PXIe-6569 Getting Started Guide

2025-03-24

Contents

Getting Started Guide

Note Before you begin, install and configure your chassis and controller.

This document explains how to install, configure, test, and use the PXIe-6569. You can program the PXIe-6569 with the following software options.

- NI-FlexRIO driver software
- NI LabVIEW Instrument Design Libraries for FlexRIO (instrument design libraries)

Note Adapter modules are not installable or interchangeable on the PXIe-6569 device.

The PXIe-6569 is available in the following fixed LVDS configurations:

- PXIe-6569 with 32 LVDS In, 32 LVDS Out
- PXIe-6569 with 64 LVDS In
- PXIe-6569 with 64 LVDS Out

Note In this document, all variants are referred to inclusively as the PXIe-6569.

FlexRIO Documentation and Resources

Use the following resources to find more information about the PXIe-6569.

All documentation can be found at ni.com/manuals or in LabVIEW by clicking **Help**.

Table 1. FlexRIO Documentation and Resources

Document	Contents
PXIe-6569 Getting Started Guide (this document)	• Installation instructions • Basic programming information
PXIe-6569 Specifications	• Operating environment requirements • DIO specifications • Clocking specifications • Physical and mechanical specifications
PXIe-6569 Safety, Environmental, and Regulatory Information	• Safety and compliance information • Environmental information
LabVIEW FPGA Module Help	• Basic functionality of the FPGA module • Instructions for developing and debugging custom hardware logic
FlexRIO 21.7 Readme	• Minimum system requirements • Supported Application Development Environments (ADEs) • Known issues and bug fixes • Recent updates
FlexRIO Help	• FlexRIO driver API and programming information • I/O Component Level IP (CLIP) development information
LabVIEW Examples	• Examples showing how to run FPGA VIs on your device • Examples showing how to run host VIs on your device

Unpacking the Kit

Notice To prevent electrostatic discharge (ESD) from damaging the module, ground yourself using a grounding strap or by holding a grounded object, such as your computer chassis.

1. Touch the antistatic package to a metal part of the computer chassis.
2. Remove the module from the package and inspect it for loose components or other signs of damage.

   Notice Never touch the exposed pins of connectors.

   Note Do not install a module if it appears damaged in any way.
3. Unpack any other items and documentation from the kit.

Store the module in the antistatic package when the module is not in use.

What You Need to Get Started

Kit Contents

Verify that the following items are included in the PXIe-6569 kit.

- PXIe-6569 hardware
- Documentation
  - ***PXIe-6569 Getting Started Guide*** (this document)
  - ***PXIe-6569 Safety, Environmental, and Regulatory Information***

Recommended Cables

NI offers two lengths of optional SEARAY™ to SEARAY cables for connecting to the pins on the front panel. The following table shows the details for each.

Model Name	Part Number
SR240M-SR240M Cable, LVDS with SE, 0.5m	787317-0R5
SR240M-SR240M Cable, LVDS with SE, 1.0m	787317-01

Installing the Software

You must be an Administrator to install NI software on your computer.

1. Install an ADE, such as LabVIEW or LabWindows™/CVI™.
2. Download the driver software installer from ni.com/downloads.
   NI Package Manager downloads with the driver software to handle the installation. Refer to the NI Package Manager Manual for more information about installing, removing, and upgrading NI software using NI Package Manager.
3. Follow the instructions in the installation prompts.

   Note Windows users may see access and security messages during installation. Accept the prompts to complete the installation.

4. When the installer completes, select **Restart** in the dialog box that prompts you to restart, shut down, or restart later.

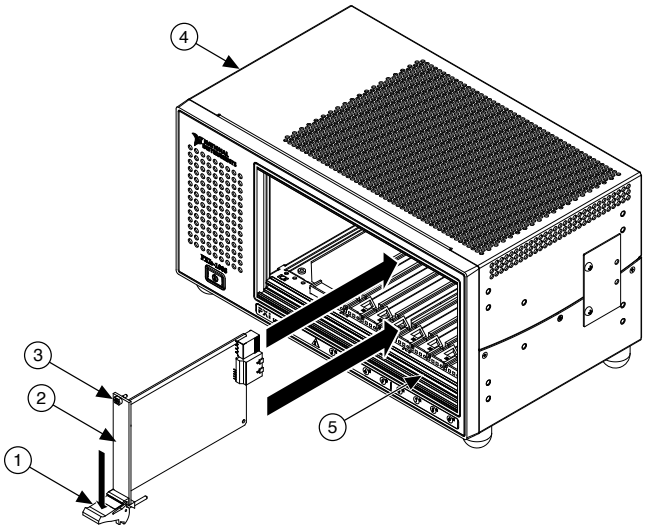
Installing FlexRIO Modules

This section contains general installation instructions for installing a FlexRIO module in a PXI Express chassis. Refer to your chassis user manual for specific instructions and warnings. To install a module, complete the following steps:

1. Connect the AC power source to the PXI Express chassis before installing the module. The AC power cord grounds the chassis and protects it from electrical damage while you install the module.
2. Ensure that the chassis is powered off.
3. Install a module into a chassis slot by first placing the module card PCB into the front of the card guides (top and bottom), as shown in the following figure. Slide the module to the rear of the chassis, making sure that the injector/ejector handle is pushed down, as shown in the following figure.
4. When you begin to feel resistance, push up on the injector/ejector handle to fully

seat the module into the chassis frame. Secure the module front panel to the chassis using the module front-panel mounting screws.

Figure 1. Installing PXI Express Modules

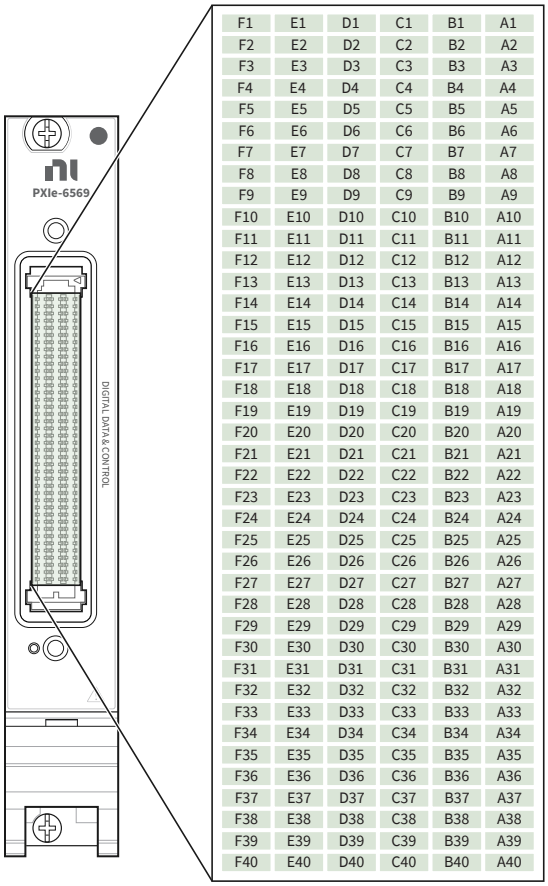


1. Injector/Ejector Handle
2. FlexRIO Module
3. Front Panel Mounting Screws (2x)
4. PXI Express Chassis
5. Injector/Ejector Rail

PXIe-6569 Front Panel and Connectors

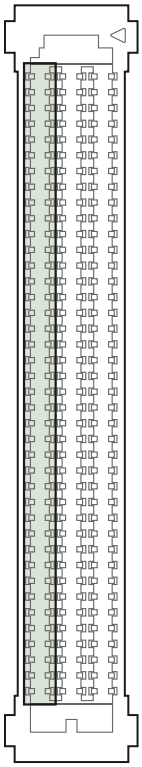
The following figure shows the front panel and pin layout of the Digital Data & Control (DDC) connector on the PXIe-6569.

Figure 2. PXIe-6569 Front Panel Layout



The following figures show the pinout of the DDC connector on the PXIe-6569 for each connector type. Clock-capable pins are denoted in bold.

