential signals are routed directly from the Kintex-7 FPGA pins to the PORT 0, PORT 1, PORT 2, and PORT 3 connector pins, as shown in the following figure.

Figure 5-2. PXIe-6592R Signal Routing



Socketed CLIP Interface

Socketed CLIP allows you to insert HDL IP into an FPGA target, enabling VHDL code to communicate directly with an FPGA VI. Socketed CLIP also allows the CLIP to communicate directly with circuitry external to the FPGA.

The following sections provide information about how to configure your device for use with socketed CLIP.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

PXIe-6592R Socketed CLIP

Refer to the following diagram for an overview of the PXIe-6592R socketed CLIP interface.

Figure 5-3. PXIe-6592R Socketed CLIP Diagram

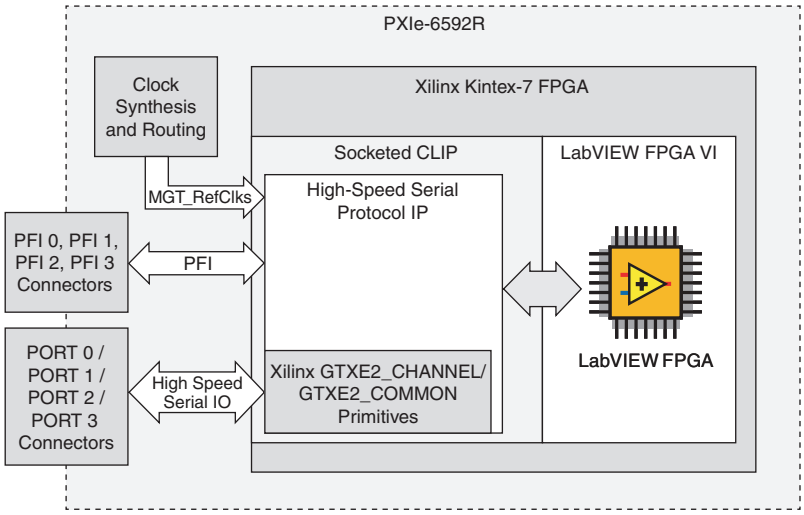


Table 5-4. PXIe-6592R CLIP Signals

Port	Direction	Clock Domain	Description
MGT_RefClk0_p	In (pad)	N/A	Differential input clock that you must connect to an IBUFDS_GTE2 input buffer primitive when this input clock is used in your design.
MGT_RefClk0_n	In (pad)	N/A	
MGT_RefClk1_p	In (pad)	N/A	
MGT_RefClk1_n	In (pad)	N/A	
MGT_RefClk2_p	In (pad)	N/A	Fixed 156.25 MHz Reference Clock for the transceivers that you must connect to an IBUFDS_GTE2 input buffer primitive when this input clock is used in your design.
MGT_RefClk2_n	In (pad)	N/A	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
MGT_RefClks_ ExtPllLocked	In	Async	Indicates the state of the PLL within the clocking logic and provides the Reference Clock to the FPGA MGTs (**MGT_ RefClk*x*** signals). Use this signal with **MGT_ RefClks_Valid** to gate and/or reset the clocking signals into any CLIP that depends on the **MGT_RefClk*x*** signals.
MGT_RefClks_ Valid	In	Async	Indicates if the selected clock input to the clocking logic is valid and the PLL within the clocking logic has locked. Use this signal to gate and/or reset the clocking signals into any CLIP that depends on the **MGT_RefClk*x*** signals. On the rising edge of **MGT_ RefClks_Valid**, you may need to reset or relock state machines and/or internal PLLs sensitive to **MGT_RefClk*x*** signals.
DebugClks(3:0)	Out	Clock	Debug ports to aid in debugging the clocking connections in the CLIP. These ports connect to frequency counters that can monitor the frequency of any clock that you connect to these ports. Refer to the *Debugging Clocks Using Frequency Counters* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for details about how to use these signals.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
ExportedUser ReferenceClk	Out	Clock	Reserved for future use.
LED_ActiveRed	Out	Async	The front panel Active indicator's red LED turns on when this signal is driven high. The CLIP's access to this LED may be temporarily overridden to show error conditions, temperature faults, and power faults. This signal is conditioned with the pulse stretcher to guarantee a minimum assertion time of 100 ms to comply with PXI guidelines and to facilitate visual perception. You can drive this signal asynchronously if you provide a 50 ns minimum assertion time. You can also drive this signal synchronously for a minimum 1 cycle of SocketClk40.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
LED_ActiveGreen	Out	Async	The front panel Active indicator's green LED turns on when this signal is driven high. The CLIP's access to this LED may be temporarily overridden to show error conditions, temperature faults, and power faults. This signal is conditioned with the pulse stretcher to guarantee a minimum assertion time of 100 ms to comply with PXI guidelines and to facilitate visual perception. You can drive this signal asynchronously if you provide a 50 ns minimum assertion time. You can also drive this signal synchronously for a minimum 1 cycle of SocketClk40.
SocketClk40	In	Clock	A 40 MHz clock that runs continuously regardless of connectivity. This signal is connected to the **40 MHz Onboard Clock** signal, which is the default top-level clock for the LabVIEW FPGA VI.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
sFrontEnd ConfigurationDone	In	SocketClk40	Asserts high and stays high when the power-on self-configuration (POSC) state machine is finished with configuration. After the **aResetSl** signal transitions from high to low, indicating that the CLIP logic should come out of reset, a POSC reconfiguration occurs unconditionally. The required clocking signals are not valid until after this signal asserts high.
sFrontEnd ConfigurationPrepare	In	SocketClk40	Reserved for future use. NI recommends assigning this signal to **sFront EndConfigurationReady**.
sFrontEnd ConfigurationReady	Out	SocketClk40	Reserved for future use. NI recommends assigning this signal to **sFront EndConfigurationPrepare**.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
aResetSl	In	Async	This signal is not required. This signal is an asynchronous reset signal from the LabVIEW FPGA environment. If you create an input signal to your CLIP and assign it as **Reset** in the CLIP wizard, that signal is driven as an asynchronous reset signal. Reset all CLIP state machines and logic whenever this signal is logic high. This signal is driven high when you call the LabVIEW FPGA **Reset** invoke method. Call **Run** on the FPGA VI to deassert this signal. Do not use CLIP inputs from the LabVIEW FPGA VI in the CLIP until **aResetSl** is deasserted.
Port<0..3>_RX_p	In (pad)	N/A	Dedicated MGT receive signals for Port <0..3>.
Port<0..3>_RX_n	In (pad)	N/A	
Port<0..3>_TX_p	Out (pad)	N/A	Dedicated MGT transmit signals for Port <0..3>.
Port<0..3>_TX_n	Out (pad)	N/A	
Port<0..3>_Mod_ ABS	In	Async	This signal is asserted when the module for Port <0..3> is absent. This signal is also called MODDEF0. This signal is grounded when a module is connected to indicate that the module is present. This signal is pulled asserted when no module is present.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
Port<0..3>_RS0	Out	Async	Drives the Port <0..3> SFP+ module's RX rate select signal. If this signal's RX rate is more than 4.25 Gbps, drive this signal high. If this signal's RX rate is 4.25 Gbps or less, drive this signal low.
Port<0..3>_RS1	Out	Async	Drives the Port <0..3> SFP+ module's TX rate select signal. If this signal's TX rate is more than 4.25 Gbps, drive this signal high. If this signal's TX rate is 4.25 Gbps or less, drive this signal low.
Port<0..3>_Rx_LOS	In	Async	This signal is driven by the Port <0..3> SFP+ module to indicate that the SFP+ module cannot detect an RX signal.
Port<0..3>_Tx_Disable	Out	Async	Drives the Port <0..3> SFP+ module's Tx_Disable signal. Driving this signal high disables the transmitter for the respective port.
Port<0..3>_Tx_Fault	In	Async	This signal is driven by the Port <0..3> SFP+ module to indicate a TX fault. This signal indicates that the module laser is operating incorrectly.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
Port<0..3>_SCL	In/Out	Async	Bidirectional serial clock signal for the two wire communication interface on the Port<0..3> connector. Valid values are `0` and `Z` (open drain). This signal is also called **MODDEF1**. This signal has a 10 kΩ pull up to +3.3V. You must assert **sPort<0..3>_EnablePower** to enable the **Port<0..3>_SCL** and **Port<0..3>_SDA** interface.
Port<0..3>_SDA	In/Out	Async	Bidirectional serial data signal for the two wire communication interface on the Port<0..3> connector. Valid values are `0` and `Z` (open drain). This signal is also called **MODDEF2**. This signal has a 10 kΩ pull up to +3.3V. You must assert **sPort<0..3>_EnablePower** to enable the **Port<0..3>_SCL** and **Port<0..3>_SDA** interface.
sPort<0..3>_Enable Power	Out	SocketClk40	Enables or disables the optical power supply on Port <0..3>. Assert this signal to enable the optical supply for its corresponding port.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 5-4. PXIe-6592R CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
sPort<0..3>_Power Good	In	SocketClk40	Indicates that the optical power supply for Port <0..3> is enabled. This signal may deassert if an over-power condition occurs.
PFI<0..3>_GPIO_In	In	Async	Acquires GPIO input from the PFI<0..3> connectors.
PFI<0..3>_GPIO_Out	Out	Async	Drives the GPIO output data to the PFI<0..3> connectors.
PFI<0..3>_GPIO_ OutEnable_n	Out	Async	Enables or disables the GPIO output driver on PFI<0..3>. This signal is active low.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6

PXIe-7902 Hardware Architecture

The following chapter contains information about the PXIe-7902 clocking architecture.

The PXIe-7902 is a high-speed serial interfacing module. The PXIe-7902 hardware architecture allows you to fully customize your serial digital protocol application. The high-speed serial interface uses Xilinx GTX transceiver technology; you can reuse existing protocol IP that works with Xilinx GTX transceivers, or you can develop your own protocol IP. If you develop your own protocol IP, the IP must be developed for a Xilinx Virtex-7 GTX transceiver.

Note The PXIe-7902 hardware does not require calibration.

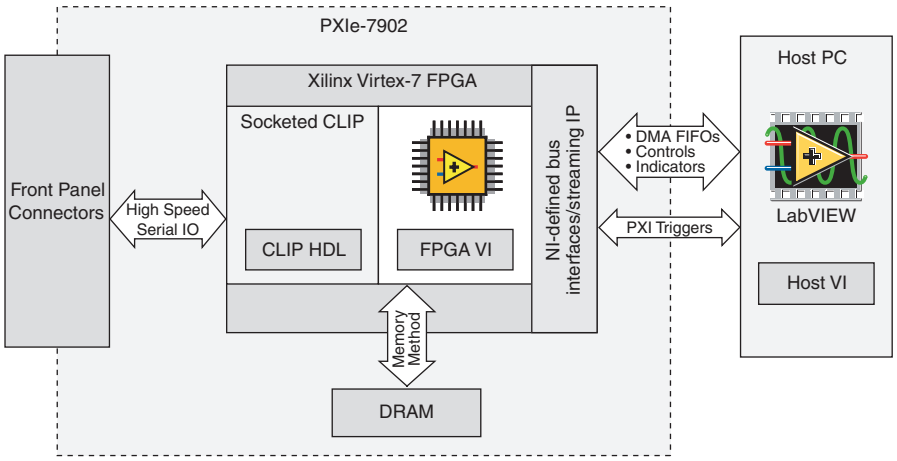
PXIe-7902-based projects consist of the following key components:

- PXIe-7902 front panel connectors for data, clocking, and triggering external to the module
- Socketed CLIP for HDL IP and interface definition from the FPGA VI to the PXIe-7902 front panel
- Xilinx Virtex-7 FPGA
- Dynamic random access memory (DRAM)
- NI-defined bus interface from the FPGA to the host PC

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following figure illustrates the key components of the PXIe-7902 architecture.

Figure 6-1. PXIe-7902 System Architecture Elements



PXIe-7902 Module Overview

The PXIe-7902 modules include the following key features. Refer to the *PXIe-7902 Specifications* for more details.

