NI 6584R User Guide and Specifications

The NI 6584 is an RS-485/422 adapter module designed to work in conjunction with your NI FlexRIO™ FPGA module. This document contains signal information and specifications for the NI 6584R, comprised of an NI FlexRIO FPGA module and the NI 6584. This document also contains tutorial sections that demonstrate how to generate and acquire samples using a LabVIEW FPGA example VI and how to create and run your own LabVIEW project with the NI 6584R.

The NI 6584 comes in full duplex and half duplex versions. The full duplex version has separate RX (receive) and TX (transmit) pins for acquisition and generation, respectively. The half duplex version performs either acquisition or generation on the same pin.

Contents

> **Note** Before configuring your NI 6584R, you must install the appropriate software and hardware. Refer to the *NI FlexRIO FPGA Module Installation Guide and Specifications* for installation instructions. Figure 1 shows an example of a properly connected NI 6584R.
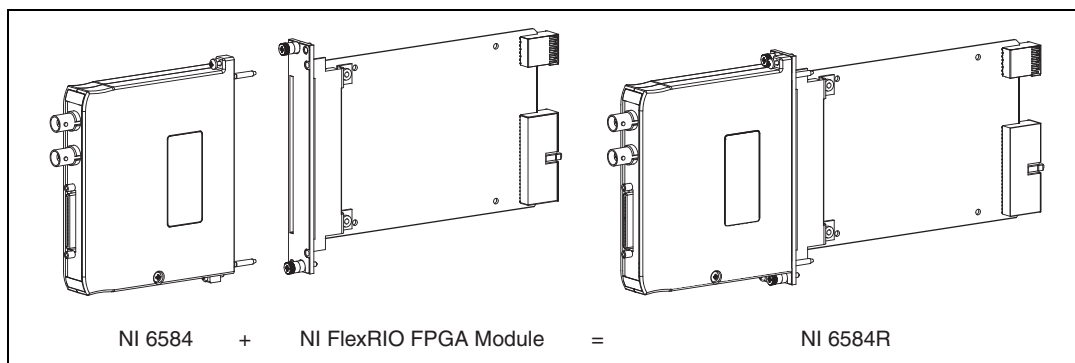

NI 6584   +   NI FlexRIO FPGA Module   =   NI 6584R

Figure 1. NI 6584R

Connecting Signals

Front Panel

Figure 2 shows the Connector 0 (Ports 1–8) pin assignments for the NI 6584 in the full duplex configuration. Connector 0 (Ports 1–8) is a 68-pin VHDCI connector that distributes its signals through the NI VHDCI-to-Eight DB9 cable (197546-01) to the device under test (DUT). RX pins are acquisition signals, and TX pins are generation signals. The full duplex configuration of the NI 6584 has individual RX and TX pins.
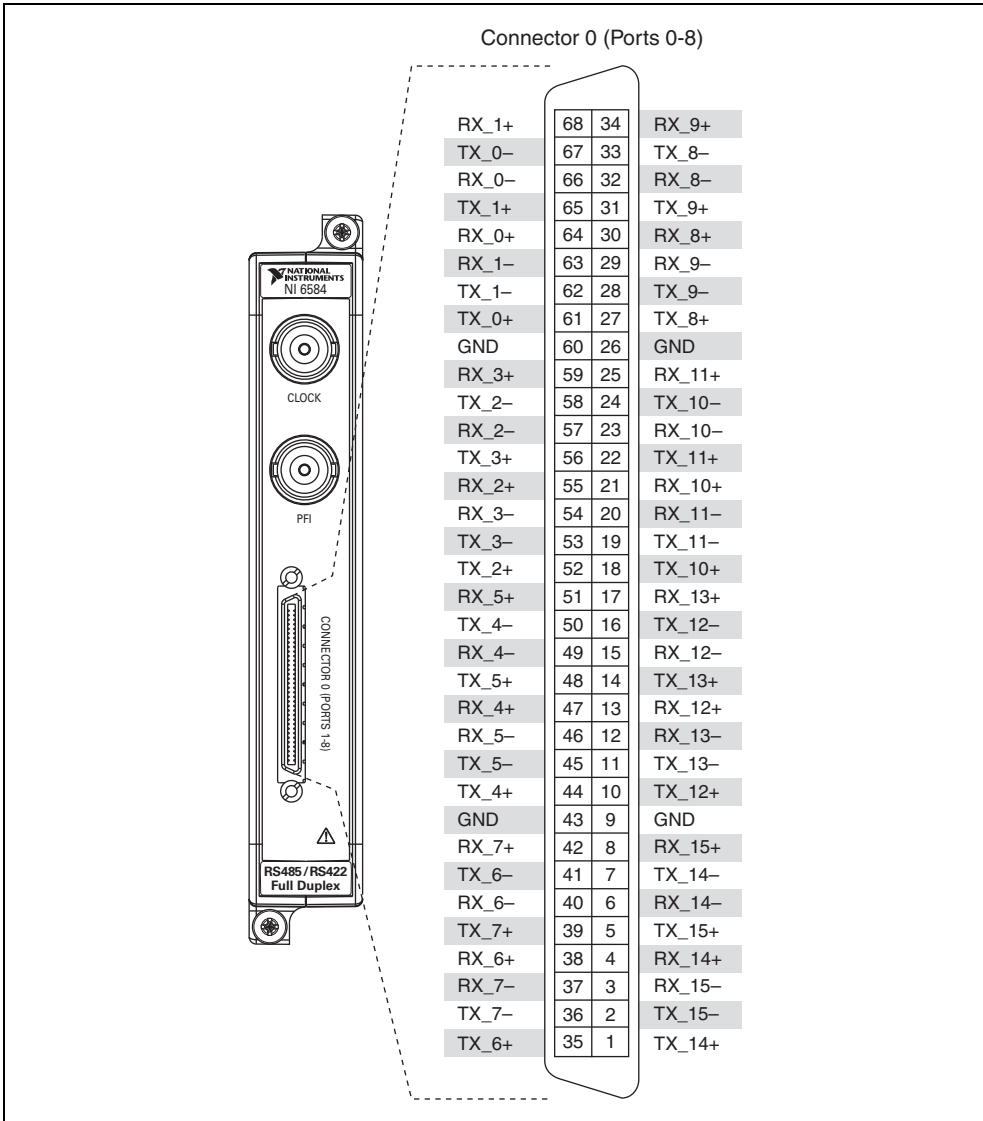
Connector 0 (Ports 0-8)