Figure 3. PXIe-6569 with 32 LVDS In, 32 LVDS Out, Rows F-E

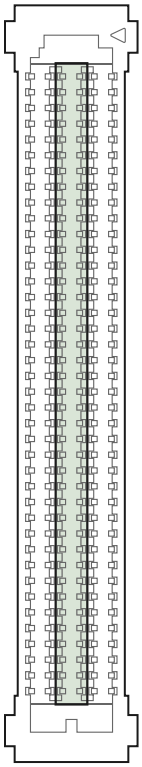


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	F1	E1	GND	
aSeGpio(1)	SE 0	F2	E2	SE 1	aSeGpio(3)
	SE_GND_TERM	F3	E3	SE_GND_TERM	
	GND	F4	E4	GND	
To clocking	CLK IN+	F5	E5	**DI 10+***	**aDiffGpio_p(58)***
circuitry	CLK IN-	F6	E6	**DI 10-***	**aDiffGpio_n(58)***
	GND	F7	E7	GND	
aDiffGpio_p(46)	DI 0+	F8	E8	DI 11+	aDiffGpio_p(50)
aDiffGpio_n(46)	DI 0-	F9	E9	DI 11-	aDiffGpio_n(50)
	GND	F10	E10	GND	
aDiffGpio_p(48)	DI 1+	F11	E11	DI 12+	aDiffGpio_p(49)
aDiffGpio_n(48)	DI 1-	F12	E12	DI 12-	aDiffGpio_n(49)
	GND	F13	E13	GND	
aDiffGpio_p(47)	DI 2+	F14	E14	**DI 13+***	**aDiffGpio_p(53)***
aDiffGpio_n(47)	DI 2-	F15	E15	**DI 13-***	**aDiffGpio_n(53)***
	GND	F16	E16	GND	
aDiffGpio_p(51)	DI 3+	F17	E17	DI 14+	aDiffGpio_p(67)
aDiffGpio_n(51)	DI 3-	F18	E18	DI 14-	aDiffGpio_n(67)
	GND	F19	E19	GND	
aDiffGpio_p(69)	DI 4+	F20	E20	DI 15+	aDiffGpio_p(65)
aDiffGpio_n(69)	DI 4-	F21	E21	DI 15-	aDiffGpio_n(65)
	GND	F22	E22	GND	
aDiffGpio_p(63)	DI 5+	F23	E23	DI 16+	aDiffGpio_p(59)
aDiffGpio_n(63)	DI 5-	F24	E24	DI 16-	aDiffGpio_n(59)
	GND	F25	E25	GND	
aDiffGpio_p(56)*	**DI 6+***	F26	E26	**DI 17+***	**aDiffGpio_p(57)***
aDiffGpio_n(56)*	**DI 6-***	F27	E27	**DI 17-***	**aDiffGpio_n(57)***
	GND	F28	E28	GND	
aDiffGpio_p(61)	DI 7+	F29	E29	DI 18+	aDiffGpio_p(60)
aDiffGpio_n(61)	DI 7-	F30	E30	DI 18-	aDiffGpio_n(60)
	GND	F31	E31	GND	
aDiffGpio_p(64)	DI 8+	F32	E32	DI 19+	aDiffGpio_p(62)
aDiffGpio_n(64)	DI 8-	F33	E33	DI 19-	aDiffGpio_n(62)
	GND	F34	E34	GND	
aDiffGpio_p(66)	DI 9+	F35	E35	DI 20+	aDiffGpio_p(68)
aDiffGpio_n(66)	DI 9-	F36	E36	DI 20-	aDiffGpio_n(68)
	GND	F37	E37	GND	
aDiffGpio_p(55)	PFI 0+	F38	E38	RSVD	
aDiffGpio_n(55)	PFI 0-	F39	E39	RSVD	
	GND	F40	E40	GND	

Bank 46

Bank 46

Figure 4. PXIe-6569 with 32 LVDS In, 32 LVDS Out, Rows D-C

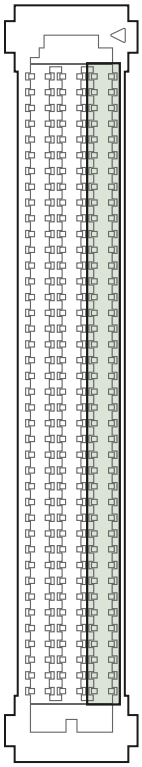


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	D1	C1	GND	
aSeGpio(5)	SE 2	D2	C2	SE 6	aSeGpio(13)
	SE_GND_TERM	D3	C3	SE_GND_TERM	
	GND	D4	C4	GND	
aSeGpio(7)	SE 3	D5	C5	SE 7	aSeGpio(15)
	SE_GND_TERM	D6	C6	SE_GND_TERM	
	GND	D7	C7	GND	
aDiffGpio_p(0)	DI 21+	D8	C8	DO 21+	aDiffGpio_p(4)
aDiffGpio_n(0)	DI 21-	D9	C9	DO 21-	aDiffGpio_n(4)
	GND	D10	C10	GND	
aDiffGpio_p(1)	DI 22+	D11	C11	DO 22+	aDiffGpio_p(8)
aDiffGpio_n(1)	DI 22-	D12	C12	DO 22-	aDiffGpio_n(8)
	GND	D13	C13	GND	
aDiffGpio_p(3)	DI 23+	D14	C14	DO 23+	aDiffGpio_p(12)
aDiffGpio_n(3)	DI 23-	D15	C15	DO 23-	aDiffGpio_n(12)
	GND	D16	C16	GND	
aDiffGpio_p(5)	DI 24+	D17	C17	DO 24+	aDiffGpio_p(11)
aDiffGpio_n(5)	DI 24-	D18	C18	DO 24-	aDiffGpio_n(11)
	GND	D19	C19	GND	
aDiffGpio_p(9)	DI 25+	D20	C20	DO 25+	aDiffGpio_p(16)
aDiffGpio_n(9)	DI 25-	D21	C21	DO 25-	aDiffGpio_n(16)
	GND	D22	C22	GND	
aDiffGpio_p(7)	DI 26+	D23	C23	DO 26+	aDiffGpio_p(14)
aDiffGpio_n(7)	DI 26-	D24	C24	DO 26-	aDiffGpio_n(14)
	GND	D25	C25	GND	
aDiffGpio_p(2)	DI 27+	D26	C26	DO 27+	aDiffGpio_p(19)
aDiffGpio_n(2)	DI 27-	D27	C27	DO 27-	aDiffGpio_n(19)
	GND	D28	C28	GND	
aDiffGpio_p(10)	DI 28+	D29	C29	DO 28+	aDiffGpio_p(17)
aDiffGpio_n(10)	DI 28-	D30	C30	DO 28-	aDiffGpio_n(17)
	GND	D31	C31	GND	
aDiffGpio_p(15)*	**DI 29+***	D32	C32	DO 29+	aDiffGpio_p(21)
aDiffGpio_n(15)*	**DI 29-***	D33	C33	DO 29-	aDiffGpio_n(21)
	GND	D34	C34	GND	
aDiffGpio_p(13)	DI 30+	D35	C35	DO 30+	aDiffGpio_p(20)
aDiffGpio_n(13)	DI 30-	D36	C36	DO 30-	aDiffGpio_n(20)
	GND	D37	C37	GND	
aDiffGpio_p(6)	DI 31+	D38	C38	DO 31+	aDiffGpio_p(18)
aDiffGpio_n(6)	DI 31-	D39	C39	DO 31-	aDiffGpio_n(18)
	GND	D40	C40	GND	