Table 6-1. PXIe-7902 Key Features

Line rate	500 Mb/s to 8 Gb/s, and 9.8 Gb/s to 12.5 Gb/s serial
Multi-gigabit transceiver lanes	24 (4 per port)
Front Panel Connectors	• Six mini-SAS x4 connectors (Port 0, Port 1, Port 2, Port 3, Port 4, and Port 5) for high-speed serial • One SMB connector (CLK IN) for clock input
FPGA	Xilinx Virtex-7 485T FFG1158 package
FPGA speed grade	-3 (XC7VX485T-3FFG1158E)
DRAM	• 2 GB onboard DRAM • 166 MHz clock frequency • Bit-width: 512-bit
Backplane connection	Gen 2x8 PXI Express, PCIe 2.0 compliant

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Clocking Architecture

The PXIe-7902 modules include dedicated clocking hardware to provide a flexible clocking solution for generating the high-speed serial transceiver reference clocks (MGT_RefClk). Use the **Clocking and IO** properties page in the LabVIEW project to configure the clock settings for your module. Refer to the *Configuring the High-Speed Serial Device LabVIEW FPGA Targets* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for more detailed information about configuring clocking in your LabVIEW project.

PXIe-7902 Clocking

The PXIe-7902 clocking architecture includes the following MGT Reference Clocks:

- MGT_RefClk0
- MGT_RefClk1
- MGT_RefClk2

MGT_RefClk0, MGT_RefClk1, and MGT_RefClk2 are separate clocks, but are derived though simple integer division of a common, higher frequency PLL clock. Refer to the following table for information about the clocks' supported frequency ranges and available sources.

Table 6-2. PXIe-7902 Reference Clocks

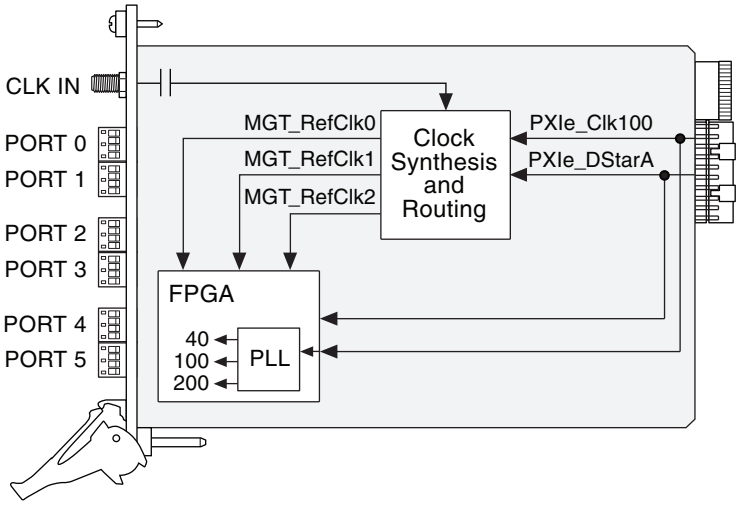
Clock Name	Frequency Range	Available Sources
MGT_RefClk0 MGT_RefClk1 MGT_RefClk2	60 MHz to 670 MHz 156.25 MHz	Backplane: PXIe_Clk100 and PXIe_DStarA Front panel: CLK IN Other: 10 MHz Onboard Clock —

Refer to the *Configuring the High-Speed Serial Device LabVIEW FPGA Targets* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for more information about how to configure Reference Clocks for your device.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following figure illustrates the clocking circuitry on the PXIe-7902.

Figure 6-2. PXIe-7902 Clocking Diagram



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

7

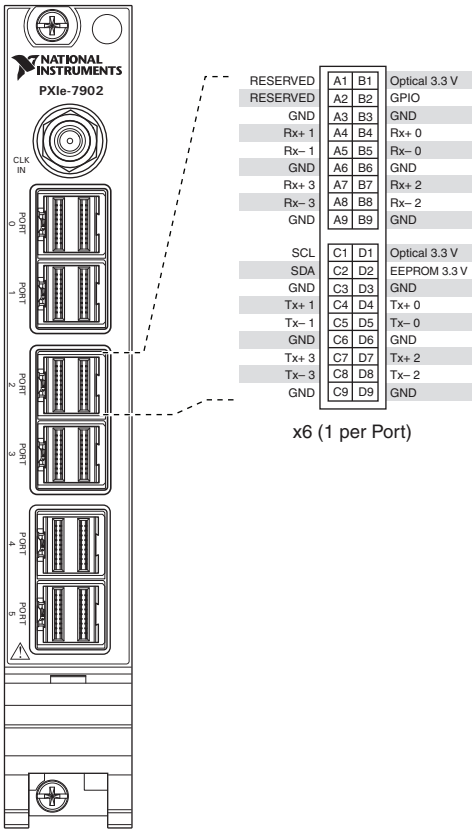
Connecting and Interfacing with the PXIe-7902

This chapter contains information about the PXIe-7902 module and its functionality, including front panel diagrams, connectors, and pinouts.

Front Panel

The following figure shows the pinouts for the PXIe-7902 front panel connectors.

Figure 7-1. PXIe-7902 Front Panel Connectors and Pinout



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Refer to the following table for a list of the PXIe-7902 front panel connectors and their descriptions.

Table 7-1. PXIe-7902 Front Panel Connectors

Connector	Type	Description
CLK IN	SMB	Reference Clock input and general-purpose I/O
Port 0	Mini-SAS HD x4	High-speed serial interfacing ports
Port 1		
Port 2		
Port 3		
Port 4		
Port 5		

Recommended Mating Cables and Connectors

Refer to the PXIe-7902 product listing page for a list of mating cables to use with your PXIe-7902.

Transceiver Lane and Quad Mapping

If your application requires multiple lanes, refer to Table 7-2 and Table 7-3 for information about transceiver and RefClk selection when using the Xilinx tools to generate protocol IP.

Table 7-2. Transceiver Lane and Quad Mapping

Connector	Lane	Quad Location	Physical Resource
PORT 0	0	Quad 3 (Q3)	GTX_X0Y15
PORT 1	0		GTX_X0Y13
PORT 2	0	Quad 2 (Q2)	GTX_X0Y10
PORT 3	0		GTX_X0Y8
PORT 4			
PORT 5			

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 7-3. Clock Signal and Quad Mapping

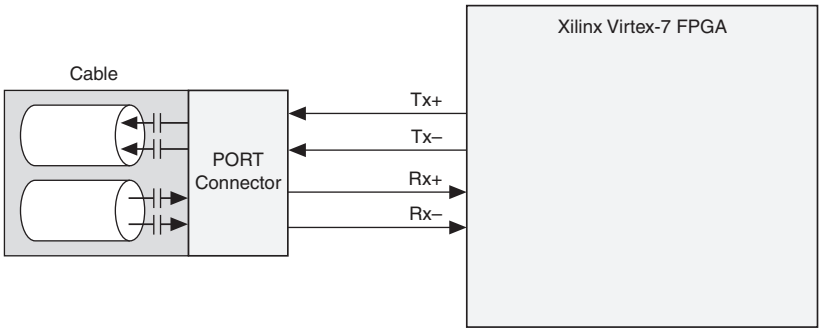
Clock Signal	Quad Location	Physical Resource
MGT_RefClk0	Quad 3 (Q3)	REFCLK1_Q3
MGT_RefClk1	Quad 2 (Q2)	REFCLK0_Q2
MGT_RefClk2	Quad 3 (Q3)	REFCLK0_Q3

For more information about lane and channel bonding caveats, including cases where you need to use single or multiple reference clocks for single or multiple transceivers, refer to *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) at `xilinx.com`.

Signal Routing

The PXIe-7902 high-speed serial differential signals are routed directly from the Virtex-7 FPGA pins to the PORT 0, PORT 1, PORT 2, PORT 3, PORT 4, and PORT 5 connector pins, as shown in the following figure. This signal routing is replicated for every lane across every port.

Figure 7-2. PXIe-7902 Signal Routing



Socketed CLIP Interface

Socketed CLIP allows you to insert HDL IP into an FPGA target, enabling VHDL code to communicate directly with an FPGA VI. Socketed CLIP also allows the CLIP to communicate directly with circuitry external to the FPGA.

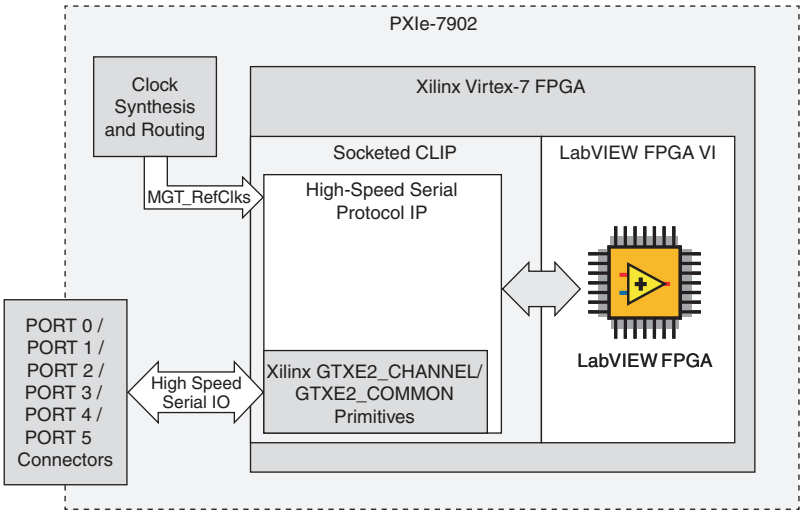
The following sections provide information about how to configure your device for use with socketed CLIP.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

PXIe-7902 Socketed CLIP

Refer to the following diagram for an overview of the PXIe-7902 socketed CLIP interface.

Figure 7-3. PXIe-7902 Socketed CLIP Diagram



The following signals are provided through the PXIe-7902 CLIP socket. You can use these signals to develop your own custom CLIP. For more information about how to configure the PXIe-7902 CLIP, refer to the *NI High-Speed Serial Instruments Help*.

Table 7-4. PXIe-7902 Socketed CLIP Signals

Port	Direction	Clock Domain	Description
MGT_RefClk2_p	In (pad)	N/A	Differential input clock that you must connect to an IBUFDS_GTE2 input buffer primitive when this input clock is used in your design.
MGT_RefClk2_n	In (pad)	N/A	
MGT_RefClk3_p	In (pad)	N/A	Differential input clock that you must connect to an IBUFDS_GTE2 input buffer primitive when this input clock is used in your design.
MGT_RefClk3_n	In (pad)	N/A	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 7-4. PXIe-7902 Socketed CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
MGT_RefClk4_p	In (pad)	N/A	Differential input clock that you must connect to an IBUFDS_GTE2 input buffer primitive when this input clock is used in your design.
MGT_RefClk4_n	In (pad)	N/A	
Port<0..5>_RX_p	In (pad)	N/A	Dedicated MGT receive signals for Port <0..5>.
Port<0..5>_RX_n	In (pad)	N/A	
Port<0..5>_TX_p	Out (pad)	N/A	Dedicated MGT transmit signals for Port <0..5>.
Port<0..5>_TX_n	Out (pad)	N/A	
GtxRxPolarity_in	In (pad)		24-bit signal that indicates the polarity of the receive signal. Some MGT signals have inverted polarity external to the FPGA; use this signal to determine which channels, if any, are inverted.
GtxTxPolarity_in	In (pad)		24-bit signal that indicates the polarity of the transmit signal. Some MGT signals have inverted polarity external to the FPGA; use this signal to determine which channels, if any, are inverted.
SocketClk40	In	Clock	A 40 MHz clock that runs continuously regardless of connectivity. This signal is connected to the **40 MHz Onboard Clock** signal, which is the default top-level clock for the LabVIEW FPGA VI.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 7-4. PXIe-7902 Socketed CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
Port<0..5>_SCL	In/Out	Async	Bidirectional serial clock signal for the two wire communication interface on the Port<0..5> connector. Valid values are `0` and `Z` (open drain). This signal is also called **MODDEF1**. This signal has a 10 kOhm pull up to +3.3V. You must assert **aPort<_EnablePower** to enable the **Port<0..5>_SCL** and **Port<0..5>_SDA** interface.
Port<0..5>_SDA	In/Out	Async	Bidirectional serial data signal for the two wire communication interface on the Port<0..3> connector. Valid values are `0` and `Z` (open drain). This signal is also called **MODDEF2**. This signal has a 10 kOhm pull up to +3.3V. You must assert **aPort_EnablePower** to enable the **Port<0..5>_SCL** and **Port<0..5>_SDA** interface.
Port<0..5>_ModPrs_n	In (pad)		Indicates the presence of a device on the other end of the cable for each port.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 7-4. PXIe-7902 Socketed CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
MGT_RefClks_ ExtPllLocked	In	Async	Indicates the state of the PLL within the clocking logic and provides the Reference Clock to the FPGA MGTs (**MGT_ RefClk*x*** signals). Use this signal with **MGT_ RefClks_Valid** to gate and/or reset the clocking signals into any CLIP that depends on the **MGT_RefClk*x*** signals.
MGT_RefClks_ Valid	In	Async	Indicates if the selected clock input to the clocking logic is valid and the PLL within the clocking logic has locked. Use this signal to gate and/or reset the clocking signals into any CLIP that depends on the **MGT_RefClk*x*** signals. On the rising edge of **MGT_ RefClks_Valid**, you may need to reset or relock state machines and/or internal PLLs sensitive to **MGT_RefClk*x*** signals.
DebugClks(5:0)	Out	Clock	Debug ports to aid in debugging the clocking connections in the CLIP. These ports connect to frequency counters that can monitor the frequency of any clock that you connect to these ports. Refer to the *Debugging Link Connections Using Eye Scan* section of Chapter 8, *Developing Applications for the High-Speed Serial Device*, for details about how to use these signals.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 7-4. PXIe-7902 Socketed CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
ExportedUser ReferenceClk	Out	Clock	Reserved for future use.
sFrontEnd ConfigurationDone	In	SocketClk40	Asserts high and stays high when the power-on self-configuration (POSC) state machine is finished with configuration. After the **aResetSl** signal transitions from high to low, indicating that the CLIP logic should come out of reset, a POSC reconfiguration occurs unconditionally. The required clocking signals are not valid until after this signal asserts high.
sFrontEnd ConfigurationPrepare	In	SocketClk40	Reserved for future use. NI recommends assigning this signal to **sFront EndConfigurationReady**.
sFrontEnd ConfigurationReady	Out	SocketClk40	Reserved for future use. NI recommends assigning this signal to **sFront EndConfigurationPrepare**.
aPort_Enable Power	Out	SocketClk40	Enables or disables the power supply on Port <0..5>. Assert this signal to enable the power supply for its corresponding port. You can leave this signal disabled when using copper cables. This signal is required when using optical cables.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 7-4. PXIe-7902 Socketed CLIP Signals (Continued)