RX_1+	68	34	RX_9+
TX_0–	67	33	TX_8–
RX_0–	66	32	RX_8–
TX_1+	65	31	TX_9+
RX_0+	64	30	RX_8+
RX_1–	63	29	RX_9–
TX_1–	62	28	TX_9–
TX_0+	61	27	TX_8+
GND	60	26	GND
RX_3+	59	25	RX_11+
TX_2–	58	24	TX_10–
RX_2–	57	23	RX_10–
TX_3+	56	22	TX_11+
RX_2+	55	21	RX_10+
RX_3–	54	20	RX_11–
TX_3–	53	19	TX_11–
TX_2+	52	18	TX_10+
RX_5+	51	17	RX_13+
TX_4–	50	16	TX_12–
RX_4–	49	15	RX_12–
TX_5+	48	14	TX_13+
RX_4+	47	13	RX_12+
RX_5–	46	12	RX_13–
TX_5–	45	11	TX_13–
TX_4+	44	10	TX_12+
GND	43	9	GND
RX_7+	42	8	RX_15+
TX_6–	41	7	TX_14–
RX_6–	40	6	RX_14–
TX_7+	39	5	TX_15+
RX_6+	38	4	RX_14+
RX_7–	37	3	RX_15–
TX_7–	36	2	TX_15–
TX_6+	35	1	TX_14+

CONNECTOR 0 (PORTS 1-8)

NI 6584

CLOCK

PFI

RS485 / RS422
Full Duplex

Figure 2. NI 6584 Connector 0 Pin Assignments (Full Duplex)

Figure 3 shows the Connector 0 (Ports 1–8) pin assignments for the NI 6584 in the half duplex configuration. Connector 0 (Ports 1–8) is a 68-pin VHDCI connector that distributes its signals through the NI VHDCI-to-Eight DB9 cable (197546-01) to the DUT. The half duplex configuration of the NI 6584 places both the RX (acquisition) and TX (generation) signals on the same pins.
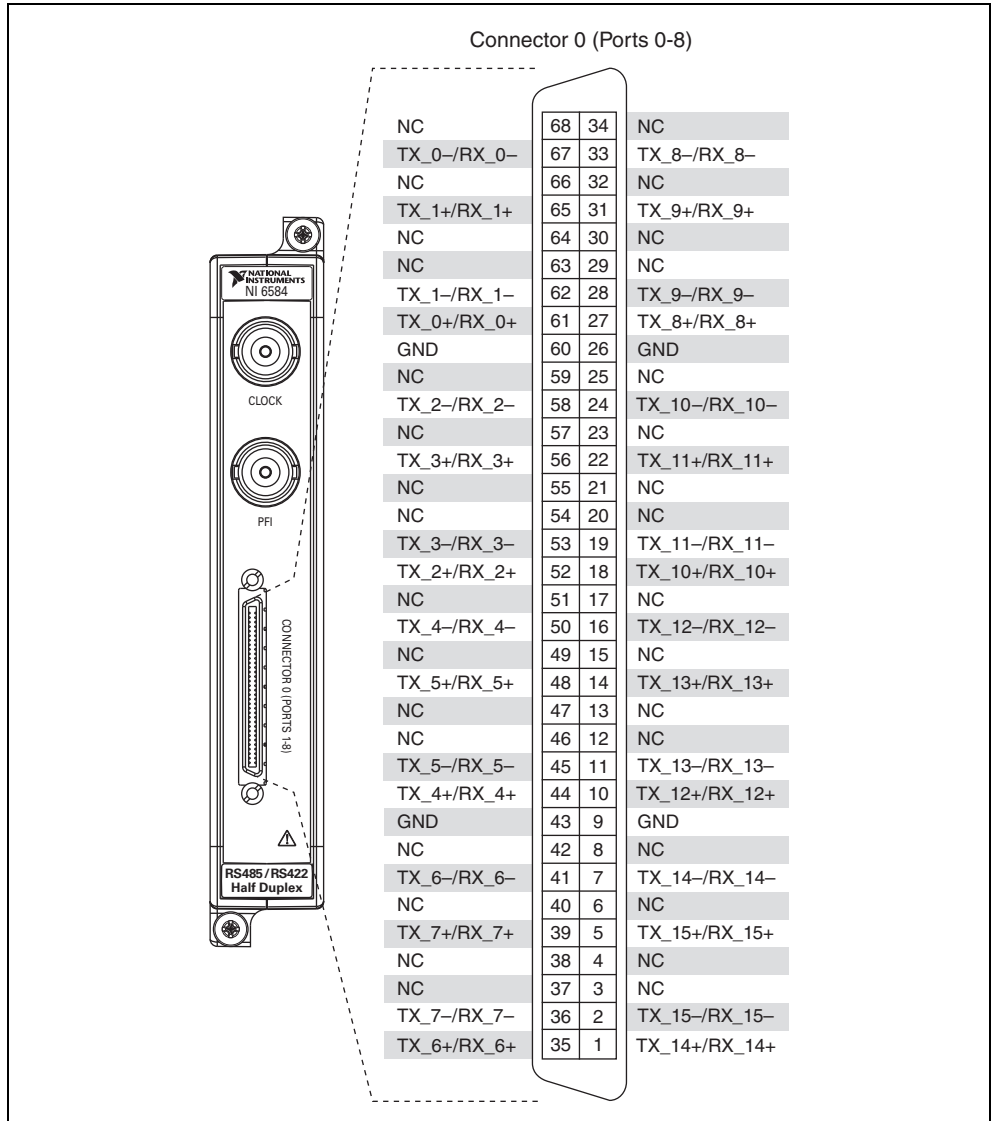
Connector 0 (Ports 0-8)

Signal	Pin	Pin	Signal
NC	68	34	NC
TX_0–/RX_0–	67	33	TX_8–/RX_8–
NC	66	32	NC
TX_1+/RX_1+	65	31	TX_9+/RX_9+
NC	64	30	NC
NC	63	29	NC
TX_1–/RX_1–	62	28	TX_9–/RX_9–
TX_0+/RX_0+	61	27	TX_8+/RX_8+
GND	60	26	GND
NC	59	25	NC
TX_2–/RX_2–	58	24	TX_10–/RX_10–
NC	57	23	NC
TX_3+/RX_3+	56	22	TX_11+/RX_11+
NC	55	21	NC
NC	54	20	NC
TX_3–/RX_3–	53	19	TX_11–/RX_11–
TX_2+/RX_2+	52	18	TX_10+/RX_10+
NC	51	17	NC
TX_4–/RX_4–	50	16	TX_12–/RX_12–
NC	49	15	NC
TX_5+/RX_5+	48	14	TX_13+/RX_13+
NC	47	13	NC
NC	46	12	NC
TX_5–/RX_5–	45	11	TX_13–/RX_13–
TX_4+/RX_4+	44	10	TX_12+/RX_12+
GND	43	9	GND
NC	42	8	NC
TX_6–/RX_6–	41	7	TX_14–/RX_14–
NC	40	6	NC
TX_7+/RX_7+	39	5	TX_15+/RX_15+
NC	38	4	NC
NC	37	3	NC
TX_7–/RX_7–	36	2	TX_15–/RX_15–
TX_6+/RX_6+	35	1	TX_14+/RX_14+

NATIONAL INSTRUMENTS
NI 6584

CLOCK

PFI

CONNECTOR 0 (PORTS 1-8)

RS485 / RS422
Half Duplex

Figure 3. NI 6584 Connector 0 Pin Assignments (Half Duplex)

Caution Connections that exceed any of the maximum ratings of input or output signals on the NI 6584R can damage the device and the chassis. NI is *not* liable for any damage resulting from such signal connections. For the maximum input and output ratings for each signal, refer to the *Specifications* section of this document.