Bank 45

Bank 45

Figure 5. PXIe-6569 with 32 LVDS In, 32 LVDS Out, Rows B-A

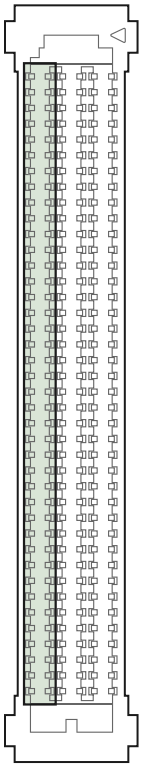


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	B1	A1	GND	
aSeGpio(11)	SE 5	B2	A2	SE 4	aSeGpio(9)
	SE_GND_TERM	B3	A3	SE_GND_TERM	
	GND	B4	A4	GND	
aDiffGpio_p(22)	DO 10+	B5	A5	CLK OUT+	From clocking
aDiffGpio_n(22)	DO 10-	B6	A6	CLK OUT-	circuitry
	GND	B7	A7	GND	
aDiffGpio_p(26)	DO 11+	B8	A8	DO 0+	aDiffGpio_p(28)
aDiffGpio_n(26)	DO 11-	B9	A9	DO 0-	aDiffGpio_n(28)
	GND	B10	A10	GND	
aDiffGpio_p(24)	DO 12+	B11	A11	DO 1+	aDiffGpio_p(23)
aDiffGpio_n(24)	DO 12-	B12	A12	DO 1-	aDiffGpio_n(23)
	GND	B13	A13	GND	
aDiffGpio_p(25)	DO 13+	B14	A14	DO 2+	aDiffGpio_p(29)
aDiffGpio_n(25)	DO 13-	B15	A15	DO 2-	aDiffGpio_n(29)
	GND	B16	A16	GND	
aDiffGpio_p(27)	DO 14+	B17	A17	DO 3+	aDiffGpio_p(33)
aDiffGpio_n(27)	DO 14-	B18	A18	DO 3-	aDiffGpio_n(33)
	GND	B19	A19	GND	
aDiffGpio_p(31)	DO 15+	B20	A20	DO 4+	aDiffGpio_p(41)
aDiffGpio_n(31)	DO 15-	B21	A21	DO 4-	aDiffGpio_n(41)
	GND	B22	A22	GND	
aDiffGpio_p(32)	DO 16+	B23	A23	DO 5+	aDiffGpio_p(38)
aDiffGpio_n(32)	DO 16-	B24	A24	DO 5-	aDiffGpio_n(38)
	GND	B25	A25	GND	
aDiffGpio_p(40)	DO 17+	B26	A26	DO 6+	aDiffGpio_p(34)
aDiffGpio_n(40)	DO 17-	B27	A27	DO 6-	aDiffGpio_n(34)
	GND	B28	A28	GND	
aDiffGpio_p(36)	DO 18+	B29	A29	DO 7+	aDiffGpio_p(42)
aDiffGpio_n(36)	DO 18-	B30	A30	DO 7-	aDiffGpio_n(42)
	GND	B31	A31	GND	
aDiffGpio_p(45)	DO 19+	B32	A32	DO 8+	aDiffGpio_p(30)
aDiffGpio_n(45)	DO 19-	B33	A33	DO 8-	aDiffGpio_n(30)
	GND	B34	A34	GND	
aDiffGpio_p(39)	DO 20+	B35	A35	DO 9+	aDiffGpio_p(43)
aDiffGpio_n(39)	DO 20-	B36	A36	DO 9-	aDiffGpio_n(43)
	GND	B37	A37	GND	
	RSVD	B38	A38	PFI 1+	aDiffGpio_p(52)
	RSVD	B39	A39	PFI 1-	aDiffGpio_n(52)
	GND	B40	A40	GND	

Bank 44

Bank 44

Figure 6. PXIe-6569 with 64 LVDS In, Rows F-E

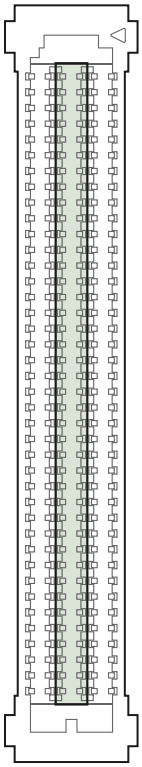


	FPGA Signal	Connector Signal			Connector Signal	FPGA Signal	
		GND	F1	E1	GND		
	aSeGpio(1)	SE 0	F2	E2	SE 1	aSeGpio(3)	
		SE_GND_TERM	F3	E3	SE_GND_TERM		
		GND	F4	E4	GND		
	To clocking	CLK IN+	F5	E5	**DI 43+***	**aDiffGpio_p(58)***	
	circuitry	CLK IN-	F6	E6	**DI 43-***	**aDiffGpio_n(58)***	
		GND	F7	E7	GND		
	aDiffGpio_p(46)	DI 54+	F8	E8	DI 44+	aDiffGpio_p(50)	
	aDiffGpio_n(46)	DI 54-	F9	E9	DI 44-	aDiffGpio_n(50)	
		GND	F10	E10	GND		
	aDiffGpio_p(48)	DI 55+	F11	E11	DI 45+	aDiffGpio_p(49)	
	aDiffGpio_n(48)	DI 55-	F12	E12	DI 45-	aDiffGpio_n(49)	
		GND	F13	E13	GND		
	aDiffGpio_p(47)	DI 56+	F14	E14	**DI 46+***	**aDiffGpio_p(53)***	
	aDiffGpio_n(47)	DI 56-	F15	E15	**DI 46-***	**aDiffGpio_n(53)***	
		GND	F16	E16	GND		
	aDiffGpio_p(51)	DI 57+	F17	E17	DI 47+	aDiffGpio_p(67)	
	aDiffGpio_n(51)	DI 57-	F18	E18	DI 47-	aDiffGpio_n(67)	
		GND	F19	E19	GND		
	aDiffGpio_p(69)	DI 58+	F20	E20	DI 48+	aDiffGpio_p(65)	
	aDiffGpio_n(69)	DI 58-	F21	E21	DI 48-	aDiffGpio_n(65)	
		GND	F22	E22	GND		
	aDiffGpio_p(63)	DI 59+	F23	E23	DI 49+	aDiffGpio_p(59)	
	aDiffGpio_n(63)	DI 59-	F24	E24	DI 49-	aDiffGpio_n(59)	
		GND	F25	E25	GND		
	aDiffGpio_p(56)*	**DI 60+***	F26	E26	**DI 50+***	**aDiffGpio_p(57)***	
	aDiffGpio_n(56)*	**DI 60- ***	F27	E27	**DI 50-***	**aDiffGpio_n(57)***	
		GND	F28	E28	GND		
	aDiffGpio_p(61)	DI 61+	F29	E29	DI 51+	aDiffGpio_p(60)	
	aDiffGpio_n(61)	DI 61-	F30	E30	DI 51-	aDiffGpio_n(60)	
		GND	F31	E31	GND		
	aDiffGpio_p(64)	DI 62+	F32	E32	DI 52+	aDiffGpio_p(62)	
	aDiffGpio_n(64)	DI 62-	F33	E33	DI 52-	aDiffGpio_n(62)	
		GND	F34	E34	GND		
	aDiffGpio_p(66)	DI 63+	F35	E35	DI 53+	aDiffGpio_p(68)	
	aDiffGpio_n(66)	DI 63-	F36	E36	DI 53-	aDiffGpio_n(68)	
		GND	F37	E37	GND		
	aDiffGpio_p(55)	PFI 0+	F38	E38	RSVD		
	aDiffGpio_n(55)	PFI 0-	F39	E39	RSVD		
		GND	F40	E40	GND		

Bank 46

Bank 46

Figure 7. PXIe-6569 with 64 LVDS In, Rows D-C

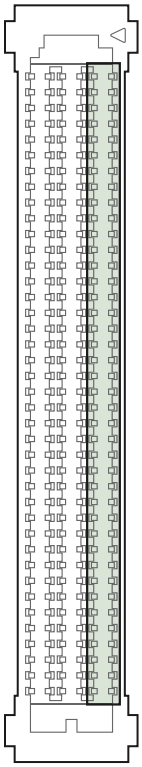


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	D1	C1	GND	
aSeGpio(5)	SE 2	D2	C2	SE 6	aSeGpio(13)
	SE_GND_TERM	D3	C3	SE_GND_TERM	
	GND	D4	C4	GND	
aSeGpio(7)	SE 3	D5	C5	SE 7	aSeGpio(15)
	SE_GND_TERM	D6	C6	SE_GND_TERM	
	GND	D7	C7	GND	
aDiffGpio_p(0)	DI 32+	D8	C8	DI 21+	aDiffGpio_p(4)
aDiffGpio_n(0)	DI 32-	D9	C9	DI 21-	aDiffGpio_n(4)
	GND	D10	C10	GND	
aDiffGpio_p(1)	DI 33+	D11	C11	DI 22+	aDiffGpio_p(8)
aDiffGpio_n(1)	DI 33-	D12	C12	DI 22-	aDiffGpio_n(8)
	GND	D13	C13	GND	
aDiffGpio_p(3)	DI 34+	D14	C14	DI 23+	aDiffGpio_p(12)
aDiffGpio_n(3)	DI 34-	D15	C15	DI 23-	aDiffGpio_n(12)
	GND	D16	C16	GND	
aDiffGpio_p(5)	DI 35+	D17	C17	DI 24+	aDiffGpio_p(11)
aDiffGpio_n(5)	DI 35-	D18	C18	DI 24-	aDiffGpio_n(11)
	GND	D19	C19	GND	
aDiffGpio_p(9)	DI 36+	D20	C20	DI 25+	aDiffGpio_p(16)
aDiffGpio_n(9)	DI 36-	D21	C21	DI 25-	aDiffGpio_n(16)
	GND	D22	C22	GND	
aDiffGpio_p(7)	DI 37+	D23	C23	DI 26+	aDiffGpio_p(14)
aDiffGpio_n(7)	DI 37-	D24	C24	DI 26-	aDiffGpio_n(14)
	GND	D25	C25	GND	
aDiffGpio_p(2)	DI 38+	D26	C26	DI 27+	aDiffGpio_p(19)
aDiffGpio_n(2)	DI 38-	D27	C27	DI 27-	aDiffGpio_n(19)
	GND	D28	C28	GND	
aDiffGpio_p(10)	DI 39+	D29	C29	DI 28+	aDiffGpio_p(17)
aDiffGpio_n(10)	DI 39-	D30	C30	DI 28-	aDiffGpio_n(17)
	GND	D31	C31	GND	
aDiffGpio_p(15)*	**DI 40+***	D32	C32	**DI 29+***	**aDiffGpio_p(21)***
aDiffGpio_n(15)*	**DI 40-***	D33	C33	**DI 29-***	**aDiffGpio_n(21)***
	GND	D34	C34	GND	
aDiffGpio_p(13)	DI 41+	D35	C35	DI 30+	aDiffGpio_p(20)
aDiffGpio_n(13)	DI 41-	D36	C36	DI 30-	aDiffGpio_n(20)
	GND	D37	C37	GND	
aDiffGpio_p(6)	DI 42+	D38	C38	DI 31+	aDiffGpio_p(18)
aDiffGpio_n(6)	DI 42-	D39	C39	DI 31-	aDiffGpio_n(18)
	GND	D40	C40	GND	