Port	Direction	Clock Domain	Description
aPort_Power Good	In	SocketClk40	Indicates that the power supply for Port <0..5> is enabled. This signal may deassert if an over-power condition occurs.
aOptical_Enable Power	Out	SocketClk40	Enables or disables the optical power supply on Port <0..5>. Assert this signal to enable the optical supply for its corresponding port.
aOptical_Power Good	In	SocketClk40	Indicates that the optical power supply for Port <0..5> is enabled. This signal may deassert if an over-power condition occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8

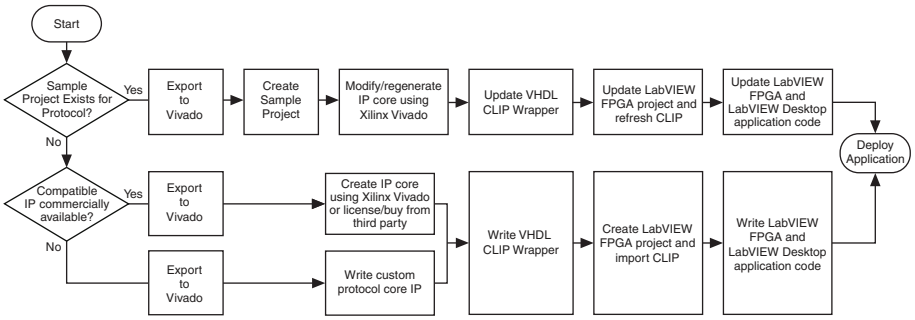
Developing Applications for the High-Speed Serial Device

This chapter provides information about how to develop applications for the high-speed serial device, including information about creating socketed CLIP and using LabVIEW.

Development Flow

Refer to the following diagram for an overview of the high-speed serial development process.

Figure 8-1. High-Speed Serial Development Process



If the sample project code is sufficient for your application, you do not have to modify the IP core, update the VHDL CLIP wrapper, or refresh the CLIP.

Socketed CLIP Development

This section provides steps for creating socketed CLIP for use with your application. Socketed CLIP provides the following functionality:

- Allows you to insert HDL IP into an FPGA target, enabling VHDL code to communicate directly with an FPGA VI.
- Allows the CLIP to communicate directly with circuitry external to the FPGA.
- Allows your IP to communicate directly with both the FPGA VI and the external FPGA module connector interface.

You can develop socketed CLIP either by using the Xilinx Vivado tools, or by exporting a LabVIEW FPGA VI as a Vivado Design Suite project.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

- The Xilinx Vivado tools create a blank project, from which you can develop socketed CLIP. For more information about using the Xilinx Vivado tools to develop socketed CLIP, refer to the *Accessing the Xilinx Vivado Tools* section.
- You can also design your project in LabVIEW, then export the project to the Vivado Design Suite. For more information about exporting a LabVIEW FPGA VI as a Vivado Design Suite project, refer to the *Exporting to Vivado* section.

Accessing the Xilinx Vivado Tools

Complete the following steps to run Xilinx Vivado and generate a blank project for CLIP development.

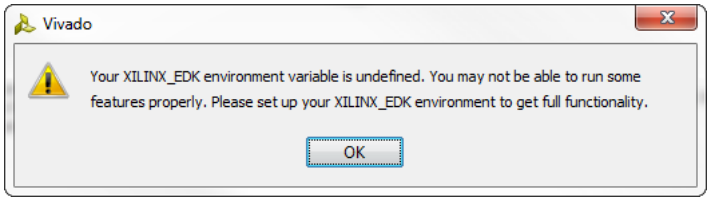
1. Ensure that your system meets the system requirements listed in the *NI High-Speed Serial Instruments Readme*, available from the Start menu and at `ni.com/manuals`. You can check the versions of the tool on your machine by selecting **Start»Control Panel» Programs and Features**.

   Note If Vivado is installed by LabVIEW FPGA, it does not appear in **Programs and Features**.

2. Open the Xilinx Vivado Tool directory by navigating to `C:\NIFPGA\programs\VivadoXXXX_Y`, where `XXXX` and `Y` refer to the Xilinx Vivado tool versions. For example, <VIVADO_DIR> version 2013.4 is located at `C:\NIFPGA\programs\Vivado2013_4`.
3. Run the Xilinx Vivado batch file: `<XilinxVivadoDir>\bin\vivado.bat`.

   You may receive the following warning when launching Vivado.

Figure 8-2. EDK Environment Error Message



   This error message is expected. You can ignore the error message if you are not using the Xilinx Embedded Development Kit (EDK). The EDK is not required for development with the high-speed serial device.
4. Click **New Project** and follow the instructions in the wizard.

You can also export an FPGA VI as a Vivado Design Suite project, as detailed in the *Exporting to Vivado* section.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

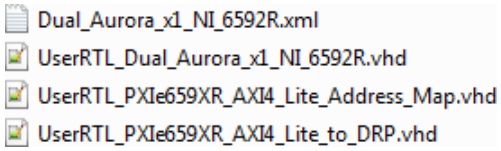
Exporting to Vivado

Exporting a LabVIEW FPGA VI as a Vivado project allows you to design the exported project in LabVIEW, then compile it into a bitfile in the Vivado Design Suite. You can then run the bitfile on an FPGA target in the FPGA Module. This option takes advantage of the design features provided by the Vivado Design Suite while making full use of LabVIEW FPGA hardware resources. Refer to the Exporting FPGA VIs as Vivado Design Suite Projects (FPGA Module) topic in the *LabVIEW Help* for more information about this feature.

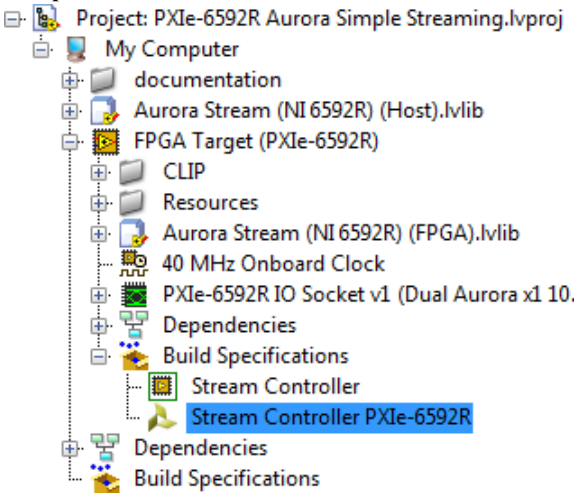
The PXIe-6591R, PXIe-6592R, and PXIe-7902 devices support exporting to Vivado. The Aurora sample projects for the PXIe-6591R, PXIe-6592R, and PXIe-7902R provide out-of-the-box support that demonstrates how to use hardware design files as an entry point when exporting to Vivado.

Complete the following steps to export a LabVIEW FPGA VI as a Vivado Design Suite project.

1. Open the Aurora Simple Streaming sample project for your hardware device. The following image shows the PXIe-6592R Aurora Simple Streaming sample project as an example. The files with the `UserRTL_` prefix are used as the entry point when the top-level FPGA VI is exported to Vivado.



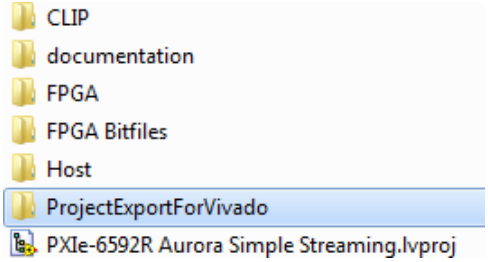
2. In the LabVIEW project, select the **Stream Controller [device name]** build specification under the **Build Specifications** item.



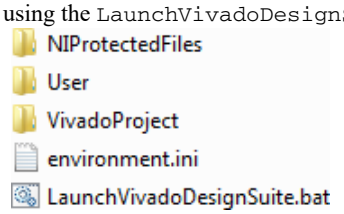
3. Navigate to the LabVIEW project root directory. The exported project is located inside the **ProjectExportForVivado** folder. The exported project contains encrypted LabVIEW

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

FPGA files, the unencrypted design files with the `UserRTL_` prefix, and the Vivado project files.



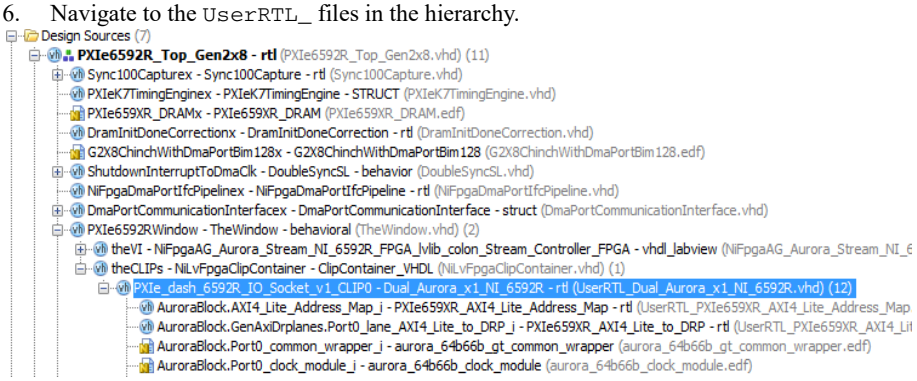
4. Open the Vivado project using the `LaunchVivadoDesignSuite.bat` file.



5. The source hierarchy loads once Vivado launches.

Note The hierarchy source is encrypted, except for the design files prefixed with `UserRTL_` and added to the FPGA target as a socketed CLIP.

6. Navigate to the `UserRTL_` files in the hierarchy.



7. Use the unencrypted design files as an entry point for IP you are developing in Vivado. You can compile the design from Vivado to generate a LabVIEW bitfile (`.lvbitx`) that can be loaded on the respective NI targets.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Generating an IP Core from the Xilinx Vivado IP Catalog

You may need to purchase and install additional licenses to generate some protocol IP core from Xilinx or third-party IP vendors. Refer to *UG 973: Vivado Design Suite: Release Notes, Installation, and Licensing* at `xilinx.com` for information about managing licenses.