Cables

Figure 4 shows the NI VHDCI-to-Eight DB9 cable (197546-01) connected to the NI 6584 at Connector 0 (Ports 1–8) and then fanning out to display each of its eight, nine-pin port connectors.
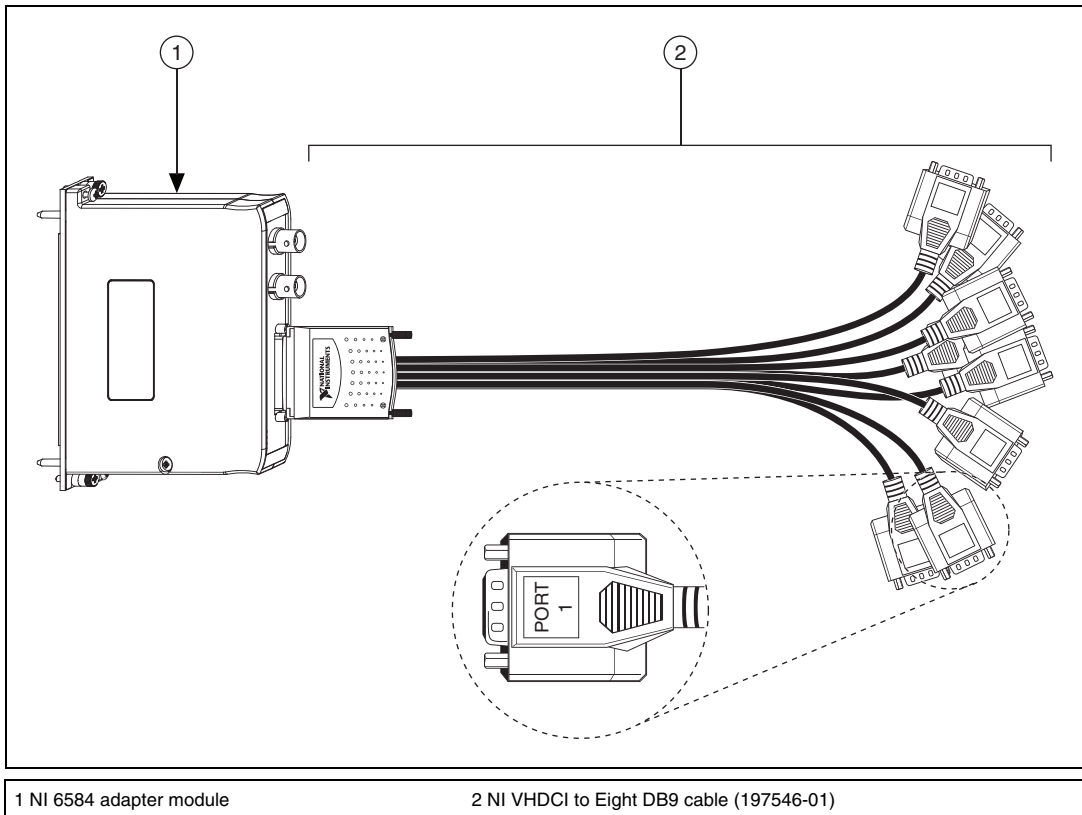

| 1 NI 6584 adapter module | 2 NI VHDCI to Eight DB9 cable (197546-01) |

Figure 4. NI 6584 and NI VHDCI to Eight DB9 Cable

Tables 1 and 2 show the pin assignments for each of the eight port connectors on the NI VHDCI-to-Eight DB9 cable. Two channels are assigned to each port.

Table 1. Pin Assignments on Each Port (Full Duplex)
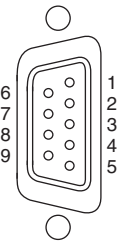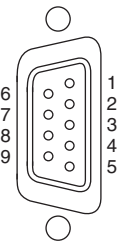
Connector	Pin	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	Port 8
	1	GND	GND	GND	GND	GND	GND	GND	GND
	2	RX_1+	RX_3+	RX_5+	RX_7+	RX_9+	RX_11+	RX_13+	RX_15+
	3	TX_1+	TX_3+	TX_5+	TX_7+	TX_9+	TX_11+	TX_13+	TX_15+
	4	RX_0+	RX_2+	RX_4+	RX_6+	RX_8+	RX_10+	RX_12+	RX_14+
	5	RX_0–	RX_2	RX_4–	RX_6–	RX_8–	RX_10–	RX_12–	RX_14–
	6	RX_1–	RX_3–	RX_5–	RX_7–	RX_9–	RX_11–	RX_13–	RX_15–
	7	TX_1–	TX_3–	TX_5–	TX_7–	TX_9–	TX_11–	TX_13–	TX_15–
	8	TX_0+	TX_2+	TX_4+	TX_6+	TX_8+	TX_10+	TX_12+	TX_14+
	9	TX_0–	TX_2–	TX_4–	TX_6–	TX_8–	TX_10–	TX_12–	TX_14–
GND = Ground									

Table 2. Pin Assignments on Each Port (Half Duplex)

Connector	Pin	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7	Port 8
	1	GND	GND	GND	GND	GND	GND	GND	GND
	2	NC	NC	NC	NC	NC	NC	NC	NC
	3	TX_1+/ RX_1+	TX_3+/ RX_3+	TX_5+/ RX_5+	TX_7+/ RX_7+	TX_9+/ RX_9+	TX_11+/ RX_11+	TX_13+/ RX_13+	TX_15+/ RX_15+
	4	NC	NC	NC	NC	NC	NC	NC	NC
	5	NC	NC	NC	NC	NC	NC	NC	NC
	6	NC	NC	NC	NC	NC	NC	NC	NC
	7	TX_1–/ RX_1–	TX_3–/ RX_3–	TX_5–/ RX_5–	TX_7–/ RX_7–	TX_9–/ RX_9–	TX_11–/ RX_11–	TX_13–/ RX_13–	TX_15–/ RX_15–
	8	TX_0+/ RX_0+	TX_2+/ RX_2+	TX_4+/ RX_4+	TX_6+/ RX_6+	TX_8+/ RX_8+	TX_10+/ RX_10+	TX_12+/ RX_12+	TX_14+/ RX_14+
	9	TX_0–/ RX_0–	TX_2–/ RX_2–	TX_4–/ RX_4–	TX_6–/ RX_6–	TX_8–/ RX_8–	TX_10–/ RX_10–	TX_12–/ RX_12–	TX_14–/ RX_14–
GND = Ground NC = No connect									