Bank 45

Bank 45

Figure 8. PXIe-6569 with 64 LVDS In, Rows B-A

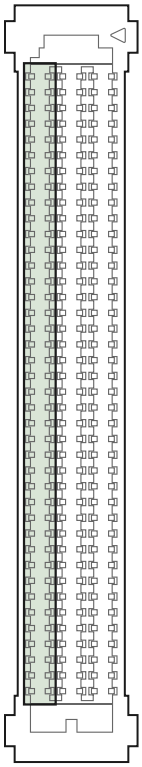


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	B1	A1	GND	
aSeGpio(11)	SE 5	B2	A2	SE 4	aSeGpio(9)
	SE_GND_TERM	B3	A3	SE_GND_TERM	
	GND	B4	A4	GND	
aDiffGpio_p(22)	DI 10+	B5	A5	CLK OUT+	From clocking
aDiffGpio_n(22)	DI 10-	B6	A6	CLK OUT-	circuitry
	GND	B7	A7	GND	
aDiffGpio_p(26)	DI 11+	B8	A8	**DI 0+***	**aDiffGpio_p(28)***
aDiffGpio_n(26)	DI 11-	B9	A9	**DI 0-***	**aDiffGpio_n(28)***
	GND	B10	A10	GND	
aDiffGpio_p(24)	DI 12+	B11	A11	DI 1+	aDiffGpio_p(23)
aDiffGpio_n(24)	DI 12-	B12	A12	DI 1-	aDiffGpio_n(23)
	GND	B13	A13	GND	
aDiffGpio_p(25)	DI 13+	B14	A14	DI 2+	aDiffGpio_p(29)
aDiffGpio_n(25)	DI 13-	B15	A15	DI 2-	aDiffGpio_n(29)
	GND	B16	A16	GND	
aDiffGpio_p(27)	DI 14+	B17	A17	DI 3+	aDiffGpio_p(33)
aDiffGpio_n(27)	DI 14-	B18	A18	DI 3-	aDiffGpio_n(33)
	GND	B19	A19	GND	
aDiffGpio_p(31)	DI 15+	B20	A20	**DI 4+***	**aDiffGpio_p(41)***
aDiffGpio_n(31)	DI 15-	B21	A21	**DI 4-***	**aDiffGpio_n(41)***
	GND	B22	A22	GND	
aDiffGpio_p(32)	DI 16+	B23	A23	DI 5+	aDiffGpio_p(38)
aDiffGpio_n(32)	DI 16-	B24	A24	DI 5-	aDiffGpio_n(38)
	GND	B25	A25	GND	
aDiffGpio_p(40)	DI 17+	B26	A26	DI 6+	aDiffGpio_p(34)
aDiffGpio_n(40)	DI 17-	B27	A27	DI 6-	aDiffGpio_n(34)
	GND	B28	A28	GND	
aDiffGpio_p(36)	DI 18+	B29	A29	DI 7+	aDiffGpio_p(42)
aDiffGpio_n(36)	DI 18-	B30	A30	DI 7-	aDiffGpio_n(42)
	GND	B31	A31	GND	
aDiffGpio_p(45)	DI 19+	B32	A32	DI 8+	aDiffGpio_p(30)
aDiffGpio_n(45)	DI 19-	B33	A33	DI 8-	aDiffGpio_n(30)
	GND	B34	A34	GND	
aDiffGpio_p(39)	DI 20+	B35	A35	DI 9+	aDiffGpio_p(43)
aDiffGpio_n(39)	DI 20-	B36	A36	DI 9-	aDiffGpio_n(43)
	GND	B37	A37	GND	
	RSVD	B38	A38	PFI 1+	aDiffGpio_p(52)
	RSVD	B39	A39	PFI 1-	aDiffGpio_n(52)
	GND	B40	A40	GND	

Bank 44

Bank 44

Figure 9. PXIe-6569 with 64 LVDS Out, Rows F-E

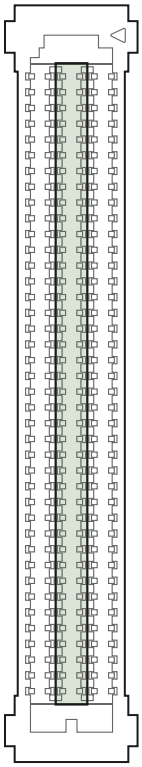


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	F1	E1	GND	
aSeGpio(1)	SE 0	F2	E2	SE 1	aSeGpio(3)
	SE_GND_TERM	F3	E3	SE_GND_TERM	
	GND	F4	E4	GND	
To clocking	CLK IN+	F5	E5	DO 43+	aDiffGpio_p(58)
circuitry	CLK IN-	F6	E6	DO 43-	aDiffGpio_n(58)
	GND	F7	E7	GND	
aDiffGpio_p(46)	DO 54+	F8	E8	DO 44+	aDiffGpio_p(50)
aDiffGpio_n(46)	DO 54-	F9	E9	DO 44-	aDiffGpio_n(50)
	GND	F10	E10	GND	
aDiffGpio_p(48)	DO 55+	F11	E11	DO 45+	aDiffGpio_p(49)
aDiffGpio_n(48)	DO 55-	F12	E12	DO 45-	aDiffGpio_n(49)
	GND	F13	E13	GND	
aDiffGpio_p(47)	DO 56+	F14	E14	DO 46+	aDiffGpio_p(53)
aDiffGpio_n(47)	DO 56-	F15	E15	DO 46-	aDiffGpio_n(53)
	GND	F16	E16	GND	
aDiffGpio_p(51)	DO 57+	F17	E17	DO 47+	aDiffGpio_p(67)
aDiffGpio_n(51)	DO 57-	F18	E18	DO 47-	aDiffGpio_n(67)
	GND	F19	E19	GND	
aDiffGpio_p(69)	DO 58+	F20	E20	DO 48+	aDiffGpio_p(65)
aDiffGpio_n(69)	DO 58-	F21	E21	DO 48-	aDiffGpio_n(65)
	GND	F22	E22	GND	
aDiffGpio_p(63)	DO 59+	F23	E23	DO 49+	aDiffGpio_p(59)
aDiffGpio_n(63)	DO 59-	F24	E24	DO 49-	aDiffGpio_n(59)
	GND	F25	E25	GND	
aDiffGpio_p(56)	DO 60+	F26	E26	DO 50+	aDiffGpio_p(57)
aDiffGpio_n(56)	DO 60-	F27	E27	DO 50-	aDiffGpio_n(57)
	GND	F28	E28	GND	
aDiffGpio_p(61)	DO 61+	F29	E29	DO 51+	aDiffGpio_p(60)
aDiffGpio_n(61)	DO 61-	F30	E30	DO 51-	aDiffGpio_n(60)
	GND	F31	E31	GND	
aDiffGpio_p(64)	DO 62+	F32	E32	DO 52+	aDiffGpio_p(62)
aDiffGpio_n(64)	DO 62-	F33	E33	DO 52-	aDiffGpio_n(62)
	GND	F34	E34	GND	
aDiffGpio_p(66)	DO 63+	F35	E35	DO 53+	aDiffGpio_p(68)
aDiffGpio_n(66)	DO 63-	F36	E36	DO 53-	aDiffGpio_n(68)
	GND	F37	E37	GND	
aDiffGpio_p(55)	PFI 0+	F38	E38	RSVD	
aDiffGpio_n(55)	PFI 0-	F39	E39	RSVD	
	GND	F40	E40	GND	