Complete the following steps to create a Xilinx Vivado project:

1. Refer to the *Xilinx Licensing Information* section of Chapter 1, *Before You Begin*, for information about licensing before creating a Xilinx Vivado project.
2. Launch the Xilinx Vivado IP catalog.
   a. Select **Manage IP** on the Vivado start screen.
   b. Locate the appropriate IP core to launch the configuration dialog. For example, the Aurora 64B66B IP core is located in **Communication and Networking»Serial Interfaces»Aurora 64B66B**.
3. Select the IP core settings. NI recommends that you select AXI4-Stream for high-speed data streams when possible.

Note NI does not recommend selecting AXI4-Lite for DRP accesses in the Xilinx IP cores because compatibility with LabVIEW FPGA AXI4-Lite adapters cannot be guaranteed. Refer to the Aurora sample projects for an example of how to use the LabVIEW FPGA AXI4-Lite adapters to connect to DRP within the CLIP.

Modifying Third-Party IP Core Logic

If you modify a third-party IP core for your high-speed serial protocol, consult the *Xilinx Product Guide* for the IP you are using before attempting to make any modifications.

Adhere to the following guidelines when modifying third-party IP core logic:

- Ensure all clocks are connected.
- Ensure AXI4-Lite management signals are connected correctly to the Xilinx DRP signals on the GTXE2_CHANNEL and GTXE2_COMMON primitives.
- Select **Include Shared Logic in example design** in the IP wizard to access various resources outside of the IP core logic, such as MGT_RefClk input buffers and QPLL wrappers.

The following examples explain the differences in how the IBUFDS_GTE2 resource is exposed with and without the **Include Shared Logic in example design** option.

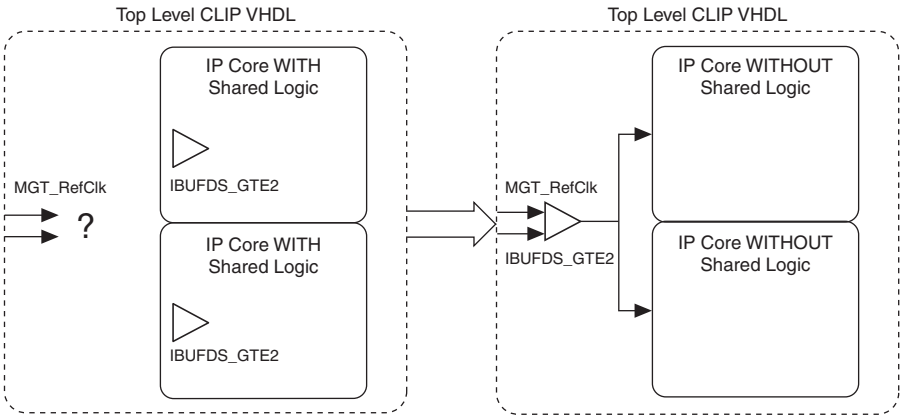
- Option 1: Include the IBUFDS_GTE2 input buffer primitive inside the core by selecting **Include Shared Logic in core** in the IP wizard. The image on the left in Figure 8-3 shows this option.
- Option 2: Instantiate a single IBUFDS_GTE2 input buffer in your top level CLIP VHDL, connect its output signal to both cores, and select **Include Shared Logic in example design** in the IP wizard. The image on the right in Figure 8-3 shows this option.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note Do not modify the IP core unless you understand the required reference clock(s) and clocking resources.

The following figure shows the difference between the top-level CLIP VHDL with shared logic in the core (left) and without shared logic (right).

Figure 8-3. Top-Level CLIP VHDL and Shared Logic



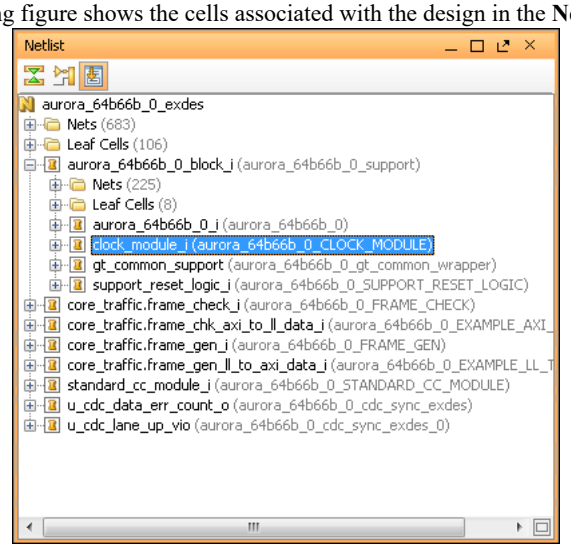
Building a Netlist from the IP Core

LabVIEW FPGA does not support Verilog source files in Component Level IP. However, you can generate EDIF netlists from any synthesized Verilog components in the IP you are using and instantiate the netlist in a VHDL wrapper.The following steps are an example of how to generate an EDIF netlist from the IP core:

1. Open the example project for your IP core in Vivado.
2. Set the appropriate top-level source file for which you plan to generate a netlist.
3. Run synthesis.
4. Open the Synthesized Design using one of the following methods.
   - Select **Open Synthesized Design** in the **Synthesis Completed** pop-up window.
   - Select the **Design Run** tab, then select **Open Synthesized Design** in the left hand pane.
5. In the Tcl Console, enter `write_edif <name of entity>.edf` to create the netlist that you use when you import the IP core into your LabVIEW project. The netlist location is indicated by the Tcl Console window.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6. The following figure shows the cells associated with the design in the **Netlist** window.



7. To build `.edf` files for an associated cell, enter the following command:
   `write_edif -cell <name of cell> <file name>.edf`
   For example, to create an `.edf` for `clock_module_i`, enter the following command:
   `write_edif -cell clock_module_i`
   `aurora_64b66b_clock_module.edf`

Note You may have to specify a longer path name depending on the location of the cell in your project. For example, **clock_module_i** may be located under `aurora_64b66b_0_block_i/clock_module_i`.

8. Copy the netlist into your LabVIEW FPGA CLIP directory.
9. Include your netlist in the list of synthesis files when running the CLIP Wizard.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Writing a VHDL Wrapper Around the Protocol IP Core

A VHDL wrapper is generally necessary to adapt the protocol signals to the dataflow semantics used within the LabVIEW FPGA diagram. NI recommends that you adhere to the following guidelines when writing a VHDL wrapper around the protocol IP core:

- Keep the interface between the CLIP and the LabVIEW FPGA diagram as simple as possible.



Note LabVIEW stores values in big-endian format, and your IP may accept only little-endian format. NI recommends performing any conversions in the CLIP and keeping endian conversions off the LabVIEW diagram for ease of use.

- Do not pass asynchronous signals to the LabVIEW FPGA diagram. Register the signals in a clock domain in the VHDL logic before passing them to the LabVIEW FPGA diagram.
- Use AXI4-Stream and AXI4-Lite interfaces for streaming data and register accesses. NI provides AXI4-Stream and AXI4-Lite wrappers to use on the LabVIEW FPGA diagram. Refer to the *Generating an IP Core from the Xilinx Vivado IP Catalog* section of this chapter for more information about IP core logic.
- If you expose an AXI4-Lite endpoint, use Xilinx AXI4 interconnect IP to expose only one AXI4-Lite endpoint to the LabVIEW FPGA diagram.
- Document the frequency of clocks coming from CLIP. Consider supporting enable chain removal. Refer to the *Improving Performance in Larger Designs through Enable Chain Removal* section of this chapter for more information about how to enable chain removal.
- Implement a state machine that allows asynchronous resets. If you declare an input signal as a reset signal in the CLIP wizard, then that signal is asserted when the LabVIEW FPGA VI is not running.
- Implement a state machine that resets the protocol cores when the **PORT#** module is absent if your state machine does not already account for this.
- Connect various clocks from your CLIP to the DebugClks std_logic_vector in order to use host-side frequency counter debugging utilities.
- Provide timing constraints in XDC for your CLIP. Include timing constraints for clocks within your CLIP, but do not include pin/location constraints on MGTs transceiver lanes and RefClks. Refer to *UG 903: Vivado Design Suite User Guide: Using Constraints* at `xilinx.com` for more information about timing constraints in XDC for your CLIP.
- Use the TXOUTCLK and/or RXOUTCLK clock constraints for your high-speed serial CLIP if your protocol uses it directly.
  - The following is an example syntax for the constraint: `create_clock -period <period in ns> [get_pins %ClipInstancePath%/<path to your clock pin relative to the top level CLIP VHDL>]`.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

- If you generate an asynchronous reset within your CLIP VHDL, create a false path constraint from the register that generates the reset signal. Include a “don’t touch” attribute for any false path constraints.
  - The following is an example syntax for the “don’t touch” attribute: `attribute dont_touch : string; attribute dont_touch of <signal name> : signal is "true";`
  - The following is an example syntax for the false path constraint: `set_false_path -from [get_cells %ClipInstancePath%/<path to your register>]`
- When writing constraints, you may need to refer to the CLIP’s instance name or the absolute path to the CLIP instance in the VHDL hierarchy. Refer to the *Constraints and Hierarchy* section of this chapter for more information about using the search-and-replace keywords **%ClipInstanceName%** and **%ClipInstancePath%**.

Constraints and Hierarchy

You can include CLIP-specific user constraints in the compilation using a constraints file, depending on your specific FPGA target. You can use this mechanism for all constraints except pin placement constraints. For example, you can access a clock directly from a global clock input pin through a global clock buffer for socketed CLIP. You must constrain the period of this clock.

For constraints on specific components within CLIP, you might need to specify the location of the component within the overall VHDL hierarchy. In such cases, consider prefacing the constraints with the following macros. Prefacing allows the constraints to be applied regardless of the component location in the VHDL hierarchy. If you want to use this example code, copy the code to a text file and save the file as `DemoClipAdder.xdc`. Add this constraints file along with the VHD file as synthesis files in the Configuring CLIP wizard to implement this constraint.

Xilinx Vivado

```
create_clock -period 10.000 -name %ClipInstanceName%Clk -waveform
{0.000 5.000} -add [get_pins %ClipInstancePath%/clk]

set_clock_latency -clock [get_clocks {%ClipInstanceName%CLK}] 10.0
[get_pins {%ClipInstancePath%/cAddOut[0]}]
```

To instantiate the CLIP multiple times, each CLIP instance must have a unique name, and the name must follow VHDL naming conventions. When you include these macros, you do not need to include a separate constraints file for each instance because the FPGA Module creates a unique instance name.

If a CLIP signal is not used, the Xilinx compilation tools might remove the signal from the bitstream. In such cases, you might get an NGBuild error during compilation. To resolve this issue, remove the constraint or use the signal in an FPGA VI.

Caution In order to guarantee data integrity and timing closure, verify that I/O nodes from the CLIP are written in the same clock domain in which they are read on the LabVIEW diagram and that I/O nodes to the CLIP are read in the same clock

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

domain in which they are written on the LabVIEW diagram. In rare cases where crossing clock domains is desirable, refer to KnowledgeBase `6OB8E8FM` at `ni.com/kb` for more information about how to write timing constraints between the CLIP and the LabVIEW diagram in order to specify timing exceptions on these paths and achieve timing closure. Note that data corruption might still occur when crossing clock domains.

Documenting Your IP

NI recommends documenting the behavior of your CLIP. Refer to the following guidelines for information about how to document your CLIP and how documenting your CLIP can affect the rest of your design:

- Document the endianness of your CLIP in order to properly interface your CLIP to the LabVIEW FPGA diagram. Refer to the *Writing a VHDL Wrapper Around the Protocol IP Core* section of this chapter for more information about how CLIP endianess affects the design process.
- Clearly define the portion of your entity interface that is facing the diagram, and which portion of your entity is facing the front panel.
- Clearly define your LED behavior.
- Document the connector signals by describing which signals are used, which signals are unused, and the manner in which the signal is used. Signal use can affect which ports are active with your IP and the behavior of cables upon insetion and removal.
- Use the DebugClks signal to determine the health of the internal clocks being sent to the IP. Define which bits of the 4-bit vector correspond to the clock being monitored.
- Document how you integrate AXI4-Lite signals with LabVIEW data types. Some AXI4-Lite signals do not integrate easily with LabVIEW data types; for example, address ports can have widths of 11, but LabVIEW only provides addresses with widths of 8, 16, 32, and 64. Additionally, the AXI4-Lite and AXI4-Stream adapters are configured for use with fixed-point I/O.
- Document how clocks are used and how they are routed in your CLIP for use with the IP. You must route clocks to the diagram for use with the single-cycle timed loop (SCTL) in LabVIEW FPGA.
- Document the address map of individual components within any AXI4-Lite interfaces.