FPGA

Table 3 lists the NI 6584 connector signals and the corresponding NI FlexRIO FPGA module signals necessary for designing a custom component-level IP (CLIP). Channel 0 and Clock I/O are global clocks. The _CC suffix on the RX Data signals identifies channels that are capable of receiving a regional clock. Refer to the *Virtex-5 FPGA User Guide* at www.xilinx.com for more information about regional clocks on your FPGA.

Table 3. NI 6584 Connector Signals and NI FlexRIO FPGA Module Signals

NI 6584		NI FlexRIO FPGA Module							
Connector	Signal Name	TX Data	TX Data Bank	TX Data Enable	TX Data Enable Bank	RX Data	RX Data Bank	RX Data Enable	RX Data Enable Bank
Connector 0 (VHDCI)	Channel 0+/–	GPIO_21	1	GPIO_1	0	GCLK_SE	—	GPIO_18	1
	Channel 1+/–	GPIO_22	1	GPIO_14	0	GPIO_7_CC	0	GPIO_14_n	0
	Channel 2+/–	GPIO_8	0	GPIO_12	0	GPIO_23_CC	1	GPIO_27	1
	Channel 3+/–	GPIO_9	0	GPIO_13	0	GPIO_24_CC	1	GPIO_29	1
	Channel 4+/–	GPIO_54	3	GPIO_60	3	GPIO_39_CC	2	GPIO_61	3
	Channel 5+/–	GPIO_43	2	GPIO_47	2	GPIO_40_CC	2	GPIO_65	3
	Channel 6+/–	GPIO_52	3	GPIO_51	3	GPIO_37_CC	2	GPIO_35	2
	Channel 7+/–	GPIO_53	3	GPIO_63	3	GPIO_38_CC	2	GPIO_46	2
	Channel 8+/–	GPIO_19	1	GPIO_16	1	GPIO_5_CC	0	GPIO_17	1
	Channel 9+/–	GPIO_20	1	GPIO_29_n	1	GPIO_6_CC	0	GPIO_30	1
	Channel 10+/–	GPIO_10	0	GPIO_2	0	GPIO_25_CC	1	GPIO_3	0
	Channel 11+/–	GPIO_11	0	GPIO_28	0	GPIO_26_CC	1	GPIO_28_n	1
	Channel 12+/–	GPIO_55	3	GPIO_35_n	2	GPIO_58_CC	3	GPIO_36	2
	Channel 13+/–	GPIO_44	2	GPIO_48	2	GPIO_59_CC	3	GPIO_64	3
	Channel 14+/–	GPIO_41	2	GPIO_49	3	GPIO_56_CC	3	GPIO_50	3
	Channel 15+/–	GPIO_42	2	GPIO_62	3	GPIO_57_CC	3	GPIO_45	2
BNC	Clock I/O	GPIO_16_n	1	GPIO_17_n	1	GCLK_LVDS	—	—	—
BNC	PFI	GPIO_0	0	GPIO_1_n	0	GPIO_4_CC	0	—	—

Using Your NI 6584R with a LabVIEW FPGA Example VI

The NI FlexRIO Adapter Module Support software includes a variety of example projects to help get you started creating your LabVIEW FPGA program. This section demonstrates how to use an existing LabVIEW FPGA example project and a basic universal asynchronous receiver/transmitter (UART) interface to perform serial generation and/or acquisition with the NI 6584R. This example requires an NI VHDCI-to-Eight DB9 cable.

Note Examples available for your device are dependent on the device-specific minimum software requirements. For more information about the minimum software requirements for your device, visit `ni.com/info` and enter `rdsoftwareversion` as the Info Code.

Each NI 6584R example project includes the following parts:

- A LabVIEW FPGA VI that can be compiled and run embedded in FPGA hardware
- A Host VI that runs in LabVIEW for Windows and interacts with the LabVIEW FPGA VI

Note In NI application software, NI FlexRIO adapter modules are referred to as *IO Modules*.

Complete the following steps to run an example that generates and/or acquires data.

1. Connect the VHDCI end of an NI VHDCI-to-Eight DB9 cable to Connector 0 (Ports 1–8) on the NI 6584R.
2. Connect the Port 1 DB9 connector from the NI VHDCI-to-Eight DB9 cable to the DUT with proper termination.

Note This exercise shows how to complete these tasks for the full duplex version of the NI 6584. To complete these tasks with a half duplex version, replace every instance of *Full Duplex* in these instructions with *Half Duplex*.

3. Launch LabVIEW.
4. In the **Getting Started** window, select **Find Examples** to display the NI Example Finder.
5. In the **NI Example Finder** window, select **Hardware Input and Output»FlexRIO»IO Modules» NI 6584**.
6. Select **NI 6584 Basic Software UART Full Duplex.lvproj**.
7. In the **Project Explorer** window, double-click **NI 6584 Basic Serial UART Full Duplex (Host).vi** under **My Computer**. The host VI opens. This VI uses the NI 7952R as the FPGA target by default. To use the NI 6584, complete the following steps to change the FPGA target.

Note All example projects are configured for RIO0. If your device is not named RIO0 (verify in MAX), you must update the target device name by right-clicking your device in the **Project Explorer** window, selecting **Properties**, and entering the correct target device name in the **Resource** control.

   a. Select **Window»Show Block Diagram** to open the VI block diagram.
   b. On the block diagram, right-click the Open FPGA VI Reference (PXI-7952R) function and select **Configure Open FPGA VI Reference**.
   c. In the **Configure Open FPGA VI Reference** window, click the **Browse Project** button in the Open VI section.
   d. In the **Select VI** window that opens, expand the tree view for your device, select the VI under your device and click **OK**.
   e. Click **OK** in the **Configure Open FPGA VI Reference** window.
   f. Save the VI.