Bank 46

Bank 46

Figure 10. PXIe-6569 with 64 LVDS Out, Rows D-C

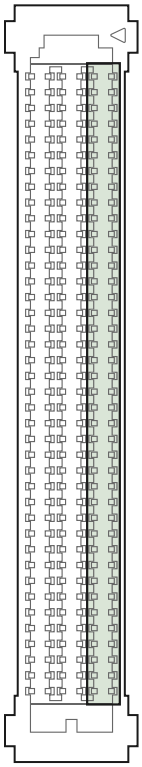


FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	D1	C1	GND	
aSeGpio(5)	SE 2	D2	C2	SE 6	aSeGpio(13)
	SE_GND_TERM	D3	C3	SE_GND_TERM	
	GND	D4	C4	GND	
aSeGpio(7)	SE 3	D5	C5	SE 7	aSeGpio(15)
	SE_GND_TERM	D6	C6	SE_GND_TERM	
	GND	D7	C7	GND	
aDiffGpio_p(0)	DO 32+	D8	C8	DO 21+	aDiffGpio_p(4)
aDiffGpio_n(0)	DO 32-	D9	C9	DO 21-	aDiffGpio_n(4)
	GND	D10	C10	GND	
aDiffGpio_p(1)	DO 33+	D11	C11	DO 22+	aDiffGpio_p(8)
aDiffGpio_n(1)	DO 33-	D12	C12	DO 22-	aDiffGpio_n(8)
	GND	D13	C13	GND	
aDiffGpio_p(3)	DO 34+	D14	C14	DO 23+	aDiffGpio_p(12)
aDiffGpio_n(3)	DO 34-	D15	C15	DO 23-	aDiffGpio_n(12)
	GND	D16	C16	GND	
aDiffGpio_p(5)	DO 35+	D17	C17	DO 24+	aDiffGpio_p(11)
aDiffGpio_n(5)	DO 35-	D18	C18	DO 24-	aDiffGpio_n(11)
	GND	D19	C19	GND	
aDiffGpio_p(9)	DO 36+	D20	C20	DO 25+	aDiffGpio_p(16)
aDiffGpio_n(9)	DO 36-	D21	C21	DO 25-	aDiffGpio_n(16)
	GND	D22	C22	GND	
aDiffGpio_p(7)	DO 37+	D23	C23	DO 26+	aDiffGpio_p(14)
aDiffGpio_n(7)	DO 37-	D24	C24	DO 26-	aDiffGpio_n(14)
	GND	D25	C25	GND	
aDiffGpio_p(2)	DO 38+	D26	C26	DO 27+	aDiffGpio_p(19)
aDiffGpio_n(2)	DO 38-	D27	C27	DO 27-	aDiffGpio_n(19)
	GND	D28	C28	GND	
aDiffGpio_p(10)	DO 39+	D29	C29	DO 28+	aDiffGpio_p(17)
aDiffGpio_n(10)	DO 39-	D30	C30	DO 28-	aDiffGpio_n(17)
	GND	D31	C31	GND	
aDiffGpio_p(15)	DO 40+	D32	C32	DO 29+	aDiffGpio_p(21)
aDiffGpio_n(15)	DO 40-	D33	C33	DO 29-	aDiffGpio_n(21)
	GND	D34	C34	GND	
aDiffGpio_p(13)	DO 41+	D35	C35	DO 30+	aDiffGpio_p(20)
aDiffGpio_n(13)	DO 41-	D36	C36	DO 30-	aDiffGpio_n(20)
	GND	D37	C37	GND	
aDiffGpio_p(6)	DO 42+	D38	C38	DO 31+	aDiffGpio_p(18)
aDiffGpio_n(6)	DO 42-	D39	C39	DO 31-	aDiffGpio_n(18)
	GND	D40	C40	GND	

Bank 45

Bank 45

Figure 11. PXIe-6569 with 64 LVDS Out, Rows B-A



FPGA Signal	Connector Signal			Connector Signal	FPGA Signal
	GND	B1	A1	GND	
aSeGpio(11)	SE 5	B2	A2	SE 4	aSeGpio(9)
	SE_GND_TERM	B3	A3	SE_GND_TERM	
	GND	B4	A4	GND	
aDiffGpio_p(22)	DO 10+	B5	A5	CLK OUT+	From clocking
aDiffGpio_n(22)	DO 10-	B6	A6	CLK OUT-	circuitry
	GND	B7	A7	GND	
aDiffGpio_p(26)	DO 11+	B8	A8	DO 0+	aDiffGpio_p(28)
aDiffGpio_n(26)	DO 11-	B9	A9	DO 0-	aDiffGpio_n(28)
	GND	B10	A10	GND	
aDiffGpio_p(24)	DO 12+	B11	A11	DO 1+	aDiffGpio_p(23)
aDiffGpio_n(24)	DO 12-	B12	A12	DO 1-	aDiffGpio_n(23)
	GND	B13	A13	GND	
aDiffGpio_p(25)	DO 13+	B14	A14	DO 2+	aDiffGpio_p(29)
aDiffGpio_n(25)	DO 13-	B15	A15	DO 2-	aDiffGpio_n(29)
	GND	B16	A16	GND	
aDiffGpio_p(27)	DO 14+	B17	A17	DO 3+	aDiffGpio_p(33)
aDiffGpio_n(27)	DO 14-	B18	A18	DO 3-	aDiffGpio_n(33)
	GND	B19	A19	GND	
aDiffGpio_p(31)	DO 15+	B20	A20	DO 4+	aDiffGpio_p(41)
aDiffGpio_n(31)	DO 15-	B21	A21	DO 4-	aDiffGpio_n(41)
	GND	B22	A22	GND	
aDiffGpio_p(32)	DO 16+	B23	A23	DO 5+	aDiffGpio_p(38)
aDiffGpio_n(32)	DO 16-	B24	A24	DO 5-	aDiffGpio_n(38)
	GND	B25	A25	GND	
aDiffGpio_p(40)	DO 17+	B26	A26	DO 6+	aDiffGpio_p(34)
aDiffGpio_n(40)	DO 17-	B27	A27	DO 6-	aDiffGpio_n(34)
	GND	B28	A28	GND	
aDiffGpio_p(36)	DO 18+	B29	A29	DO 7+	aDiffGpio_p(42)
aDiffGpio_n(36)	DO 18-	B30	A30	DO 7-	aDiffGpio_n(42)
	GND	B31	A31	GND	
aDiffGpio_p(45)	DO 19+	B32	A32	DO 8+	aDiffGpio_p(30)
aDiffGpio_n(45)	DO 19-	B33	A33	DO 8-	aDiffGpio_n(30)
	GND	B34	A34	GND	
aDiffGpio_p(39)	DO 20+	B35	A35	DO 9+	aDiffGpio_p(43)
aDiffGpio_n(39)	DO 20-	B36	A36	DO 9-	aDiffGpio_n(43)
	GND	B37	A37	GND	
	RSVD	B38	A38	PFI 1+	aDiffGpio_p(52)
	RSVD	B39	A39	PFI 1-	aDiffGpio_n(52)
	GND	B40	A40	GND	

Bank 44

Bank 44

Signal Descriptions

The following table describes the signal connections for the PXIe-6569.

Connector Name	Signal Type	Description
DO <0...63>+/-	Data	Positive and negative differential terminals for digital output channels 0 through 63.
DI <0...63>+/-	Data	Positive and negative differential terminals for digital input channels 0 through 63.
PFI <0,1>+/-	Control	Positive and negative differential terminals for bidirectional PFI channels 0 and 1.

Connector Name	Signal Type	Description
SE <0...7>	Data	Terminals for bidirectional single-ended channels 0 through 7.
SE_GND_TERM	Termination	Terminals that provide coupling to the single-ended channels. These signals should be terminated for the best single-ended signal integrity. These are terminated on the PXIe-6569 with 56 Ω to ground. NI recommends that these also be terminated on the user side of the SEARAY cable through a 56 Ω (±10%) resistor to GND. If the user-side termination is not possible, leave it disconnected.
CLKIN+/-, CLKOUT+/-	Clock	Terminals for clocking inputs and outputs.
GND	Ground	Ground reference for signals.
RSVD	Reserved	These pins are reserved and use of them is not supported by NI. Leave these terminals disconnected.

Verifying the Installation in MAX

Use Measurement & Automation Explorer (MAX) to configure your NI hardware. MAX informs other programs about which NI hardware products are in the system and how they are configured. MAX is automatically installed with FlexRIO.

1. Launch MAX.
2. In the configuration tree, expand **Devices and Interfaces** to see the list of installed NI hardware.
   Installed modules appear under the name of their associated chassis.

3. Expand your **Chassis** tree item.
   MAX lists all modules installed in the chassis. Your default names may vary.

   Note Device Manager identifies the PXIe-6569 as the "NI FlexRIO Module (BT - KU035)" or "NI FlexRIO Module (BT - KU060)".