Improving Performance in Larger Designs through Enable Chain Removal

By default, LabVIEW adds code to the FPGA code to enforce data flow. This code addition is referred to as the enable chain. In larger applications, the enable chain can create routing congestion and limit performance. You can remove the enable chain under certain circumstances. Refer to *Improving Timing Performance in Large Designs (FPGA Module)* in the *LabVIEW FPGA Module Help* for more information about how to remove enable chains and when to do so.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Developing with LabVIEW FPGA

After configuring the CLIP in VHDL, you can use LabVIEW FPGA to continue the development process. LabVIEW FPGA provides FPGA target support, configuration for clocking and routing, and interfacing with LabVIEW on your host computer for a fully integrated development experience.

Refer to the *Related Documentation* section of *About This Manual* for a list of LabVIEW FPGA documentation that you may find helpful as you develop your application.

Configuring the High-Speed Serial Device LabVIEW FPGA Targets

Complete the following steps to configure your high-speed serial device LabVIEW project:

1. Create a new project by selecting **File»New»Project**, or open an existing project by selecting **File»Open**.
2. Right-click **My Computer** in the **Project Explorer** window and select **New»Targets and Devices** from the shortcut menu to display the **Add Targets and Devices** dialog box.
3. Select **New target or device** and select your device.
4. Add the protocol IP through your CLIP. Right-click the device name and select **Properties»Component-Level IP.**

Note If you are using example CLIP or pre-made CLIP, you can import the CLIP using the dialog box, or you can click on the **Create File** icon to create a new CLIP using the CLIP Wizard.

Note You can modify a CLIP by selecting the preexisting CLIP Declaration Name and clicking **Modify File**.

5. If you are generating new CLIP, follow the instructions in the CLIP Wizard to interface your CLIP with LabVIEW FPGA. You do not need to use the CLIP Wizard if you are reusing an existing CLIP. Refer to the *FPGA Module Help* for more detailed information about the CLIP Wizard. The CLIP Wizard guides you through the following tasks.
   - Adding VHDL source, XDC constraints, and EDF/EDN/EDIF netlists
   - Configuring device types
   - Configuring generics
   - Performing syntax checks
   - Specifying how to use the signals in your CLIP

Note In Step 2 of the CLIP Wizard, select the appropriate Component Level IP Type for your target.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note After you create the CLIP and add the files, you do not need to modify the CLIP for any changes to take place if you do not change the source paths. If you change the source paths or modify the CLIP source files, you must use the CLIP Wizard.

6. Instantiate the CLIP in the I/O socket. When you add a new target to the project, LabVIEW automatically creates a compatible I/O socket in the project. Right-click the socket and select **Properties**, then select **General** under **Category**.
7. Select a declaration from the drop-down menu under **Socketed Component Level IP Declaration**.
8. Click **OK**. The user-defined signals in your CLIP appear under the socket item.
9. Right-click the IO socket and select **Properties»Clocking and IO** to configure the Clocking and IO Configuration properties for your device.

Note Clocking and routing information is compile-time static and cannot be reconfigured at runtime.

Note If you are using an example project or a Sample Project, the **Clocking** and **IO Configuration** tabs are already configured. If you create a new project, default values are configured, but you must review the settings and ensure that they are correct. If you insert a CLIP into the socket but do not configure the **Clocking and IO** page, an error is returned.

Note The high-speed serial devices support empty sockets.

10. Select the **Clocking** tab.
11. Under **Input Clock Configuration**, select your input clock and its frequency in MHz. The input clock is sourced from one of the following locations:
    - Device backplane (PXIe_Clk100)
    - CLK IN connector on the front panel
    - 10 MHz Onboard Clock (PXIe-6592R only)
12. Under **Output Clock Configuration**, select any output reference clocks you want to use and specify their frequencies in MHz. The output clocks that you select here are routed to your CLIP for use with your application.

Note **(PXIe-6591R only)** When CLK IN/OUT is enabled as an output reference clock, it routes the specified frequency to the SMA connector.

Note **(PXIe-6592R only)** When PFI 0/CLK OUT, PFI 1, PFI 2, and PFI 3 are enabled as output reference clocks, they route the specified frequency to the corresponding SMB connector.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note If you specify an invalid combination of input clock frequencies and output clock frequencies, an error message appears and the **OK** button is dimmed until you reconfigure the clocks to a valid frequency combination.

13. Select the **IO Configuration** tab.
14. Under **Active Serial Lanes**, select the checkbox for any serial lanes that you want configured for Transmit (TX) or Receive (RX). Select the **All/None** checkbox to select/deselect all serial lanes.

Note Selecting the active serial lanes adds additional placement constraints to your project based on the lanes you select, so ensure that you select all lanes used in your project.

15. Under **GPIO Configuration**, use the **Voltage Family** selector box to specify the voltage level used by the GPIO (**Digital Data and Control** on the PXIe-6591R, and **PFI 0**, **PFI 1**, **PFI 2**, and **PFI 3** on the PXIe-6592R).

    Refer to Table 8-1 for a list of clocking and routing dependencies for the PXIe-6591R and PXIe-7902. Refer to Table 8-2 for a list of clocking and routing dependencies for the PXIe-6592R.
16. Click **OK**.

Table 8-1. PXIe-6591R and PXIe-7902 Clocking and Routing Dependencies

Connector	Valid Configurations
CLK IN/OUT	Input clock or output clock

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 8-2. PXIe-6592R Clocking and Routing Dependencies

Connector/Clock	Valid Configurations	Notes
PFI 0/CLK IN	Input clock or output clock	When enabled as output clocks, PFI 0/CLK IN/OUT, PFI 1/CLK OUT, PFI 2/CLK OUT, and PFI 3/CLK OUT must share the same frequency. The output of any enabled PFI*x* line is 10 MHz when Enable CPRI Output Clock Configuration is enabled. If PFI 0/CLK IN/OUT is not configured as the input clock or enabled as an output clock, you can configure PFI 0/CLK IN/OUT for GPIO.
PFI 1	Output clock	When enabled as output clocks, PFI 0/CLK IN/OUT, PFI 1/CLK OUT, PFI 2/CLK OUT, and PFI 3/CLK OUT must share the same frequency. The output of any enabled PFI*x* line is 10 MHz when **Enable CPRI Output Clock Configuration** is enabled. If PFI 1/CLK OUT, PFI 2/CLK OUT, and PFI 3/CLK OUT are not configured as output clocks, you can use them individually for GPIO.
PFI 2	Output clock	
PFI 3	Output clock	
MGT_RefClk0 and MGT_RefClk1	—	When **Enable CPRI Output Clock Configuration** is enabled, the internal reference clock is set to 153.6 MHz. This imposes frequency restrictions on MGT_RefClk0 and MGTRefClk1. The output of any PFI lines is 10 MHz, phase-aligned with the 153.6 MHz clock.
MGT_RefClk2	—	This clock's frequency is always 156.25 MHz.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following steps show an example of how to configure the PXIe-6592R for 140 MHz output on MGT_RefClk0 from a 100 MHz input clock on PFI0/CLK IN.

1. Right-click the IO socket and select **Properties»Clocking and IO**.
2. Under **Input Clock Configuration**, select **PFI0/CLK IN** and enter **100**.
3. Under **Output Clock Configuration**, select the checkbox next to **MGT_RefClk0** to enable it.
4. Enter **140** in the text box to the right of **MGT_RefClk0**.
5. Click **OK**.

Note By selecting an input clock and an output clock in this example, the MGT_RefClk0 is phase loop-locked to the incoming 100 MHz clock.

Refer to Chapter 2, *PXIe-6591R Hardware Architecture*, for more information about PXIe-6591R clocking capabilities.

Refer to Chapter 4, *PXIe-6592R Hardware Architecture*, for more information about PXIe-6592R clocking capabilities.

Refer to Chapter 6, *PXIe-7902 Hardware Architecture*, for more information about PXIe-7902 clocking capabilities.

Using Existing VHDL IP inside CLIP or IPIN

To use existing IP in your project, refer to the *Importing External IP Into LabVIEW FPGA* white paper at `ni.com`.

CLIP does not support custom user libraries in the VHDL. If your VHDL uses custom user libraries, use one of the following workarounds:

- Create a netlist from the VHDL and integrate the netlist using CLIP.
- Reference the default reference library instead of a custom user library.

Refer to the *Creating or Acquiring IP (FPGA Module)* topic in the *LabVIEW FPGA Module Help* for more information about using existing VHDL IP inside CLIP or IPIN.

Adding High-Speed Serial Device Target I/O

Complete the following steps to add target I/O for the high-speed serial device and to access signals from any instantiated CLIP on the block diagram:

1. Place an FPGA I/O node on the FPGA target block diagram. The FPGA I/O node is located on the palette under **Functions»FPGA I/O»FPGA I/O Node**.
2. Right-click the FPGA I/O node and select **Add New FPGA I/O**.
3. In the **New FPGA I/O** dialog box, select resources under **Available Resources** and add them to **New FPGA I/O** using the right arrow button.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4. To remove a resource, select the resource under **New FPGA I/O** and click the left arrow button.
5. Click **OK**.

Using the NI Common Instrument Design Libraries

Instrument design libraries can speed up your application development. The instrument design libraries are located at `<LVDir>\instr.lib\_niInstr`.

In LabVIEW, the common instrument design libraries are located on the niHighSpeedSerial Host palette at **FPGA Interface»Software-Defined Instruments»HSS Host**.

In LabVIEW FPGA, the common instrument design libraries are located on the **<Target Name>** palette.

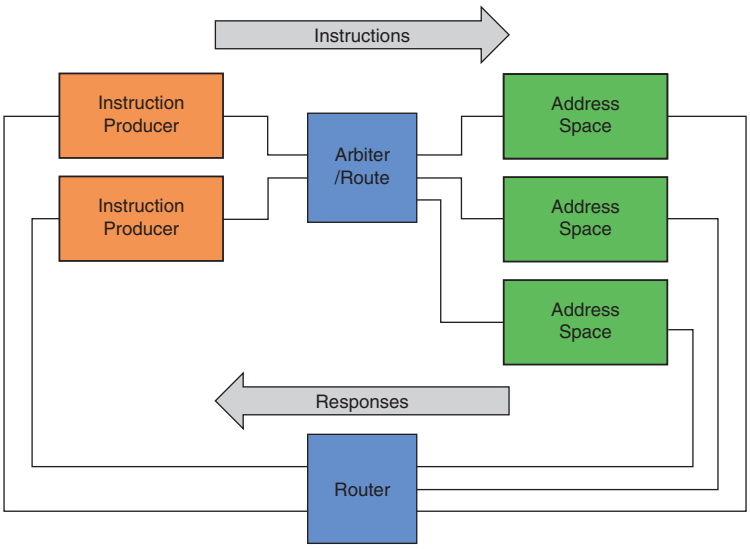
The following sections provide an overview of the instrument design libraries. For more information about the instrument design libraries, refer to the *NI High-Speed Serial Instruments Help*.

Using niInstr Instruction Framework

Use the Instruction Framework instrument design library to build a communication network in LabVIEW FPGA. The network has two types of endpoints: Instruction Producers and Address Spaces. Instruction Producers issue read and write instructions that are targeted for a particular Address Space. When the destination Address Space completes an instruction, it provides a response which is routed back to the Instruction Producer that issued the instruction, as shown in the following diagram.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Figure 8-4. Instruction Framework Overview



The Instruction Framework instrument design library includes some host VIs to help with communication between host libraries and FPGA address spaces. The Subsystem Map class provides a way to look up the location of an Address Space on the host, using the Address Space's unique identifier (UID). A context object is returned from this lookup function, which can be used to specify the destination for a read or write operation.