8. On the front panel, complete the following steps to generate data.

   a. Enter data into the **Gen Data (Hex)** control. This displays the data to be generated in hexadecimal format.

   b. Set the **Gen Data Width** control to a number between 1 and 32. The default is 8.

   c. Set the **Stop Bits** control to a number between 1 and 65,535. The default is 1.

   d. Set the **Gen Clock Divider** control. The default generation rate of the UART is 16 Mbit/s, but it is capable of running at any integer division of this rate. For example, to generate at a rate of 9,600 Kbit/s, set the **Gen Clock Divider** to 1,667. (16 Mbit/s divided by 1,667 clock cycles produces a clock rate of 9,598 Kbit/s.)

   e. To generate data continuously, leave the **Write Data** control enabled. To stop generation, click the **Write Data** button.

9. On the front panel, complete the following steps to acquire data.

   a. Set the **Acq Data Width** control to a number between 1 and 32. The default is 8.

   b. Set the **Samples to Acquire** control to a number between 1 and 1,023. The default is 5.

   c. Set the **Acq Clock Divider**. The default acquisition rate of the UART is 16 Mbit/s, but it is capable of running at any integer division of this rate. For example, to acquire at a rate of 9,600 Kbit/s, set the **Acq Clock Divider** to 1,667. (16 Mbit/s divided by 1,667 clock cycles produces a clock rate of 9,598 Kbit/s.)

   d. To acquire data continuously, leave the **Read Data** control enabled. To stop acquisition, click the **Read Data** button.

Note If you are using the half duplex version of the NI 6584R, you cannot both acquire and generate data at the same time. You must choose whether to enable the **Write Data** button or the **Read Data** button.

10. Click the **Run** button to run the VI. The NI 6584R generates and/or acquires data on Channel 0 of Port 1. If a timeout error occurs, the number of samples acquired did not reach the value specified in the **Samples to Acquire** control.

Figure 5 shows the front panel controls and indicators configured for generating and acquiring data in this example VI. Close the VI when you are finished generating and acquiring data.
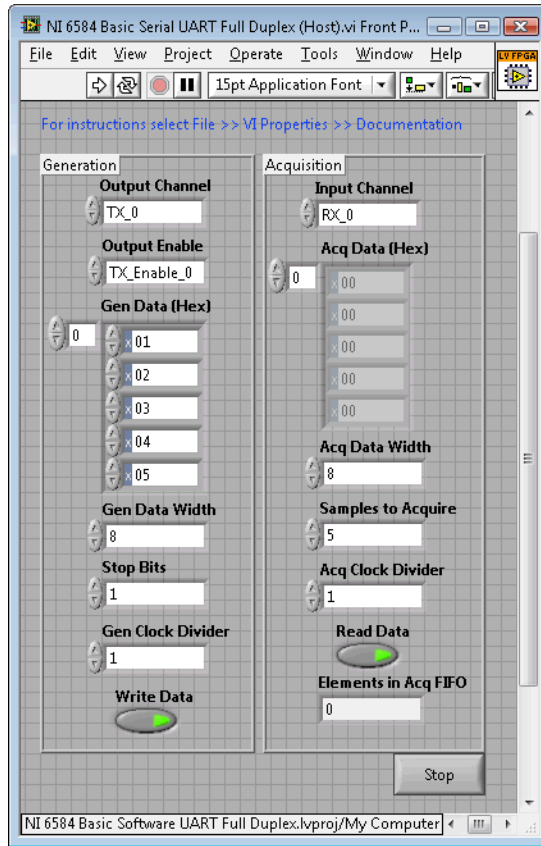

Figure 5. NI 6584 Basic Serial UART Full Duplex (Host) VI Front Panel

Creating a LabVIEW Project and Running a VI on an FPGA Target

This section demonstrates how to create a LabVIEW project, an FPGA VI, and a host VI that writes and reads data on the NI 6584R. This exercise also demonstrates how to compile the FPGA VI on your target and run a VI on the host machine. This exercise shows how to complete these tasks for the full duplex version of the NI 6584. To complete these tasks with the half duplex version, replace every instance of *Full Duplex* in these instructions with *Half Duplex*.

Note Disconnect all signals from the NI 6584R connectors before running this VI.

Creating a Project

1. Launch LabVIEW.

2. In the **Getting Started** window, select **Empty Project**. The new project opens in the **Project Explorer** window.

3. Save the project as Data RW.lvproj.

Creating an FPGA Target VI

1. In the **Project Explorer** window, right-click **My Computer** and select **New»Targets and Devices**.

2. In the **Add Targets and Devices on My Computer** window, select the **Existing Target or Device** radio button and expand the FPGA Target. The target is displayed.

3. Select your device and click **OK**. The target and target properties are loaded into the project tree.

4. In the **Project Explorer** window, right-click the FPGA target and select **New»VI**. A blank VI opens.

5. Select Window **Show»Block Diagram** to open the block diagram window.

6. In the **Project Explorer** window, select **FPGA Target** (**RIO***x*, **PXI-79***xx***R**).

7. Right-click **IO Module** and select **Properties**. In the General category, you can see the available component-level IP (CLIP) for the NI 6584 in the Component Level IP pane. If the category information is dimmed, select the **Enable IO Module** checkbox.

   Note   For more information about CLIP for the NI 6584, refer to the *NI 6584 Component-Level Intellectual Property* section of this document.

8. Select **NI 6584 Full-Duplex** to configure the software for the full duplex version of the NI 6584R.

9. Select **NI 6584 Basic Connector** to use the connector-based CLIP. Click **OK**.

10. In the **Project Explorer** window, select **IO Module (NI 6584 Full-Duplex : NI 6584 Basic Connector)**.

11. Select **RX**, **TX**, and **TX_Enable** and drag them onto the block diagram.

12. Add a While Loop around the three Data nodes.

13. Wire an indicator from the output of the **IO Module\RX**.

14. Wire a control to the input of the **IO Module\TX**.

15. Wire a numeric constant to the input of the **IO Module\TX_Enable** node. Set the value to 65,535. (This enables all 16 TX channels like `0xFFFF` in hexadecimal.)

16. Wire a False constant to the stop condition of the While Loop.

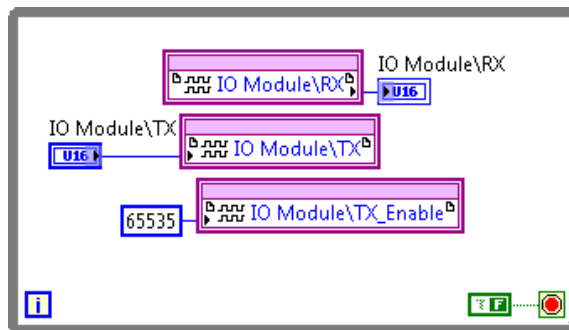   Your block diagram should now resemble the block diagram in Figure 6.