   Note If you do not see your module listed, press <F5> to refresh the list of installed modules. If the module is still not listed, power off the system, ensure the module is correctly installed, and restart.

4. Record the identifier MAX assigns to the hardware. Use this identifier when programming the PXIe-6569.
5. Self-test the hardware by selecting the item in the configuration tree and clicking **Self-Test** in the MAX toolbar.
   MAX self-test performs a basic verification of hardware resources.

Accessing FlexRIO with Integrated I/O Examples

The FlexRIO driver includes several example applications for LabVIEW. These examples serve as interactive tools, programming models, and as building blocks in your own applications. To access all FlexRIO with Integrated I/O getting started examples, complete the following steps.

1. In LabVIEW, click **Help** » **Find Examples**.
2. In the NI Example Finder window that opens, click **Hardware Input and Output** » **FlexRIO** » **Integrated IO** » **Getting Started**.
3. Double click `Getting Started with FlexRIO Integrated IO.vi`.

   The FlexRIO with Integrated IO Project Creator window opens.

4. Select the example that corresponds to the name of your FlexRIO module. The Description window includes a short description of the getting started example for your device. Rename the project, select a location for the project, and click **OK**.
   The Project Explorer window for your new project opens.

Online examples are also available to demonstrate FlexRIO basics, such as using DRAM, acquiring data, and performing high throughput streaming. To access these examples, search `FlexRIO examples` in the **Search the community** field at ni.com/

examples.

PXIe-6569 Examples

Examples specific to PXIe-6569 can be found in the FlexRIO with Integrated IO Project Creator.

NI provides two getting started examples for each LVDS and FPGA configuration of the PXIe-6569: a basic interface and a SERDES interface. The following file names and descriptions demonstrate how the examples would be displayed to a user whose module includes a 32 LVDS In, 32 LVDS Out variation and a KU035 FPGA:

- `PXIe-6569 (32 In 32 Out - KU035) - Basic Interface.vi` demonstrates device configuration and generation/acquisition of digital data using a one sample per cycle interface.
- `PXIe-6569 (32 In 32 Out - KU035) - SerDes Interface.vi` demonstrates device configuration and generation/acquisition of digital data using an eight sample per cycle interface.

Common FlexRIO with Integrated I/O Examples

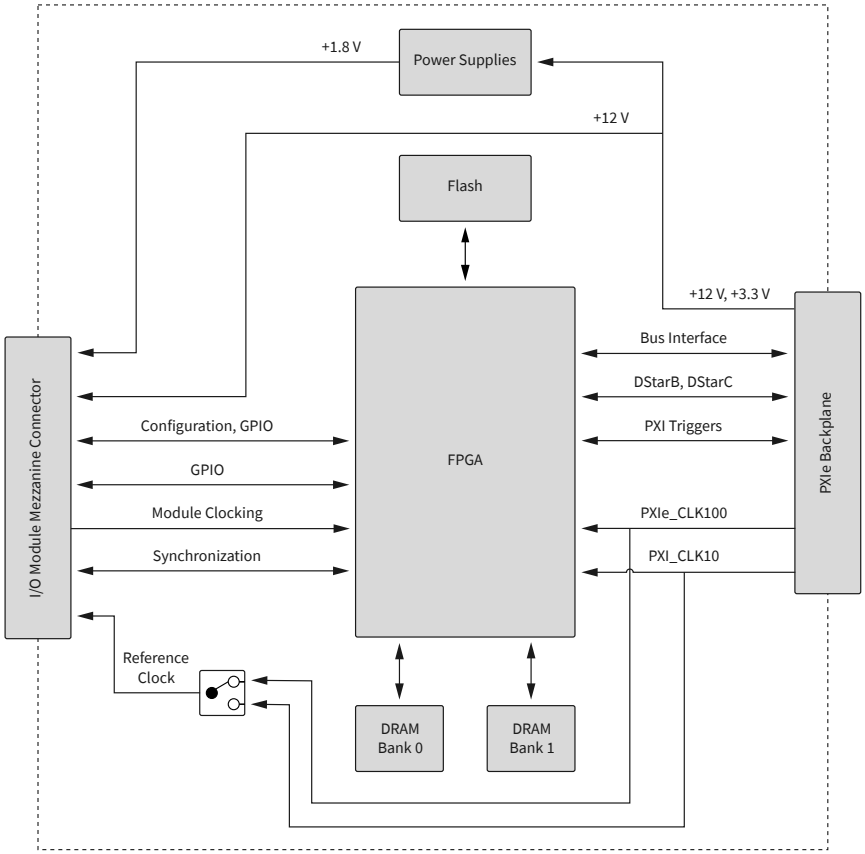
In addition to the examples within the FlexRIO with Integrated IO Project Creator, NI provides several examples that apply to all FlexRIO with Integrated I/O modules to help you perform common tasks.

The following table lists FlexRIO examples you can run from NI Example Finder:

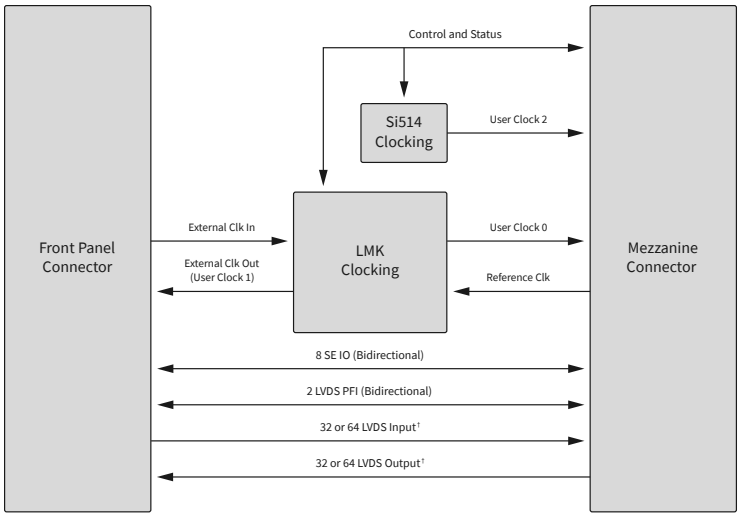
NI Example Finder FlexRIO Example	Description
`Show All FlexRIO with Integrated IO Hardware.vi`	Queries and displays a set of hardware properties from all FlexRIO with Integrated I/O devices in a chassis.
`Vivado Export Getting Started Ultrascale.lvproj`	Demonstrates how to export your LabVIEW FPGA project into Vivado in order to develop your FPGA design in the Vivado ADE.
`Read-Write Calibration Data.vi`	Demonstrates how to read and write calibration data and metadata into the storage space of FlexRIO with Integrated I/O

NI Example Finder FlexRIO Example	Description
	devices.

FPGA Carrier Block Diagram



PXIe-6569 I/O Block Diagram



†The number of fixed direction LVDS input and output depends on the variant of the PXIe-6569.

Component-Level Intellectual Property (CLIP)

The LabVIEW FPGA Module includes component-level intellectual property (CLIP) for HDL IP integration. FlexRIO devices support two types of CLIP: user-defined and socketed.

- ***User-defined CLIP*** allows you to insert HDL IP into an FPGA target, enabling VHDL code to communicate directly with an FPGA VI.
- ***Socketed CLIP*** provides the same IP integration of the user-defined CLIP, but it also allows the CLIP to communicate directly with circuitry external to the FPGA. Adapter module socketed CLIP allows your IP to communicate directly with both the FPGA VI and the external adapter module connector interface.

The PXIe-6569 ships with socketed CLIP items that add module I/O to the LabVIEW project.

Refer to ***Configuring Your Adapter Module Using LabVIEW FPGA*** in FlexRIO documentation for more information about CLIP.

PXIe-6569 CLIP

PXIe-6569 ships with two socketed CLIP options. These socketed CLIP options can be used as-is or can be edited to suit your application.

Refer to the following table for more information about each socketed CLIP's function

and the signals used in each.

Note Both PXIe-6569 CLIP items allow individual clock output inversion.

Table 2. PXIe-6569 Socketed CLIP Items

CLIP Name	Description
PXIe-6569 Basic CLIP	Provides read/write access to all low-voltage differential signal (LVDS) and single-ended channels. You can access the LVDS data and direction lines using a U64 or U32 data type in which each bit position corresponds to an individual channel. You can access the LVDS PFI lines using a boolean data type and the single-ended PFI lines using another boolean data type. Generation channels are clocked by a single generation clock signal, and acquisition channels are clocked by a single acquisition clock signal.
PXIe-6569 SERDES Channel CLIP	Provides read/write access to all LVDS and single-ended channels using a channel-based interface. You can access the LVDS data using a U8 data type and the PFI channels using a boolean data type. Each LVDS line is connected to an OSERDES or ISERDES block that serializes or deserializes, respectively, the signal by a factor of eight by default. During acquisition or generation, the PXIe-6569 reads or writes eight bits of data per channel to or from the IDELAY or ODELAY blocks, which allow for per-channel data delay up to 1.25 ns. All OSERDES and ISERDES blocks are set to double data rate (DDR) mode.