Use the Instruction Target class as an abstract interface for the mechanism used to communicate with the Instruction Framework FPGA Network. The FIFO Register Bus instrument design library provides an Instruction Target implementation on the host, which is coupled to an Instruction Producer implementation on the FPGA. This allows a host controller to send instructions to Address Spaces on the FPGA, and read the resulting responses. The instructions are sent to the FPGA using a uniquely named DMA FIFO, and the responses are received using a uniquely named indicator.

Some instrument design libraries use the Instruction Framework as a configuration mechanism. These libraries require a Subsystem Map object to look-up the location of corresponding Address Spaces on the FPGA. These libraries will also require an Instruction Target to provide the communication mechanism from the host controller to an Instruction Producer on the Instruction Framework FPGA Network. The corresponding Address Spaces must be added to the Instruction Framework FPGA Network in order for the host library to communicate properly with the FPGA library. This Address Space implementation provides register access through an Instruction Producer, such as the FIFO Register Bus library, instead of using controls and indicators on the top level of the FPGA diagram.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Using niInstr Streaming

The Streaming Instrument Design Library provides a consistent mechanism to handle both finite and continuous transfer streams. It provides stream monitoring and handshaking. It contains VIs for both the Host and FPGA.

Refer to the Aurora Simple Streaming sample project for an example of how to use the Streaming Instrument Design Library.

Using niInstr CLIP Adapters

The CLIP Adapters instrument design library includes AXI4-Lite and AXI4-Stream wrappers. Refer to the *Connecting AXI4-Lite and AXI4-Stream Interfaces to the Host* section of this chapter for more information about how to use the CLIP Adapters instrument design library.

Using niInstr Data Trigger

Use the Data Trigger Instrument Design Library VIs to generate a trigger when the input data sample is more than or less than the configured value.

This library supports multiple trigger types. For more information on supported triggers and their use, refer to the LabVIEW context help for the Data Trigger VIs.

This library supports various data types and samples per cycle.

Using niInstr Basic Elements

The Basic Elements Instrument Design Library contains VIs that perform common FPGA functions, such as detecting rising and falling edges, storing Boolean values, and data handshaking. Using this library can be beneficial when developing new FPGA logic for your instrument.

Using niInstr Eye Scan

Refer to the *Connecting Signals to Enable Eye Scan* and *Debugging Link Connections Using Eye Scan* sections of this chapter for information about using the Eye Scan instrument design library.

Connecting AXI4-Lite and AXI4-Stream Interfaces to the Host

In LabVIEW FPGA, the AXI4-Lite and AXI4-Stream wrappers are located on the `<Target Name>\CLIP Adapters` palette.

These wrappers adapt a collection of CLIP I/O that implements AXI4-Lite protocol and signaling into a simple reader or writer endpoints that present 4-wire handshaking to the diagram. This handshaking allows easier transition to many FPGA features without the need to implement the state logic on your own.

NI provides the following wrappers:

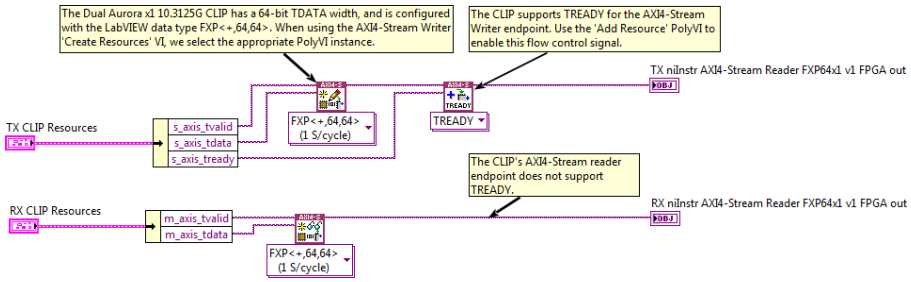
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

- AXI4-Lite FXP32x1
- AXI4-Stream Writer FXP32 (1 sample/cycle)
- AXI4-Stream Writer FXP64 (1 sample/cycle, 2 samples/cycle, and 4 samples/cycle)
- AXI4-Stream Reader FXP32 (1 sample/cycle)
- AXI4-Stream Reader FXP64 (1 sample/cycle, 2 samples/cycle, and 4 samples/cycle)

Refer to the Aurora sample projects for information about how to use the AXI4-Stream wrappers in an application.

The Aurora Simple Streaming sample project uses **Create AXI4-Stream Resources.vi** to register the CLIP Resources in the form of an AXI4-Stream endpoint. The following figure shows a snippet of the **Create AXI-4 Stream Resources.vi** block diagram.

Figure 8-5. Create AXI4 Resources VI Block Diagram



The resulting AXI4-Stream Writer and AXI4-Stream Reader can then be used to transfer data by calling **Write.vi** and **Read.vi**, respectively. Refer to the *NI High-Speed Serial Instruments Help* for more information about how to use high-speed serial VIs.

Refer to the Aurora sample projects for information about how to use the AXI4-Stream wrappers in an application.

Connecting Signals to Enable Eye Scan

You must connect the GTX2_CHANNEL primitive's registers to the Instruction Framework in order to use Eye Scan. Connect these registers using the FXP32-Address, FXP32-Data version of AXI4-Lite. For information about how to use the AXI4-Lite interface to configure Eye Scan, refer to the *Connecting AXI4-Lite and AXI4-Stream Interfaces to the Host* section of this chapter.

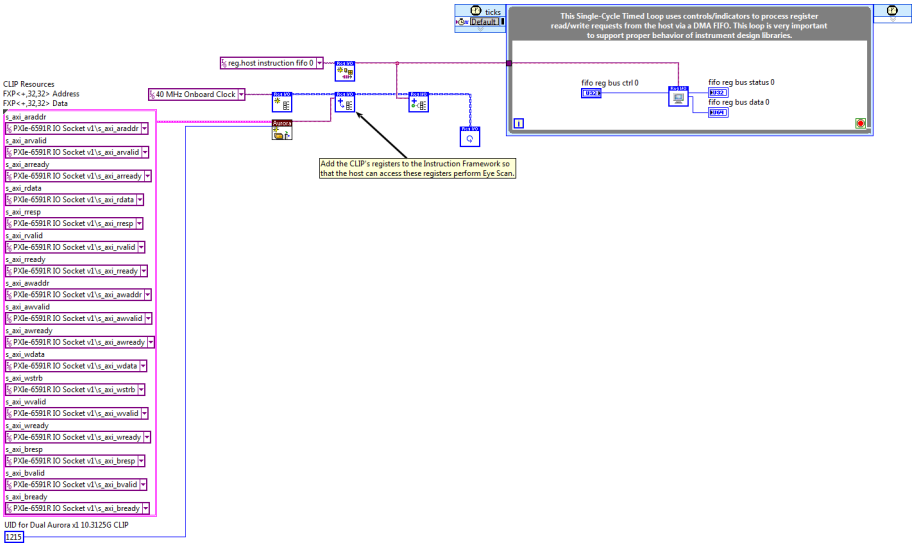
Note Eye Scan requires access to the GTX2_CHANNEL primitive's DRP signals. Refer to the Aurora sample project for an example of how to translate the AXI4-Lite signals to DRP signals within the CLIP.

The following figure demonstrates how to connect the appropriate CLIP resources to the Instruction Framework using the Create AXI4-Lite Resources VI. The Create AXI4-Lite Resources VI creates the AXI4-Lite CLIP Adapter and specifies the offsets for the DRP

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

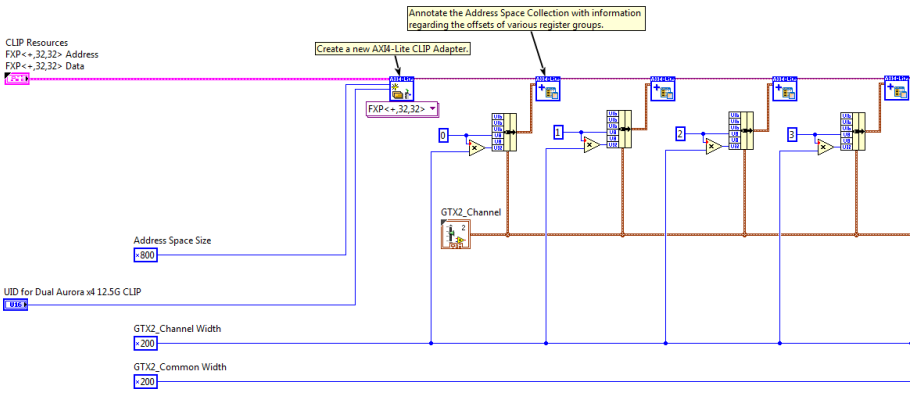
registers. The DRP subsystems in that VI are then registered with the Instruction Framework via the call to **Add Subsystems.vi**.

Figure 8-6. Connecting CLIP Resources to the Instruction Framework



The following figure shows **Create AXI4-Lite Resources.vi**, which is used by each Aurora sample project. This VI creates an AXI4-Lite CLIP Adapter object using the specified CLIP Resources. **Add Address Map Element.vi** registers the GTX2_CHANNEL at the appropriate offsets, so that they can be easily referenced on the host. It also registers the GTX2_COMMON registers at their appropriate offsets.

Figure 8-7. Create AXI4-Lite Resources.vi



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

For more information about how to use Eye Scan VIs, refer to the *NI High-Speed Serial Instruments Help*.

Compiling LabVIEW FPGA VIs

You may need to purchase and install additional licenses to compile FPGA designs that incorporate licensed cores from Xilinx or third-party IP vendors. Refer to *UG 973: Vivado Design Suite: Release Notes, Installation, and Licensing* at `xilinx.com` for information about managing licenses.

The high-speed serial devices include large FPGA devices that require a 64-bit compile worker. Refer to the *NI High-Speed Serial Readme* for more information about what platforms to use to compile bitfiles.

If you are using LabVIEW FPGA 2014 or later, you cannot add additional licenses to remote compile workers in the NI LabVIWEW FPGA Compile Cloud Service. You cannot use NI LabVIEW FPGA Compile Cloud Service to compile designs that incorporate Xilinx or other third-party licensed cores.

LabVIEW and System Integration

The following sections contain information about how to integrate your high-speed serial system in the LabVIEW application development environment.

Download, Reset, and Run Side Effects in the LabVIEW FPGA Host Interface

When the high-speed serial device FPGA loads, it performs a power-on self-configuration sequence that configures various on-board hardware. This configuration occurs at the following times:

- At device power-up after the bitfile loads.
- At the first time **Run** is called after a new bitfile is downloaded and the bitfile is not set to **Run on Load**.
- When **Run** is called after **Reset**.

For more information about **Run**, **Reset**, and other Invoke methods, refer to the *LabVIEW FPGA Module Help*.

Note When self-configuration executes, the clocking configuration enters an indeterminate state. When the clocking configuration is in an indeterminate state, you cannot rely on clocking stability from the clocking and routing hardware on the high-speed serial device.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

DMA Streaming

The high-speed serial devices support both host-to-target streaming and target-to-host streaming through DMA channels that connect the host to your target. Use DMA streaming to allow the maximum throughput of data from your host application to be streamed to the target at high rates of speed.

The high-speed serial devices provide up to 32 DMA channels that can be accessed by your Host. These channels can be used in a variety of ways to meet your application's needs. The total overall bandwidth of the module limits your DMA use, whether you use 1 DMA channel or 32.

The maximum width of a DMA channel is 256 bits. To use the full width of the DMA channel to achieve maximum throughput, you can write an array of U64 elements into the DMA FIFO and configure the FIFO for multi-element read/write (four elements per read/write) to satisfy the 256 bit width. You can also write up to 1024 bits at a time from LabVIEW FPGA, and the Ready for Input connection throttles the connection to the FIFO to prevent overflow.

Theoretically, DMA throughput is maximized and is most consistent when the DMA FIFO buffer is sized as large as possible to absorb variations in the readiness of the host memory. However, sizing the FIFO larger consumes block RAM resources on the FPGA and increases the timing pressure on the FIFO. NI recommends making the FIFO as large as you can successfully compile with, in order to sustain throughput through the PCIe bus to and from host memory. You can change the size of the FIFO by configuring the Requested Number of Elements for the FIFO in the project properties. You can validate the DMA sizing through benchmarking, and you can use VIs in the Streaming Design Library to monitor the health of a FIFO.