Figure 6. Data RW (FPGA).vi Block Diagram

17. Save the VI as `Data RW (FPGA).vi`.

18. Close the VI.

19. In the Project Explorer window under **My Computer**, expand the tree view for your device, right-click **Data RW (FPGA).vi** and select **Compile** to compile the files for your target.

The **Generating Intermediate Files** window opens and displays the compilation progress.
The **LabVIEW FPGA Compile Server** window opens and runs. The compilation takes several minutes.

20. When the compilation finishes, click the **Stop Server** button.

21. Click **OK** in the **Successful Compile Report** window. Close the VI without saving changes.

Creating a Host VI

1. In the **Project Explorer** window, right-click **My Computer** and select **New»VI**. A blank VI opens. Open the block diagram window.

2. Add the Open FPGA VI Reference function (from the FPGA Interface palette) to the block diagram.

3. Right-click the Open FPGA VI Reference function and select **Configure Open FPGA VI Reference**.

4. In the **Configure Open FPGA VI Reference** window, select the **VI** option button.

5. In the **Select VI** window that opens, select **Data RW (FPGA).vi** under your device, and click **OK**.

6. Select the **Run the FPGA VI** check box if it is not already selected.

7. Click **OK** in the **Configure Open FPGA VI Reference** window. The new target name appears under the Open FPGA VI Reference function on the block diagram.

8. Add a While Loop to the block diagram with a control on the loop condition.

9. Add the Read/Write Control function (from the FPGA Interface palette) inside the While Loop.

10. Wire the Open FPGA VI Reference function **FPGA VI Reference Out** indicator to the **FPGA VI Reference In** control on the Read/Write Control function.

11. Wire the Open FPGA VI Reference function **error out** indicator to the Read/Write Control function **error in** control.

12. Click the **Unselected** input of Read/Write Control function and select **IO Module\TX**.

13. Wire a control to the **IO Module\TX** input.

14. Expand the bottom of the Read/Write Control function to expose another input. Click the new input and select **IO Module\RX**.

15. Wire an indicator from the **IO Module\RX** output.

16. Add the Close FPGA VI Reference function (from the FPGA Interface palette) outside the While Loop.

17. Wire the Read/Write Control function **FPGA VI Reference Out** indicator to the Close FPGA VI Reference function **FPGA VI Reference In** control.

18. Wire the Read/Write Control **error out** parameter to the Close FPGA VI Reference **error in** parameter.

19. Right-click the stop condition of the While Loop and add an control. Label this control `Stop`.

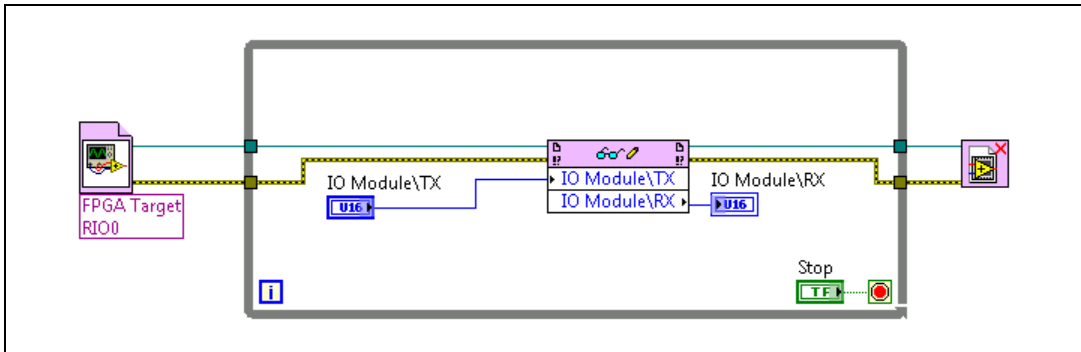Your block diagram should now resemble the block diagram in Figure 7.

Figure 7. Data RW (Host).vi Block Diagram

20. Save the VI as `Data RW (Host).vi`.

Running the Host VI

1. Open the front panel of `Data RW (Host).vi`.

2. Click the **Run** button to run the VI.

3. Enter a number in the **IO Module\TX** control to push different values to all 16 channels. When you use the half duplex version of the NI 6584R, the number you enter in the **IO Module\TX** appears in the **IO Module\RX** control. When you use the full duplex version of the NI 6584R, the **IO Module\RX** indicator returns the number 65,535 because all RX lines are floating.

4. Click the **Stop** button on the front panel and close the VI.

NI 6584 Component-Level Intellectual Property

The LabVIEW FPGA Module includes a feature for HDL IP integration called CLIP. NI FlexRIO devices support two types of CLIP: user-defined and socketed.

- *User-defined CLIP* allows users 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 functionality of the user-defined CLIP, but 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.

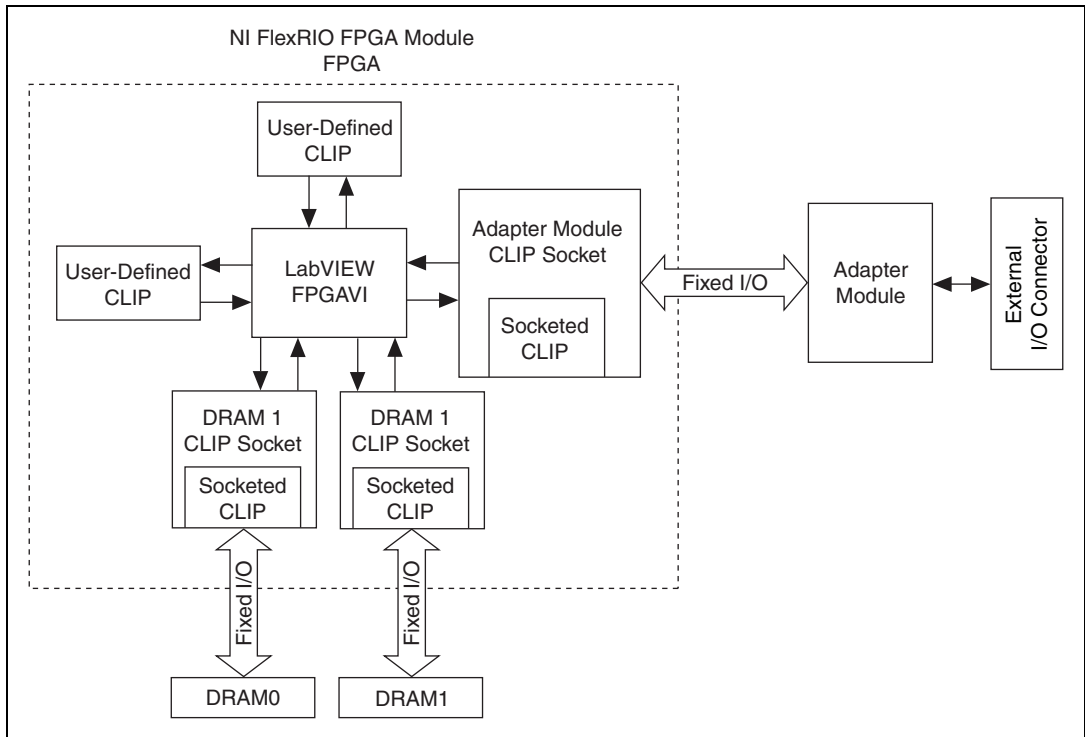Figure 8 shows the relationship between an FPGA VI and CLIP.