Socketed CLIP Signals

Each LVDS configuration variant of the PXIe-6569 has a different set of signals you must use in the socketed CLIP. Some CLIP signals and data types are specific to the module variant being used. The following table lists the term used in the CLIP signals to represent each associated module variant.

LVDS Configuration Reference in CLIP	PXIe-6569 Variant
Half-In, Half-Out (HIHO)	32 LVDS In, 32 LVDS Out
All In	64 LVDS In
All Out	64 LVDS Out

Refer to Front Panel and Connectors for PXIe-6569 connector signals and the associated FPGA signal information.

PXIe-6569 Basic Socketed CLIP Signals

CLIP Signal Name	Direction	Data Type	Description
IO Ready	From CLIP	Boolean	Indicates successful configuration of the IO module with the current clocking mode settings.
IO Error	From CLIP	I32	Returns IO module errors, to be reported by the driver.
SE_Data_Output_Enable	To CLIP	Boolean	Provides read/write access to all single-ended channels. `SE_Data_Output_Enable` values: • `1`—Use `SE_Data_Wr` to write to the SE data line. • `0`—Use `SE_Data_Rd` to read the SE data line value.
SE_Data_Rd	From CLIP	Boolean	
SE_Data_Wr	To CLIP	Boolean	
LVDS_PFI_Output_Enable	To CLIP	Boolean	Provides read/write access to all low-voltage differential signal (LVDS) PFI channels. `LVDS_PFI_Output_Enable` values: • `1`—Use `LVDS_PFI_Wr` to write to the PFI data line. • `0`—Use `LVDS_PFI_Rd` to read the PFI data line value.
LVDS_PFI_Rd	From CLIP	Boolean	
LVDS_PFI_Wr	To CLIP	Boolean	

CLIP Signal Name	Direction	Data Type	Description
LVDS_Data_Wr	To CLIP	U32 (HIHO); U64 (All Out)	Provides read/write access to all LVDS channels. The least significant bit (LSB) of the U32 (HIHO)/U64 (All In and All Out) corresponds to DIO 0.
LVDS_Data_Rd	From CLIP	U32 (HIHO); U64 (All In)	
Clk Out Inversion DO13 (HIHO)	To CLIP	Boolean	Inverts the generated clock by applying a 180-degree phase shift to the clock signal. The generated clocks are output on DO 13 (HIHO) or on DO 29 (All Out).
Clk Out Inversion DO29 (All Out)	To CLIP	Boolean	
RX Data Clock	From CLIP	Clock	The acquisition clock for acquiring the LVDS input data. Refer to Figure 1 and Figure 5 for additional information.
TX Data Clock	From CLIP	Clock	The generation clock for generating the LVDS output data or acquiring the input data. This clock can be sourced from the Si514 or from the LMK04832 onboard clocking ICs. Refer to Figure 1, Figure 3, and Figure 5 for additional information.
TX/RX Delay Adjust Steps	To CLIP	U16	Sets the number of delay steps to apply to the corresponding TX/RX data line. This delay is applied after the corresponding `TX/RX Delay Adjust Strobe` is asserted. The delay can only be adjusted within the allowable delay limits of the FPGA. Adjusting outside these limits will not change the delay on the FPGA. Refer to the `TX/RX Delay Value Rd` signal description for additional information.
TX/RX Delay Increment	To CLIP	Boolean	`TX/RX Delay Increment` values: • `1`—Increments the line delay by the number of `TX/RX Delay Adjust Steps` when the `TX/RX Delay Adjust Strobe` is asserted.

CLIP Signal Name	Direction	Data Type	Description
			• 0—Decrements the line delay by the number of TX/RX Delay Adjust Steps when the TX/RX Delay Adjust Strobe is asserted.
TX/RX Delay Adjust Strobe	To CLIP	Boolean	Applies the delay to the digital line. TX/RX Delay Adjust Steps and TX/RX Delay Increment should be configured before asserting the TX/RX Delay Adjust Strobe signal. After asserting this strobe signal, wait until TX/RX Delay Done has asserted before asserting TX/RX Delay Adjust Strobe again.
TX/RX Delay Value Rd	From CLIP	U16	Returns the delay count value in delay taps. The delay tap resolution can vary between 2.5 ps and 15 ps. Refer to the ***DS892 - Kintex UltraScale FPGAs Data Sheet: DC and AC Switching Characteristics*** document at www.xilinx.com for additional information. The FPGA delay is restricted to the **Align_Delay** tap value as the lower limit and to 511 delay taps as the upper limit. Refer to the ***UG571 - Ultrascale Architecture SelectIO Resources*** user guide at www.xilinx.com for additional information on the **Align_Delay** tap value. The PXIe-6569 CLIP enforces the upper delay limit by preventing any further delay increments when the tap delay value of 511 is reached. The PXIe-6569 CLIP also enforces the lower delay limit by preventing any further delay decrements when the **Align_Delay** tap value is reached.
TX/RX Delay Done	From CLIP	Boolean	Reports when an increment/decrement operation has completed.

PXIe-6569 SERDES Socketed CLIP Signals

CLIP Signal Name	Direction	Data Type	Description
IO Ready	From CLIP	Boolean	Indicates successful configuration of the IO module with the current clocking mode settings.
IO Error	From CLIP	I32	Returns IO module errors, to be reported by the driver.
SE_Data_Output_Enable	To CLIP	Boolean	Provides read/write access to all single-ended channels. `SE_Data_Output_Enable` values: • `1`—Use `SE_Data_Wr` to write to the SE data line. • `0`—Use `SE_Data_Rd` to read the SE data line value.
SE_Data_Rd	From CLIP	Boolean	
SE_Data_Wr	To CLIP	Boolean	
LVDS_PFI_Output_Enable	To CLIP	Boolean	Provides read/write access to all low-voltage differential signal (LVDS) PFI channels. `LVDS_PFI_Output_Enable` values: • `1`—Use `LVDS_PFI_Wr` to write to the PFI data line. • `0`—Use `LVDS_PFI_Rd` to read the PFI data line value.
LVDS_PFI_Rd	From CLIP	Boolean	
LVDS_PFI_Wr	To CLIP	Boolean	
LVDS_Data_Wr	To CLIP	U8	Signals to read/write data from the LVDS channels. One U8 control/indicator represents the serialized/deserialized 8-bit data for an LVDS channel.
LVDS_Data_Rd	From CLIP	U8	
Clk Out Inversion DO13 (HIHO)	To CLIP	Boolean	Inverts the generated clock by applying a 180-degree phase shift to the clock signal. The generated clocks are output on DO 13 (HIHO and All Out), DO 29 (All Out), and DO 54 (All Out).
Clk Out Inversion DO54 (All Out)	To CLIP	Boolean	
Clk Out Inversion DO29	To CLIP	Boolean	

CLIP Signal Name	Direction	Data Type	Description
(All Out)			
Clk Out Inversion DO13 (All Out)	To CLIP	Boolean	
RX Data Clock (HIHO)	From CLIP	Clock	The acquisition clock for acquiring the LVDS input data. This clock can be sourced from an external DI line or from the TX Data Clock. Refer to Figure 2 and Figure 5 for additional information.
RX Data Clock Bank 44 (All In)	From CLIP	Clock	
RX Data Clock Bank 45 (All In)	From CLIP	Clock	
RX Data Clock Bank 46 (All In)	From CLIP	Clock	
TX Data Clock	From CLIP	Clock	The generation clock for generating the LVDS output data or acquiring the input data. This clock can be sourced from the Si514 or from the LMK04832 onboard clocking ICs. Refer to Figure 2, Figure 3, and Figure 5 for additional information.
TX/RX Delay Adjust Steps	To CLIP	U16	Sets the number of delay steps to apply to the corresponding TX/RX data line. This delay is applied after the corresponding `TX/RX Delay Adjust Strobe` is asserted. The delay can only be adjusted within the allowable delay limits of the FPGA. Adjusting outside these limits will not change the delay on the FPGA. Refer to the `TX/RX Delay Value Rd` signal description for additional information.
TX/RX Delay Increment	To CLIP	Boolean	`TX/RX Delay Increment` values: • `1`—Increments the line delay by the number of `TX/RX Delay Adjust Steps` when the `TX/RX Delay Adjust Strobe` is asserted.

CLIP Signal Name	Direction	Data Type	Description
			• `0`—Decrements the line delay by the number of `TX/RX Delay Adjust Steps` when the `TX/RX Delay Adjust Strobe` is asserted.
TX/RX Delay Adjust Strobe	To CLIP	Boolean	Applies the delay to the digital line. `TX/RX Delay Adjust Steps` and `TX/RX Delay Increment` should be configured before asserting the `TX/RX Delay Adjust Strobe` signal. After asserting this strobe signal, wait until `TX/RX Delay Done` has asserted before asserting `TX/RX Delay Adjust Strobe` again.
TX/RX Delay Value Rd	From CLIP	U16	Returns the delay count value in delay taps. The delay tap resolution can vary between 2.5 ps and 15 ps. Refer to the ***DS892 - Kintex UltraScale FPGAs Data Sheet: DC and AC Switching Characteristics*** document at www.xilinx.com for additional information. The FPGA delay is restricted to the **Align_Delay** tap value as the lower limit and to 511 delay taps as the upper limit. Refer to the ***UG571 - Ultrascale Architecture SelectIO Resources*** user guide at www.xilinx.com for additional information on the **Align_Delay** tap value. The PXIe-6569 CLIP enforces the upper delay limit by preventing any further delay increments when the 511 tap delay value is reached. The PXIe-6569 CLIP also enforces the lower delay limit by preventing any further delay decrements when the **Align_Delay** tap value is reached.
TX/RX Delay Done	From	Boolean	Reports when an increment/decrement

CLIP Signal Name	Direction	Data Type	Description
	CLIP		operation has completed.
Rx Bitslip	To CLIP	Boolean	Rotates the U8 captured data by one bit when asserted. This signal can be used to slip channels to align them in time.

Configuring Clocks

The PXIe-6569 TX/RX Data Clocks can be driven from multiple sources.

The following figures show the different clock sources available on both the Basic and the SERDES CLIPs for all modules.

Figure 12. All In Clock Diagram for Basic CLIP

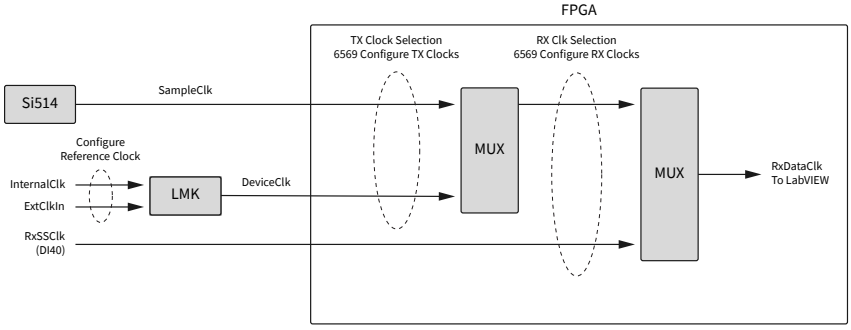


Figure 13. All In Clock Diagram for SERDES CLIP

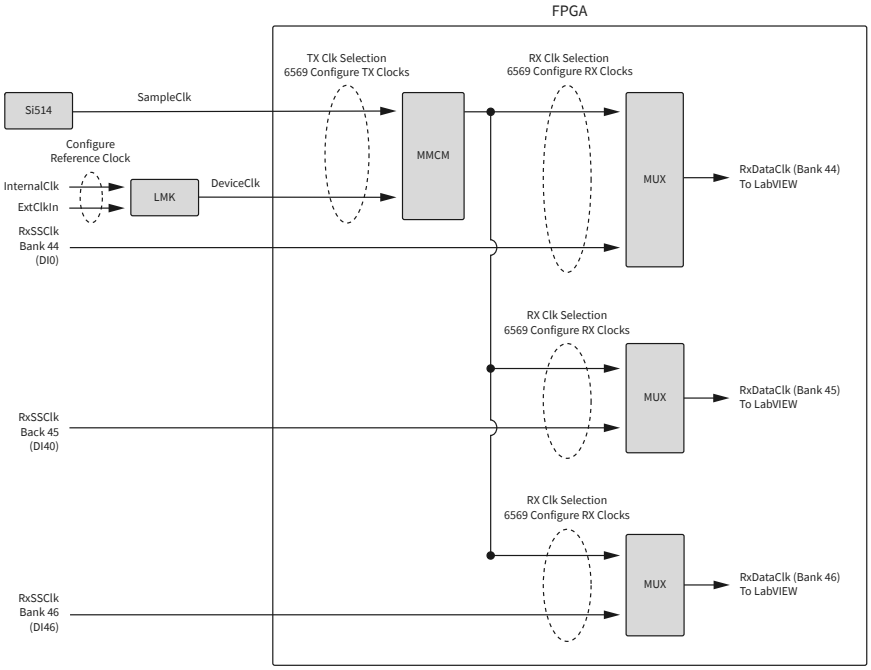


Figure 14. All Out Clock Diagram for Basic CLIP

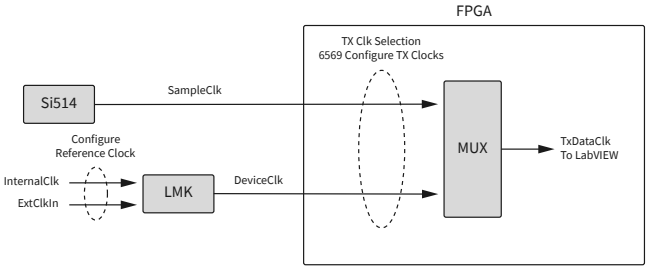


Figure 15. All Out Clock Diagram for SERDES CLIP

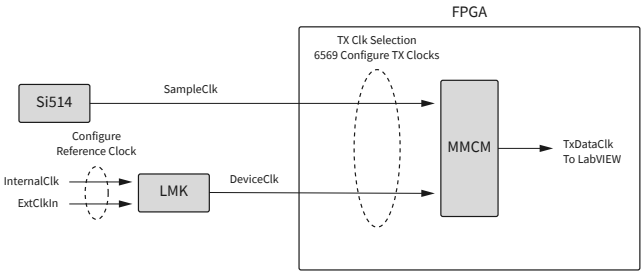


Figure 16. HIHO Clock Diagram for Basic CLIP

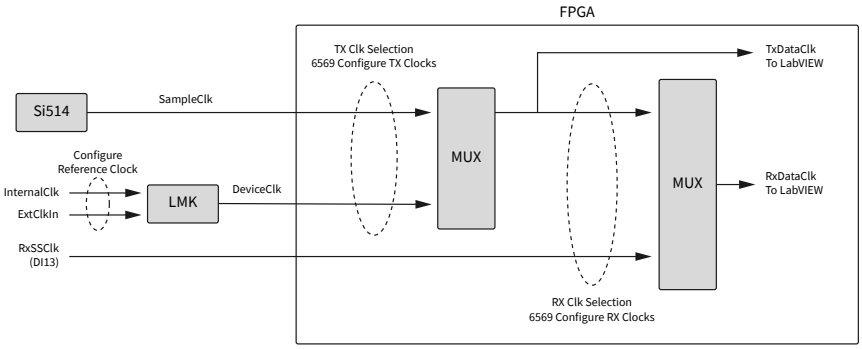
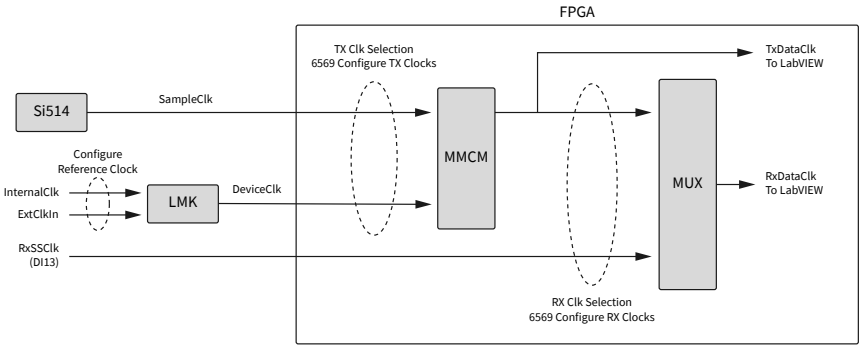


Figure 17. HIHO Clock Diagram for SERDES CLIP



NI Services

Visit ni.com/support to find support resources including documentation, downloads, and troubleshooting and application development self-help such as tutorials and examples.

Visit ni.com/services to learn about NI service offerings such as calibration options, repair, and replacement.

Visit ni.com/register to register your NI product. Product registration facilitates technical support and ensures that you receive important information updates from NI.

NI corporate headquarters is located at 11500 N Mopac Expwy, Austin, TX, 78759-3504, USA.