For more detailed information about using DMA, DMA best practices, and how to make design decisions on how to implement DMA in your application, refer to the *Transferring Data Using Direct Memory Access* topic of the *LabVIEW FPGA Help*.

Total throughput depends on the SCTL rate from the FPGA that is reading or writing the DMA channels. The data throughput is calculated by the following equation:

$$(\textit{Data Width} \times \textit{Samples per Cycle}) \times \textit{Number of DMA FIFOs} \times \textit{SCTL Clock Rate} = \textit{Data Throughput}$$



Note The total data throughput cannot exceed the maximum data specification for your device. Refer to the *Specifications* document for your device for information about data throughput limits.



Note Some remote controlling PCs and PXI Express chassis have slot bandwidth restrictions that may limit the maximum throughput of your application. Refer to the controller and chassis specifications for more information.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note The number of array elements fed into the DMA FIFO from the Host can limit the maximum throughput for your application. Use large array subsets and set your FIFO depths to be deep enough to sustain high throughput.

Peer-to-Peer Streaming

The peer-to-peer data streaming architecture is a method of transferring data between hardware devices. A peer-to-peer stream acts like a single, unidirectional pipe from which data can flow directly from one device to another. Using the peer-to-peer data streaming architecture, two or more devices can transfer data directly to each other without first going through the host processor.

The high-speed serial devices can stream Peer-to-Peer (P2P) data to or from any other NI-P2P capable device. For example, you can implement the writer on a digitizer using the NI-SCOPE instrument driver and implement the reader on an NI high-speed serial software-designed instrument device using LabVIEW with the LabVIEW FPGA Module.

For an overview of the P2P architecture, refer to *An Introduction to Peer-to-Peer Streaming* at `ni.com`. If you need more details on Peer-to-Peer and the P2P API, refer to the *NI Peer-to-Peer Streaming Help*, available at `ni.com/manuals` and as part of the *NI High-Speed Serial Instruments Help*.

Maximizing Peer-to-Peer Streaming Throughput

Maximum throughput is dependent on the streaming modules, chassis, and, if the configuration warrants it, the controller. Generally, the lowest of these rates is the maximum possible P2P bandwidth. High-speed serial devices are PXI Express x8 Gen 2 capable, meaning they allow theoretical 3.4 GB/s unidirectional streaming and theoretical 2.4 GB/s bidirectional streaming. The recommended hardware setup for getting maximum P2P streaming between NI high-speed serial instruments is to use an NI PXIe-1085 chassis with a device in slot 4 and a device in slot 6.

For example, a P2P steam that is four U64 samples wide running on the 100 MHz PXI clock can theoretically transfer data at 3.2 GB/s. When transferring data at this speed, use the handshaking interface on the P2P FIFO to implement flow control.

PXI Triggers

You can use the FPGA I/O Node to access the trigger lines on PXI devices. The following sections explain how to configure trigger pulses, reserve trigger lines, and release trigger lines.

Configuring Trigger Pulses

To ensure compatibility with other devices, configure trigger pulses on the high-speed serial device to last for at least two clock cycles of the clock on the receiving device. For example, if the clock on the receiving device is 80 MHz, which corresponds to a clock period of 12.5 nanoseconds, the trigger line must be constant for at least 25 ns, which is two cycles of an 80 MHz clock.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note Regardless of the clock speed, pulses on the trigger line must be constant for at least 18 nanoseconds.

The clocks between a high-speed serial device and another PXI device might not be perfectly synchronized. If you assert a trigger line on a high-speed serial device device, you cannot determine at what point in the clock period the trigger registers in the receiving flip-flop. If the trigger arrives during the setup or hold time of the receiving flip-flop, you cannot determine the state of the line for that clock period. Asserting the trigger pulse for two clock cycles ensures that at least one clock cycle on the receiving flip-flop registers as a rising edge and transfers as a trigger.

Reserving PXI Triggers

National Instruments recommends that you reserve the trigger lines used by PXI devices, including the high-speed serial device. If two PXI devices try to drive the same trigger line in different applications, or if the PXI devices are not programmed to work together, the application does not work, and in some cases, third-party PXI devices can be damaged. You can use Measurement & Automation Explorer (MAX) or the LabVIEW FPGA Host VI to reserve trigger lines.

Reserving Trigger Lines in MAX

If you download and run the FPGA VI interactively, configure the PXI triggers in MAX. MAX maintains the trigger reservation for the device even after you cycle power to the PXI chassis.

Reserving Trigger Lines in the LabVIEW FPGA Host VI

If you download and run the FPGA VI programmatically, reserve the trigger lines in the host VI. Use the Invoke Method function to reserve the trigger and to release the trigger reservation. LabVIEW releases the trigger reservation for the device automatically when you close the FPGA VI reference. You must run the host VI again to reserve the trigger.

Reserving Trigger Lines

Complete the following steps to reserve a trigger line for a PXI device.

1. Place the Open FPGA VI Reference function on the block diagram and configure it for the FPGA device and FPGA VI.
2. Place the Invoke Method function on the block diagram.
3. Wire the **FPGA VI Reference Out** output of the Open FPGA VI Reference function to the **FPGA VI Reference In** input of the Invoke Method function.
4. Wire the **error out** output of the FPGA VI Reference function to the **error in** input of the Invoke Method function.
5. Click the Invoke Method function and select Reserve PXI Trigger from the shortcut menu.
6. Right-click the **Trigger** input and select **Create»Constant**. An enum constant is created to help you select the trigger.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

To reserve multiple trigger lines, repeat steps 2 to 6 for each trigger line you want to reserve, wiring the **FPGA VI Reference Out** output of the existing Invoke Method function to the **FPGA VI Reference In** input of the Invoke Method node that follows it.

Releasing Trigger Lines

Complete the following steps to release a trigger line for your device.

Note LabVIEW automatically releases the trigger reservation when you close the FPGA VI Reference, but you can use an invoke node if you want to unreserve the trigger without closing the FPGA VI Reference.

1. Place the Open FPGA VI Reference function on the block diagram and configure it for the FPGA device and FPGA VI.
2. Place the Invoke Method function on the block diagram.
3. Wire the **FPGA VI Reference Out** output of the Open FPGA VI Reference function to the **FPGA VI Reference In** input of the Invoke Method function.
4. Wire the **error out** output of the FPGA VI Reference function to the **error in** input of the Invoke Method function.
5. Click the Invoke Method function and select **Unreserve PXI Trigger** from the shortcut menu.
6. Right-click the **Trigger** input and select **Create»Constant**. An enum constant is created to help you select the trigger.

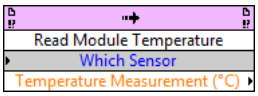
To release multiple trigger lines, repeat steps 2 to 6 for each trigger line you want to release, wiring the **FPGA VI Reference Out** output of the existing Invoke Method function to the **FPGA VI Reference In** input of the Invoke Method node that follows it.

Monitoring Power and Temperature

Due to the degree of customization possible with the high-speed serial device FPGAs, some applications may draw too much power or dissipate too much heat. Monitor the device state carefully, especially if your device pushes the power or heat limits. Use the Read Module Temperature and Read Module Power method nodes to monitor device temperature and heat, respectively.

Read Module Temperature allows you to read two onboard sensors: one sensor is embedded directly in the FPGA, and one reads the device temperature.

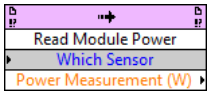
Figure 8-8. Read Module Temperature



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Read Module Power provides information about how much power the device is drawing from the chassis 3.3V and 12V power rails.

Figure 8-9. Read Module Power



Soft Shutdown

Exceeding the soft thermal and power threshold puts your device in a safe state and provides a warning. The errors `-63170` or `-63171` indicate that the device has gone into a soft shutdown. To recover from this error, power cycle the board. In order to avoid seeing this error again, improve the airflow to your chassis or consider reduced FPGA logic.

Power/Thermal Protection and Shutdown

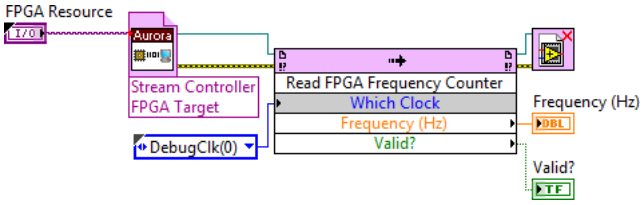
The high-speed serial devices have a fail-safe power and thermal shutdown to prevent damage to the module or other hardware in your chassis. Device shutdown can cause your host to reboot or display an error screen. In order to avoid this issue, monitor temperature and power consumption closely when developing custom FPGA images.

If a power and thermal shutdown occurs, power cycle the system and contact NI customer support at `ni.com/support`.

Debugging Clocks Using Frequency Counters

The Frequency Counter is a module inside the FPGA that provides a built-in method for debugging clock signals during CLIP development. To use the Frequency Counter, complete the following steps:

1. Attach clocks to one of the four **DebugClk(3..0)** signals in the CLIP.
2. On the host, use the method provided on all high-speed serial device FPGA reference wires.



The Read FPGA Frequency Counter method initiates a Frequency Counter measurement for 1 ms and reports the frequency of the selected clock. If the Frequency Counter cannot detect a clock signal, it returns **Valid?** as false. Valid frequencies are reliable between 2 kHz and 333.33 MHz.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Debugging Link Connections Using Eye Scan

The MGTs on the Xilinx Kintex-7 provide a debugging capability called Eye Scan. The Eye Scan API VIs are located on the LabVIEW VI palette at **FPGA Interface»Software-Designed Instruments»NI High-Speed Serial»NI Eye Scan**.

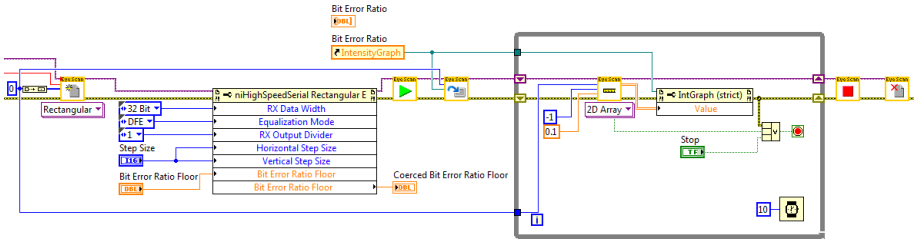
You must connect the GTX2_CHANNEL register(s) on the FPGA to use Eye Scan. Refer to the *Wiring the Instruction Targets* section for information about connecting the GTX2_CHANNEL register. Refer to the Aurora sample project to learn how to use the Eye Scan API in an application.

National Instruments offers two versions of Eye Scan: Rectangular Eye Scan and N Point Eye Scan. Use Rectangular Eye Scan to obtain a traditional eye that sweeps the unit interval and nominal voltage. Use N Point Eye Scan to measure the Bit Error Ratio at arbitrary unit interval and nominal voltage offsets. For information about the Rectangular Eye Scan VIs and N Point Eye Scan VIs, refer to the *NI High-Speed Serial Instruments Help*.

Rectangular Eye Scan

The following portion of code shows a typical use case for Rectangular Eye Scan. This code measures a single channel and updates the Bit Error Ratio graph once per 100 milliseconds.

Figure 8-10. Rectangular Eye Scan



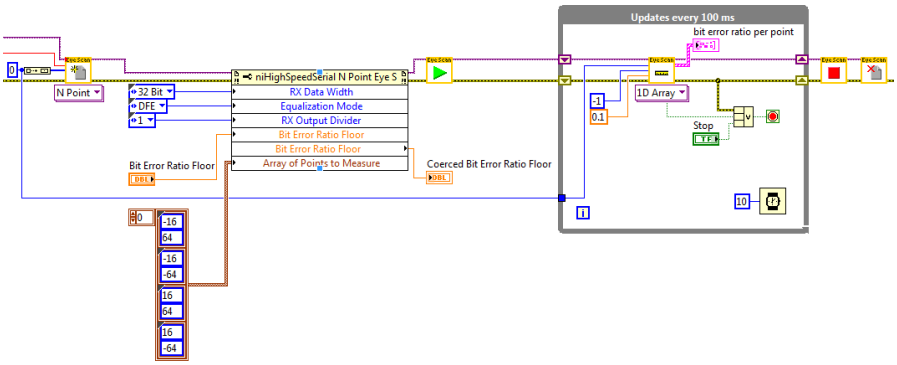
Use the **Horizontal Step Size** and **Vertical Step Size** properties to control the density of your Eye Scan plot. The points to be measured are computed based on step size starting from 0, up to the largest absolute value closest to, but not exceeding, the maximum value.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

N Point Eye Scan

The following portion of code shows a typical use case for N Point Eye Scan. This code scans 4 points with a Bit Error Ratio floor of 2.33E-10 and produces a 4-point eye, which is useful for measuring pass/fail conditions. To obtain a 6-point eye, add two extra points to the array using the **Array of Points to Measure** property.