Figure 8. CLIP Relationship

The NI 6584 ships with socketed CLIP that is used to add module I/O to the LabVIEW project. The NI-developed NI 6584 CLIP are as follows:

- **NI 6584 Basic Channel**—Provides read/write access to all RS-485/422 channels using a simple channel-based interface. Each output line has a write enable signal. Each input line is always enabled. This CLIP provides a clock signal for import or export on the BNC CLOCK connector. The clock input signals from the NI 6584 are passed to LabVIEW FPGA for use in the FPGA VI. This CLIP also allows for individual clock output signal inversion.

- **NI 6584 Basic Connector**—Provides read/write access to all RS-485/422 channels on Connector 0 (Ports 1-8). The individual data lines for each connector are accessed using a U16 data type in LabVIEW FPGA. Each output line has a write enable signal. Each input line is always enabled. This CLIP provides a clock signal for import or export on the BNC CLOCK connector. This CLIP also allows for individual clock output signal inversion.

Refer to the *NI FlexRIO Adapter Module Support* topic of the *NI FlexRIO Help* for information regarding NI FlexRIO CLIP, configuring the NI 6584 with a socketed CLIP, and a list of available socketed CLIP and provided signals.

How to Use Your NI FlexRIO Documentation Set

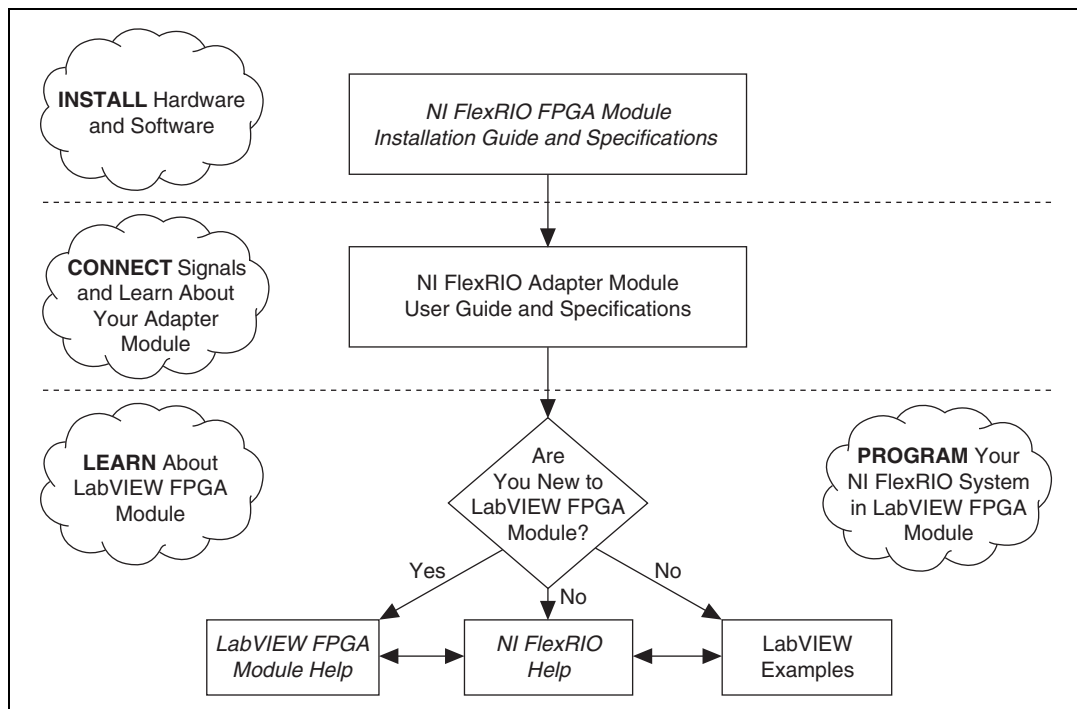Refer to the Figure 9 and Table 4 for information about how to use your NI FlexRIO documentation set.

Figure 9. How to Use Your NI FlexRIO Documentation Set

Table 4. NI FlexRIO Documentation Locations and Descriptions

Document	Location	Description
NI FlexRIO FPGA Module Installation Guide and Specifications[1]	Available in your FPGA module hardware kit and from the Start Menu.	Contains installation instructions for your NI FlexRIO system and specifications for your FPGA module.
NI Adapter Module User Guide and Specifications[1]	Available in your adapter module hardware kit and from the Start Menu.	Contains signal information, examples, and specifications for your adapter module.
LabVIEW FPGA Module Help[1]	Embedded in *LabVIEW Help*.	Contains information about the basic functionality of LabVIEW FPGA Module.
NI FlexRIO Help[1]	Embedded in *LabVIEW FPGA Module Help*.	Contains FPGA module, adapter module, and CLIP configuration information.
LabVIEW Examples	Available in LabVIEW Example Finder.	Contains examples of how to run FPGA VIs and Host VIs on your device.
Other Useful Information on `ni.com`		
`ni.com/ipnet`	Contains LabVIEW FPGA functions and intellectual property to share.	
`ni.com/flexrio`	Contains product information and data sheets for NI FlexRIO devices.	
[1] These documents are also available at `ni.com/manuals`.		

Specifications

Channel Specifications (RS-485/422)

Type of connectors ....1 VHDCI (Connector 0)

Number of RS-485/422 I/O channels ....16 data

Direction control of data channels ....Individual TX and RX enables

Power up state ....Drivers disabled, receivers enabled

I/O compatibility...RS-485/422 (differential)

Maximum input differential voltage ....5 V

Characteristic impedance ....100 Ω differential nominal

Termination impedance..100 Ω differential nominal or open

Maximum data rate ....16 Mbit/s per channel nominal
(requires proper termination on the bus)

Figures 10 and 11 show the block diagrams for the full and half duplex versions of the NI 6584.
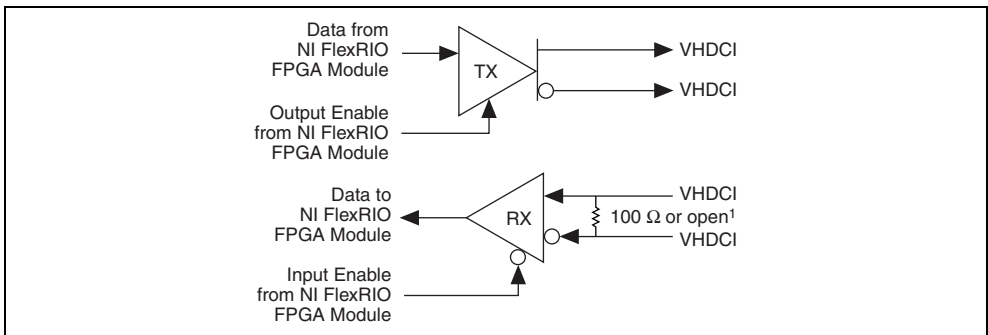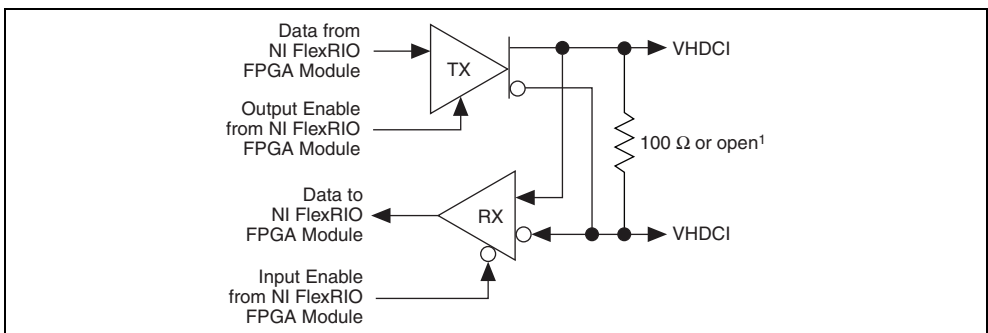

Figure 10. NI 6584 Block Diagram (Full Duplex)

Figure 11. NI 6584 Block Diagram (Half Duplex)

Note The VHDCI connector shown in the preceding figures is labeled *CONNECTOR 0 (Ports 1–8)* on the front panel of the NI 6584.

Channel Specifications (Clock/PFI)

Type of connectors...2 BNC

I/O type ....1 clock, 1 PFI (single-ended)

Direction control of clock/PFI channels ....Per channel

Characteristic impedance ....50 Ω nominal

Output impedance ....50 Ω nominal

Input impedance...50 kΩ nominal

Maximum toggle rate...16 MHz nominal

Input voltage

    High voltage threshold..................................2.5 V

    Low voltage threshold...................................0.8 V

Output voltage (100 µA load)

    High Minimum ....3.0 V

    Low Maximum...0.1 V

Power

Power requirements from the NI FlexRIO FPGA module

+12 V ....100 mA, 1.2 W max

+3.3 V ....500 mA, 1.65 W max

Physical

Dimensions ....13.1 × 2.0 × 12.9 cm (5.2 × 0.8 × 5.1 in.)

Weight ....298 g (10.5 oz)

Front panel connectors...Two BNC connectors and one 68-pin VHDCI
connector

Environmental

The NI 6584 is intended for indoor use only.

Operating environment ....0 °C to 55 °C,
tested in accordance with IEC-60068-2-1 and
IEC-60068-2-2.

Relative humidity range..10% to 90%, noncondensing,
tested in accordance with IEC-60068-2-56.

Altitude ....2,000 m at 25 °C ambient temperature.

Pollution Degree 2

Storage environment

    Ambient temperature range ....–20 °C to 70 °C,
                                                 tested in accordance with IEC-60068-2-1 and
                                               IEC-60068-2-2.

    Relative humidity range ....5% to 95%, noncondensing,
                                               tested in accordance with IEC-60068-2-56.

Note Clean the device with a soft, non-metallic brush. Make sure that the device is completely dry and free from contaminants before returning it to service.

Glossary

Maximum and *minimum* specifications are warranted not to exceed these values within certain operating conditions and include the effects of temperature and uncertainty unless otherwise noted.

Nominal specifications are unwarranted values that are relevant to the use of the product and convey the expected performance of the product.

Safety

This product meets the requirements of the following standards of safety for electrical equipment for measurement, control, and laboratory use:

- IEC 61010-1, EN 61010-1
- UL 61010-1, CSA 61010-1

Note For UL and other safety certifications, refer to the product label or the *Online Product Certification* section.

Electromagnetic Compatibility

This product meets the requirements of the following EMC standards for electrical equipment for measurement, control, and laboratory use:

- EN 61326-1 (IEC 61326-1): Class A emissions; Basic immunity
- EN 55011 (CISPR 11): Group 1, Class A emissions
- AS/NZS CISPR 11: Group 1, Class A emissions
- FCC 47 CFR Part 15B: Class A emissions
- ICES-001: Class A emissions

Note For the standards applied to assess the EMC of this product, refer to the *Online Product Certification* section.

Note For EMC compliance, operate this device with shielded cables.

CE Compliance C E

This product meets the essential requirements of applicable European Directives as follows:

- 2006/95/EC; Low-Voltage Directive (safety)
- 2004/108/EC; Electromagnetic Compatibility Directive (EMC)

Online Product Certification

Refer to the product Declaration of Conformity (DoC) for additional regulatory compliance information. To obtain product certifications and 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.

Environmental Management

National Instruments is committed to designing and manufacturing products in an environmentally responsible manner. NI recognizes that eliminating certain hazardous substances from our products is beneficial not only to the environment but also to NI customers.

For additional environmental information, refer to the *NI and the Environment* Web page at `ni.com/environment`. This page contains the environmental regulations and directives with which NI complies, as well as other environmental information not included in this document.

Waste Electrical and Electronic Equipment (WEEE)

EU Customers At the end of the product life cycle, all products *must* be sent to a WEEE recycling center. For more information about WEEE recycling centers, National Instruments WEEE initiatives, and compliance with WEEE Directive 2002/96/EC on Waste and Electronic Equipment, visit `ni.com/environment/weee`.

电子信息产品污染控制管理办法 （中国 RoHS）

中国客户 National Instruments 符合中国电子信息产品中限制使用某些有害物质指令 (RoHS)。关于 National Instruments 中国 RoHS 合规性信息，请登录 `ni.com/environment/rohs_china`。(For information about China RoHS compliance, go to `ni.com/environment/rohs_china`.)

NI Services

NI 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 MyNI user profile to get personalized access to your services.