Figure 8-11. N Point Eye Scan



The **Array of Points to Measure** property contains the horizontal and vertical offsets for each point that you wish to perform a Bit Error Ratio measurement on.

Wiring the Instruction Targets

Complete the following steps to wire the instruction targets shown in Figures 8-10 and 8-11.

Note Eye Scan uses the GTX2_CHANNEL UID with UID 1206 as a child context of the top-level UID. In the high-speed serial sample projects, the GTX2_CHANNEL UID is nested under the top-level UID. If you use a different child context UID for GTX2_CHANNEL, modify the **niHighSpeedSerial Eye Scan Single Point v1 Host.lvclass:Open Session.vi** to use your GTX2_CHANNEL's UID. Since Eye Scan is installed as a shared component, doing this will affect all uses of Eye Scan from your host computer. NI recommends that you keep the UID of GTX2_CHANNEL at 1206. If you use a different UID for GTX2_CHANNEL, create a local copy of the Eye Scan libraries in your LabVIEW project.

1. Create a FIFO Register Bus.
   a. Place **niInstr FIFO Register Bus v1 Host.lvclass:Open Session.vi** on the block diagram.
   b. Wire the **fpga ref** input parameter to the FPGA Reference obtained from **Open FPGA VI Reference**.
2. Retrieve the list of subsystems by connecting the **session out** parameter from **niInstr FIFO Register Bus v1 Host.lvclass:Open Session.vi** to the input of **niInstr Subsystem Map v1**

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Host.lvclass:Read Subsystem Map.vi. This retrieves information related to all subsystems, address spaces, and the routing context for each address space.

3. Connect the **subsystem map** output to **niInstr Subsystem Map v1 Host.lvclass: Subsystem Lookup.vi**. Specify the UID (Unique Identifier) for the top-level UID that has GTX2_CHANNEL as a child UID. By default, this is the same UID as that of the NI-provided CLIP. If you registered Eye Scan to a different top-level UID, use that UID instead.
4. Connect the **instruction target** from **niInstr Subsystem Map v1 Host.lvclass:Read Subsystem Map.vi** to the instruction target input terminal of **niHighSpeedSerialEyeScan v1 Host.lvclass:Open Session.vi**, and the **context** from **niInstr Subsystem Map v1 Host.lvclass:Subsystem Lookup.vi** to the parent context terminal of **niHighSpeedSerialEyeScan v1 Host.lvclass:Open Session.vi**.

Eye Scan State Model

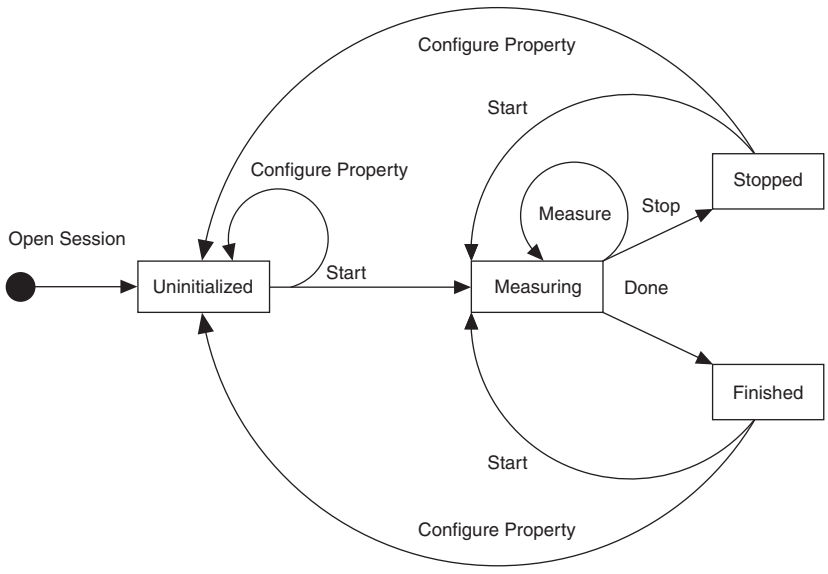
The Eye Scan API abstracts the Eye Scan algorithm provided by Xilinx into a simple API with few states. The general programming flow is as follows:

Open Session»Configure Properties»Start»Measure»Close Session.

Note You can loop measurements for frequent updates.

The following figure illustrates the Eye Scan state model.

Figure 8-12. Eye Scan State Model



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Complete the following steps to program the Eye Scan state model.

1. Open a session with **Open Session (Poly).vi**.
2. Configure the properties using the Property Node, located on the NI Eye Scan VI palette (**FPGA Interface»Software-Designed Instruments»NI High-Speed Serial»NI Eye Scan**). Some properties have default values, but you must configure the following properties before starting the scan:
   - RX Internal Data Width
     - Valid values are 16 bits, 20 bits, 32 bits, and 40 bits.



Note The RX Internal Data Width property must be set to the Internal Data Width of the RX interface datapath. Refer to the FPGA RX Interface section of the Xilinx *7 Series FPGAs GTX/GTH Transceivers User Guide* for information about the FPGA RX interface datapath configuration.

   - RX Output Divider
   - Equalization Mode
     - Valid modes are LPM (low-power mode) and DFE (decision-feedback equalization). Refer to the Xilinx *7 Series FPGAs GTX/GTH Transceivers User Guide* for more information about these modes.
3. Call **Start.vi** to begin measuring on the first point.
4. NI recommends calling **Measure (Poly).vi** with a relatively small timeout value in a loop. This allows you to receive updates to the progress of Eye Scan while measurements are still being taken, and provides updates without using a large amount of computational resources. Once Eye Scan finishes measuring the active point, it automatically reconfigures for the next point and begins measuring that point. This process continues until one of the following occurs:
   - *Number of New Points Requested* completes
   - *Timeout* is reached
   - Every requested point is scanned
5. Once the last point has been measured, Eye Scan moves to the Finished state. In the Finished state, you can call the Measure VI to retrieve all previously acquired data since the last Start call. You can also reconfigure and call Start again in order to begin a new Eye Scan.



Note When you call Start, all previously acquired data is discarded.

To stop Eye Scan before measurement is complete, call **Stop.vi**. Calling this VI moves Eye Scan to the Stopped state and discards the data from the point currently being measured. Previously acquired points may no longer be obtained by calling **Measure (Poly).vi**. Once Eye Scan is stopped, you can reconfigure properties and call **Start.vi** to restart measurement.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A

Troubleshooting

For information about how to generate and integrate Aurora IP into your LabVIEW project, refer to Knowledge Base article 6R6EOLM3.

For information about troubleshooting problems adding your high-speed serial instrument to your Real Time system, refer to Knowledge Base article 7AEBF43J.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B

Xilinx Documentation References

Xilinx FPGA documentation provides information required for the successful development of your high-speed serial device. The following table provides a list of specific Xilinx documentation resources.

All Xilinx documentation can be found at `www.xilinx.com`.

Table B-1. Xilinx 7-Series FPGA Documentation

Document	Document Part Number	Description
7 Series FPGAs Overview	DS180	Outlines the features and product selection of the Xilinx 7 series FPGAs, including Kintex-7 devices.
Kintex-7 FPGAs Data Sheet: DC and AC Switching Characteristics	DS182	Contains the DC and AC switching characteristic specifications for the Kintex-7 FPGAs.
Vivado Design Suite: Release Notes, Installation, and Licensing	UG973	Provides an overview of the new release of the Vivado Design Suite, including information on new and changed features, installation requirements for the software, and licensing information.
High-Speed Serial I/O Made Simple: A Designer's Guide, with FPGA Applications	—	Recommended for users new to high-speed serial.
7 Series FPGAs GTX/GTH Transceivers User Guide	UG476	Technical reference describing the 7 series FPGAs GTX/GTH transceivers.
Vivado Design Suite User Guide: Using Constraints	UG903	Describes using Xilinx Design Constraints (XDC) in Vivado tools.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C

NI Services

National Instruments provides global services and support as part of our commitment to your success. Take advantage of product services in addition to training and certification programs that meet your needs during each phase of the application life cycle; from planning and development through deployment and ongoing maintenance.

To get started, register your product at ni.com/myproducts.

As a registered NI product user, you are entitled to the following benefits:

- Access to applicable product services.
- Easier product management with an online account.
- Receive critical part notifications, software updates, and service expirations.

Log in to your National Instruments ni.com User Profile to get personalized access to your services.

Services and Resources

- **Maintenance and Hardware Services**—NI helps you identify your systems' accuracy and reliability requirements and provides warranty, sparing, and calibration services to help you maintain accuracy and minimize downtime over the life of your system. Visit ni.com/services for more information**.**
  - **Warranty and Repair**—All NI hardware features a one-year standard warranty that is extendable up to five years. NI offers repair services performed in a timely manner by highly trained factory technicians using only original parts at a National Instruments service center.
  - **Calibration**—Through regular calibration, you can quantify and improve the measurement performance of an instrument. NI provides state-of-the-art calibration services. If your product supports calibration, you can obtain the calibration certificate for your product at ni.com/calibration.
- **System Integration**—If you have time constraints, limited in-house technical resources, or other project challenges, National Instruments Alliance Partner members can help. To learn more, call your local NI office or visit ni.com/alliance.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

- **Training and Certification**—The NI training and certification program is the most effective way to increase application development proficiency and productivity. Visit ni.com/training for more information.
  - The Skills Guide assists you in identifying the proficiency requirements of your current application and gives you options for obtaining those skills consistent with your time and budget constraints and personal learning preferences. Visit ni.com/skills-guide to see these custom paths.
  - NI offers courses in several languages and formats including instructor-led classes at facilities worldwide, courses on-site at your facility, and online courses to serve your individual needs.
- **Technical Support**—Support at ni.com/support includes the following resources:
  - **Self-Help Technical Resources**—Visit ni.com/support for software drivers and updates, a searchable KnowledgeBase, product manuals, step-by-step troubleshooting wizards, thousands of example programs, tutorials, application notes, instrument drivers, and so on. Registered users also receive access to the NI Discussion Forums at ni.com/forums. NI Applications Engineers make sure every question submitted online receives an answer.
  - **Software Support Service Membership**—The Standard Service Program (SSP) is a renewable one-year subscription included with almost every NI software product, including NI Developer Suite. This program entitles members to direct access to NI Applications Engineers through phone and email for one-to-one technical support, as well as exclusive access to online training modules at ni.com/self-paced-training. NI also offers flexible extended contract options that guarantee your SSP benefits are available without interruption for as long as you need them. Visit ni.com/ssp for more information.
- **Declaration of Conformity (DoC)**—A DoC is our claim of compliance with the Council of the European Communities using the manufacturer's declaration of conformity. This system affords the user protection for electromagnetic compatibility (EMC) and product safety. You can obtain the DoC for your product by visiting ni.com/certification.

For information about other technical support options in your area, visit ni.com/services, or contact your local office at ni.com/contact.

You also can visit the Worldwide Offices section of ni.com/niglobal to access the branch office websites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Glossary

C

CLIP	Component-level intellectual property

D

DDR3	Double data rate. This term usually refers to the communication mechanism used to read and write DRAM.
DRAM	Dynamic random-access memory

F

FPGA	Field-programmble gate array

G

GPIO	General-purpose input/output

H

HSS	High-speed serial

I

IC	Integrated circuit
IP	Intellectual property (VHDL/Verilog/Netlist)

L

LOS	Loss of signal
LVFPGA	LabVIEW FPGA

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

M

MGT	Multi-gigabit transceiver

P

PFI	Programmable function interface
POSC	Power on self-configuration

S

SCTL	Single cycle timed loop
SFP+	Enhanced small form-factor pluggable

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP