

CompactRIO™ Reference and Procedures (FPGA Interface)

June 2008, 370984N-01

Use this book as a reference for information about which [FPGA I/O functions](#), I/O resources, arbitration options, methods, and properties each C Series module supports. This book also includes instructions for using LabVIEW with CompactRIO devices.



To view related topics, click the **Locate** button, shown at left, in the toolbar at the top of this window. The *LabVIEW Help* highlights this topic in the **Contents** tab so you can navigate the related topics.

To comment on National Instruments documentation, refer to the [National Instruments Web site](#).

© 2004–2008 National Instruments Corporation. All rights reserved.

CompactRIO Related Documentation (FPGA Interface)

CompactRIO includes the following documentation.

Help Resources

- [LabVIEW Help](#)—Use this help file to access reference information about C Series modules and instructions for using LabVIEW with CompactRIO devices. Using the **Contents** tab, navigate to **FPGA Module»CompactRIO Reference and Procedures**. You can find help topics for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.
- *Measurement & Automation Explorer (MAX) Help for CompactRIO*—Use this help file to access instructions for configuring the CompactRIO controller and enabling the cRIO-910x Reconfigurable Embedded chassis for use with LabVIEW. After you launch MAX, select **Help»MAX Help** and navigate to **Measurement & Automation Explorer Help»Installed Products»MAX Help for CompactRIO** on the **Contents** tab.

PDF Documents

These documents are available as PDFs in the CompactRIO\manuals directory. The latest versions of these documents are online at ni.com/manuals. You must have Adobe Reader with Search and Accessibility 5.0.5 or later installed to view the PDFs. You must have Adobe Reader with Search and Accessibility 6.x or later installed to [search PDF versions](#) of these manuals. Refer to the [Adobe Systems Incorporated Web site](#) to download Acrobat Reader.

- [CompactRIO R Series Expansion System Installation Instructions](#)—Use these installation instructions to learn how to install the cRIO-9151 R Series Expansion chassis.
- [CompactRIO Reconfigurable Embedded System Installation Instructions](#)—Use these installation instructions to learn how to install the cRIO-910x Reconfigurable Embedded chassis.
- [CompactRIO cRIO-9002/9004 Operating Instructions](#)—Use these operating instructions to learn how to connect the controller to the network and use the features of the controller.
- [CompactRIO cRIO-9012/9014 Operating Instructions and Specifications](#)—Use these operating instructions to learn how to connect the controller to the network and use the features of the controller.
- [CompactRIO cRIO-9052 Operating Instructions](#)—Use these operating instructions to learn how to connect the controller to the network and use the features of the controller.
- [CompactRIO cRIO-9072/9074 Operating Instructions and Specifications](#)—Use these operating instructions to learn how to connect the integrated chassis/controller to the network and use the features of the integrated chassis/controller.
- *Getting Started with CompactRIO and LabVIEW*—Use this tutorial to learn how to develop a CompactRIO application in LabVIEW. While developing the application, you can learn concepts and techniques that you can apply when you develop your own CompactRIO application. This tutorial is available online at ni.com/manuals.
- *NI 9xxx Operating Instructions and Specifications*—Use the operating instructions for the C Series module to learn about

module specifications and how to use the module. For example, use the [NI 9403 Operating Instructions and Specifications](#) to learn about the NI 9403.

Readme Documents

- *NI-RIO Readme*—Use this file to learn important last-minute information about NI-RIO, including installation instructions and descriptions of known issues for installing and using NI-RIO. Access the *NI-RIO Readme* on the NI-RIO installation CD.
- [Software Support for CompactRIO, CompactDAQ, and R Series Devices](#)—Use this NI Developer Zone document to determine what versions of the NI-RIO software, the LabVIEW Real-Time Module, and the LabVIEW FPGA Module you need for the C Series modules, CompactRIO chassis, CompactRIO controllers, Single-Board RIO devices, and R Series devices you are using.

Searching PDF Versions of CompactRIO Manuals (FPGA Interface)

Use Adobe Reader with Search and Accessibility 6.x or later to search PDF versions of all the [CompactRIO manuals](#). Refer to the [Adobe Systems Incorporated Web site](#) to download Acrobat Reader.

Complete the following steps to search all the PDF versions of CompactRIO manuals.

1. In Adobe Reader, select **Edit»Search** to display the **Search PDF** window.
2. Enter a word or phrase in the **What word or phrase would you like to search for** text box.
3. Click the **All PDF Documents in** button and select **Browse for Location** from the drop-down list. The **Browse for Folder** dialog box appears.
 - a. Navigate to the CompactRIO\manuals directory.
 - b. Click the **OK** button to close the dialog box and return to the **Search PDF** window.
4. Click the **Search** button.

Refer to the Adobe Reader Help for more information about searching all the PDF documents in a directory for a word or phrase.

Using CompactRIO and Single-Board RIO (FPGA Interface)

Use this book as a reference for information about using LabVIEW with CompactRIO and Single-Board RIO (sbRIO) devices.



Note C Series modules can take up to 2 seconds to initialize after you reset the FPGA VI.



To view related topics, click the **Locate** button, shown at left, in the toolbar at the top of this window. The *LabVIEW Help* highlights this topic in the **Contents** tab so you can navigate the related topics.

C Series Module IDs (FPGA Interface)

You can use the Module ID property with the [FPGA I/O Property Node](#) to read the IDs associated with the C Series modules.

By ID

ID	Module
0x709C	NI 9435
0x709D	NI 9411
0x709E	NI 9423
0x709F	NI 9421 with screw terminal
0x70A0	NI 9474
0x70A1	NI 9472 with screw terminal
0x70A2	NI 9481
0x70A3	NI 9211
0x70A4	NI 9201
0x70A5	NI 9221
0x70A6	NI 9215 with screw terminal
0x70A7	NI 9263 with screw terminal
0x70A8	NI 9233
0x7129	NI 9203
0x712A	NI 9205 with DSUB
0x712B	NI 9217
0x712C	NI 9265
0x712E	NI 9421 with DSUB
0x712F	NI 9425
0x7130	NI 9401
0x7131	NI 9403
0x7132	NI 9472 with DSUB
0x7133	NI 9476
0x7134	NI 9802
0x7135	NI 9215 with BNC
0x714F	NI 9853
0x71C2	NI 9239 with screw terminal

0x71C3	NI 9237 with RJ-50
0x71CA	NI 9422
0x71CB	NI 9477
0x71EA	NI 9205 with spring terminal
0x71EB	NI 9206
0x71ED	NI 9505
0x71F3	NI 9852
0x71F6	NI 9485
0x72B5	NI 9234
0x72F6	NI 9264
0x72FD	NI 9229 with screw terminal
0x7305	NI 9870
0x7306	NI 9871
0x730C	NI 9219 with spring terminal
0x731A	NI 9478
0x7328	NI 9402
0x732A	NI 9235
0x732B	NI 9236
0x7367	NI 9239 with BNC
0x7368	NI 9229 with BNC
0x7369	NI 9263 with BNC
0x736A	NI 9426
0x736F	NI 9219 with LEMO
0x7377	NI 9475
0x7383	NI 9225
0x73A3	NI 9213
0x73A4	NI 9237 with DSUB

By Module

Module	ID
NI 9201	0x70A4
NI 9203	0x7129
NI 9205 with DSUB	0x712A
NI 9205 with spring terminal	0x71EA
NI 9206	0x71EB
NI 9211	0x70A3
NI 9213	0x73A3
NI 9215 with BNC	0x7135
NI 9215 with screw terminal	0x70A6
NI 9217	0x712B
NI 9219 with LEMO	0x736F
NI 9219 with spring terminal	0x730C
NI 9221	0x70A5
NI 9225	0x7383
NI 9229 with BNC	0x7368
NI 9229 with screw terminal	0x72FD
NI 9233	0x70A8
NI 9234	0x72B5
NI 9235	0x732A
NI 9236	0x732B
NI 9237 with DSUB	0x73A4
NI 9237 with RJ-50	0x71C3
NI 9239 with BNC	0x7367
NI 9239 with screw terminal	0x71C2
NI 9263 with BNC	0x7369
NI 9263 with screw terminal	0x70A7
NI 9264	0x72F6

<u>NI 9265</u>	0x712C
<u>NI 9401</u>	0x7130
<u>NI 9402</u>	0x7328
<u>NI 9403</u>	0x7131
<u>NI 9411</u>	0x709D
<u>NI 9421 with DSUB</u>	0x712E
<u>NI 9421 with screw terminal</u>	0x709F
<u>NI 9422</u>	0x71CA
<u>NI 9423</u>	0x709E
<u>NI 9425</u>	0x712F
<u>NI 9426</u>	0x736A
<u>NI 9435</u>	0x709C
<u>NI 9472 with DSUB</u>	0x7132
<u>NI 9472 with screw terminal</u>	0x70A1
<u>NI 9474</u>	0x70A0
<u>NI 9475</u>	0x7377
<u>NI 9476</u>	0x7133
<u>NI 9477</u>	0x71CB
<u>NI 9478</u>	0x731A
<u>NI 9481</u>	0x70A2
<u>NI 9485</u>	0x71F6
<u>NI 9505</u>	0x71ED
<u>NI 9802</u>	0x7134
<u>NI 9852</u>	0x71F3
<u>NI 9853</u>	0x714F
<u>NI 9870</u>	0x7305
<u>NI 9871</u>	0x7306

CompactRIO Discovery Status Dialog Box (FPGA Interface)

When you add a CompactRIO controller or chassis or a Single-Board RIO device to a LabVIEW project, LabVIEW finds all C Series modules in the new system. This dialog box appears when LabVIEW returns errors while looking for C Series modules installed in the system. The dialog box lists any errors that occur during the discovery process. The **Description** field of the dialog box contains a detailed description of each type of error condition.

CompactRIO Error Codes (FPGA Interface)

The [FPGA I/O functions](#) and [Invoke Method](#) function can return the following error codes for CompactRIO.

Code	Description
-65580	The FPGA personality running on the RIO target does not have enough specialty digital resources to support this module.
-65537	The module that was detected is different than the module that was expected. Make sure the slot the module is configured for in software matches the physical location of the module.
-65407	Too many specialty digital slots. In Scan Interface mode, you can configure only two slots for specialty digital I/O. If you want more specialty digital slots, add an FPGA target under the chassis to put the chassis in LabVIEW FPGA interface mode.
65000	Unable to mount drive. The given device is either not present or not recognizable as a mountable device.
65001	No partitions found. The partition table on the device is corrupt or the device has zero partitions on it.
65002	Invalid drive handle. The given handle does not represent an active mounted drive.
65003	Drive already mounted. The given device is already mounted as a drive.
65004	The channel, slot, or connector number you wired to the method input is invalid. Change the method input to match the configuration of the CompactRIO system.
65005	The C Series module at the specified location does not support TEDS or TEDS access is not enabled for the module.
65006	Communication with the module timed out. The module is busy performing another action or LabVIEW is unable to communicate with the module.
65007	No TEDS sensor was detected on the specified channel. Make sure that the C Series module and sensor are properly connected. Make sure the specified location matches the sensor location.

65008	CompactRIO does not support the TEDS sensor connected to this channel.
65009	The PXI trigger that you have selected to reserve or unreserve is invalid. Valid PXI triggers are 0 through 7, inclusive.
65200	DIO Line Access Conflict. Invalid module configuration. A single physical DIO line cannot be accessed by multiple types of output nodes when the Number of Synchronizing Registers=0 and any of the output accesses is in a Single-Cycle Timed Loop. Either change the number of Synchronizing Registers to 1 in the module's properties dialog, or access the resource exclusively as a port or an individual line, not both.
65201	Duplicate Terminals In The Same Node. An FPGA I/O Node has duplicate terminals. Delete the duplicate terminal from the I/O Node.
65202	Digital Resource Access Conflict. The digital I/O resource cannot be accessed in a Single-Cycle Timed Loop from both a Digital Output function and a Digital Port Output function. If you need to access this resource in a Single-Cycle Timed Loop, please exclusively use just the Digital Output function or Digital Port Output function, not both.
65203	Module Timebase Configuration Error. LabVIEW detected an invalid configuration for an FPGA I/O Node that contains channels from a module with a configurable timebase. If channels of multiple modules with a configurable timebase are in the same FPGA I/O Node, make sure you configure the modules to share the same timebase. Use the C Series Module Properties dialog box to configure the module timebase. Refer to the LabVIEW Help for information about how to synchronize multiple C Series modules.
65204	Digital Resource Access Conflict. The digital I/O resource cannot be accessed from both a Digital Output function and a Digital Port Output function if the Never Arbitrate option is used. Please change the arbitration of the Digital Line and/or Digital Port to something other than Never Arbitrate, or exclusively use just the Digital Output function or Digital Port Output function, not both.

65205	Invalid C Series Module Configuration. Possible reasons for the invalid configuration include that the master timebase source module is unable to be identified, the master timebase source module is not configured to export its timebase, or the master timebase source module is not a valid module type. Use the C Series Module Properties dialog box to configure the module timebase. Refer to the LabVIEW Help for information about how to synchronize multiple C Series modules.
65206	Invalid top-level clock. You must use a top-level clock of 40 MHz when using this module. To change the top-level clock, right click on your FPGA target in the LabVIEW Project Explorer and select properties. From the Top-Level Clock category, choose a 40 MHz clock.
65207	The digital output resource cannot be accessed from both a Digital Output function and a Digital Port Output function. Please exclusively use just the Digital Output function or Digital Port Output function, not both.
65208	The cRIO-9151 R Series Expansion chassis no longer supports synchronizing multiple NI 9225/9229/923x modules. Use the C Series Module Properties dialog box to set the master timebase source of the slave module(s) to the onboard clock. Right-click the module in the project and select Properties to display the C Series Module Properties dialog box. Contact National Instruments technical support with questions or concerns.
65209	You cannot write to the Sleep channel if you are using the Scan Interface with any modules in the system.
65400	The FPGA target is either running an FPGA VI or has loaded an FPGA VI.
65401	One or more discovered C Series modules are not supported by the current versions of LabVIEW and NI-RIO.
65402	An internal software error in NI-RIO has occurred. Please contact National Instruments technical support at ni.com/support .
65403	An unexpected error occurred when Discovering C Series Modules. Make sure the LabVIEW Project is set up properly.

65404	The controller you selected has an unconfigured (0.0.0.0) IP address. If the controller is online, configure it in Measurement & Automation Explorer (MAX), then make sure the IP address in LabVIEW matches the IP address in MAX. If it is offline, you cannot discover connected targets and devices, but you can add new, offline targets and devices.
65405	Module not found. The module whose configuration you deployed is not present in the chassis.
65406	Different module. The module whose configuration you deployed does not match the model currently in the chassis.
65407	Too many specialty digital slots. In Scan Interface mode, you can configure only two slots for specialty digital I/O. If you want more specialty digital slots, add an FPGA target under the chassis to put the chassis in LabVIEW FPGA Interface mode.
65536	Unable to communicate with the module. Reinsert the module and check connections.
65537	The module that was detected is different than the module that was expected. Make sure the slot the module is configured for in software matches the physical location of the module.
65538	The operation failed to complete in time. Make sure the module is not busy and the system is configured properly.
65539	The input function missed one or more data points. Make sure the loop can execute as fast as the module data rate.
65540	The I/O Resource is not in communication mode. You must start communication mode before you can perform this operation.
65541	The I/O Resource is in communication mode. You must stop communication mode before you can perform this operation.
65542	One or more channels have detected an open current loop. Check the module connections.
65543	The power supply voltage level is out of range. Check the supply voltage and the module connections.
65544	One or more channels are in overcurrent protection mode. The device connected to the channel is passing more current than is allowed through the channel. Check for possible shorts or external device failure.

65545	An input parameter, or a combination of parameters, is invalid.
65546	Your application uses a feature that is not supported by your C Series hardware.
65547	Too many CAN bus error frames are detected. Please refer to the description of the 'Error Terminals' for more information.
65548	One or more channels are in overcurrent or overvoltage protection mode. Check the terminals for any fault condition that could be causing an out-of-range voltage or current on the channels.
65549	A general or undefined error has occurred. Verify that the card is inserted properly and that the door is closed. If the error occurs again, run CHKDSK on the card.
65550	A problem was found in the file system. Remove the SD card and run CHKDSK.
65551	The SD card is in use by RT. Try again after RT unmounts the card.
65552	The SD card is not ready. Verify that the card is inserted properly and that the door is closed.
65553	The SD card door was opened while a file on the card was open.
65554	The specified file does not exist on the card.
65555	The Open method tried to open a new file for writing, but a file with the same name already exists on the disk.
65556	A Read or Write method tried to access a file that was not opened in the required mode.
65558	A Close method tried to close a file that was not open.
65559	A method tried to open a file on the SD card when a file was already open. This device supports only one file open at a time. This error is also returned when an illegal attempt to call the Delete File or Get File Size method is made when a file is open on the card.
65560	An attempt to allocate storage failed because the file system is full.

65561	Attempted to exceed the limit of 512 root directory entries.
65562	A Read method tried to read beyond the end of a file. This may have occurred when the method read the end of a file using a U16 or U32 data type when the actual number of bytes in the file (as reported by the directory) was not an integer multiple of the number of bytes in the read data type.
65563	A problem was found with the format of the SD card. Verify that the SD card is formatted with a valid FAT16 file system.
65577	An open thermocouple was detected on at least one channel. Check the module connections.
65578	The common-mode voltage is outside of acceptable limits on at least one channel. Check the terminals for any fault condition that could be causing an out-of-range voltage on the channels.

CompactRIO Interface Methods (FPGA Interface)

In addition to the methods described in the [Invoke Method](#) topic, you can use the following method with the Invoke Method function for CompactRIO devices.

- **Read TEDS**—Reads TEDS information from a C Series module that has TEDS support enabled.
 - **Connector** specifies the connector to which the cRIO-9151 R Series Expansion chassis is connected. This input is available only if the [Open FPGA VI Reference](#) function is configured for an R Series device. Valid values are 0 through 3 for the NI 781xR, 1 or 2 for the NI 7831R and NI 7833R, and 1 for the NI 7830R.
 - **Slot** specifies the chassis slot of the C Series module from which you want to read TEDS information. If the slot does not have a C Series module that supports this feature, LabVIEW returns an error. Valid values are 1 through N , where N is the number of slots in the chassis.
 - **Channel** specifies the channel on the module from which you want to read TEDS information. If a sensor is not connected to the channel, the Invoke Method function returns an error. Valid values are 0 through 3 for the NI 9219, NI 9233, NI 9234, and NI 9237.
 - **TEDS Binary** represents the TEDS information of the sensor connected to the specified channel in an array of bits. You can use the TEDS Toolkit to parse this data.
 - **v0.9 (TEDS)** determines whether the sensor stored in the TEDS is in the released IEEE 1451.4 standard format or in the legacy version (v0.9) format. **v0.9** returns TRUE if the TEDS is in the legacy format. The TEDS Toolkit VIs require this output to properly parse information.
- **Mount SD Card**—Prepares the file system of an [NI 9802](#) SD Card so you can access the files using the [File I/O VIs and functions](#). You must use the Unmount SD Card method to unmount the file system after accessing files.



Note The FPGA VI must be running for the Real-Time VI to access the Mount SD Card and Unmount SD Card methods. You cannot access an SD Card from the Real-Time VI and the FPGA VI at the same time.



Module specifies the chassis slot of the NI 9802 module. If the slot does not have an NI 9802 module, LabVIEW returns an error. Valid values are 1 through N , where N is the number of slots in the chassis.



Card Slot specifies the card slot of the SD Card which you want to access. Valid values are 0 or 1.



Drive Handle returns a handle to the mounted drive. Wire this output to the **Drive Handle** input of the Unmount SD Card method.



Drive Letter returns the base path of the SD Card file system you mounted.



Note The cRIO-9002/9004 controller does not support drive letter access.

Refer to the NI 9802 RT Access VI in the `labview\examples\CompactRIO\Module Specific\NI 9802\NI 9802 RT Access` directory for an example of using this method.

■ Open example

- **Unmount SD Card**—Unmounts the SD Card file system you mounted into a drive. This method attempts to execute even if the **error in** parameter contains an error.



Drive Handle specifies the drive to unmount.

CompactRIO Reconfigurable Embedded Chassis (FPGA Interface)

A user-reconfigurable [FPGA](#) controls the digital and analog I/O lines on the cRIO-910x Reconfigurable Embedded chassis of the CompactRIO Embedded system. The FPGA chip is embedded within the CompactRIO chassis and is connected to the C Series I/O modules in a star topology, providing direct access to each module for precise control and flexibility in timing, triggering, and synchronization. You can [configure the CompactRIO system](#) and change the functionality of the FPGA on the CompactRIO chassis in LabVIEW using the LabVIEW FPGA Module and NI-RIO to [create](#) and [download](#) a custom VI to the FPGA.

Configuring a Project for a CompactRIO R Series Expansion System (FPGA Interface)

Complete the following steps to add a cRIO-9151 R Series Expansion chassis system to a new or existing project.



Note If you have not yet created a LabVIEW project, you also can use the [FPGA Project Wizard](#) to create a project for the CompactRIO system.

Configuring a Project with Connected Hardware

Complete the following steps to configure the project if you have hardware installed. The R Series Expansion chassis must have C Series modules installed and must be connected to an R Series device installed in the computer. Refer to the [CompactRIO R Series Expansion System Installation Instructions](#) for information about installing the chassis.

1. [Create a new project](#) or open an existing project.
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 the appropriate R Series device under **R Series** and click the **OK** button. LabVIEW adds an FPGA target item for the R Series device to the project.
4. Right-click the **FPGA Target** for the R Series device in the **Project Explorer** window and select **New»R Series Expansion Chassis** from the shortcut menu to display the **New R Series Expansion Chassis** dialog box.
5. Select the connector to which the R Series Expansion chassis is connected in the **Location** pull-down menu.
6. Check the **Discover C Series modules** checkbox and click the **OK** button. LabVIEW adds items for the chassis and all installed C Series modules to the project. LabVIEW also adds FPGA I/O items to the project for all installed C Series module I/O.
7. Right-click a module item in the **Project Explorer** window and select **Properties** from the shortcut menu to configure module-specific settings in the **C Series Module Properties** dialog box. Some modules do not have any settings to configure other than the module name and chassis slot location. Click the **Help** button on the **C Series Module Properties** dialog box for information about the module settings.

Configuring a Project with Offline Hardware

Complete the following steps to configure the project if you do not have hardware installed.

1. [Create a new project](#) or open an existing project.
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. Click the **New target or device** radio button, select the appropriate R Series device under **R Series**, and click the **OK** button. LabVIEW adds an FPGA target item for the R Series device to the project.
4. Right-click the **FPGA Target** for the R Series device in the **Project Explorer** window and select **New»R Series Expansion Chassis** from the shortcut menu to display the **New R Series Expansion Chassis** dialog box.
5. Click the **OK** button. LabVIEW adds a chassis item to the project.
6. Right-click the chassis item in the **Project Explorer** window and select **New»C Series Modules** from the shortcut menu to display the **Add Targets and Devices** dialog box.
7. Click the **New target or device** radio button, select **C Series Module**, and click the **OK** button to display the **New C Series Module** dialog box.
8. Select the appropriate C Series module from the **Module Type** pull-down menu and click the **OK** button. LabVIEW adds a module item and FPGA I/O items for the module I/O to the project.
9. Repeat steps 6 through 8 to add additional C Series modules to the project.
10. Right-click a module item in the **Project Explorer** window and select **Properties** from the shortcut menu to configure module-specific settings in the **C Series Module Properties** dialog box. Some modules do not have any settings to configure other than the module name and chassis slot location. Click the **Help** button on the **C Series Module Properties** dialog box for information about the module settings.

Configuring a Project for a CompactRIO Reconfigurable or Integrated System (FPGA Interface)

Complete the following steps to add a cRIO-910x Reconfigurable Embedded system or cRIO-9072/9074 Integrated controller and chassis system to a new or existing project.



Note If you have not yet created a LabVIEW project, you also can use the [FPGA Project Wizard](#) to create a project for the CompactRIO system.

Configuring a Project with Connected Hardware

Complete the following steps to configure the project if you have hardware installed. The controller must be attached to a chassis with C Series modules installed, connected to the same subnet as the host computer, and powered on. Refer to the [controller operating instructions](#) for information about installing the controller on a chassis, connecting the controller to a network, and wiring power to the controller. The controller also must be configured in Measurement & Automation Explorer (MAX). Refer to the [Measurement & Automation Explorer \(MAX\) Help for CompactRIO](#) for information about configuring the controller.

1. [Create a new project](#) or open an existing project.
2. Right-click the project root 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 the appropriate controller under **Real-Time CompactRIO** and click the **OK** button. If you are using a controller and chassis that are supported in Scan Interface mode, the [Select Programming Mode](#) dialog box appears.
4. If the **Select Programming Mode** dialog box appears, select **LabVIEW FPGA Interface** and click the **Continue** button to put the system into LabVIEW FPGA Interface mode.
5. Click the **Discover** button in the **Discover C Series Modules?** dialog box that appears. LabVIEW adds items for the controller, the chassis, the FPGA target, and all installed C Series modules to the project. LabVIEW also adds FPGA I/O items to the project for all installed C Series module I/O.
6. Right-click a module item in the **Project Explorer** window and select **Properties** from the shortcut menu to configure module-specific settings in the **C Series Module Properties** dialog box. Some modules do not have any settings to configure other than the module name and chassis slot location. Click the **Help** button on the **C Series Module Properties** dialog box for information about the module settings.

Configuring a Project with Offline Hardware

Complete the following steps to configure the project if you do not have hardware installed.

1. [Create a new project](#) or open an existing project.
2. Right-click the project root 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. Click the **New target or device** radio button, select the appropriate controller under **Real-Time CompactRIO**, and click the **OK** button. LabVIEW adds an RT target item for the controller to the project.
4. Right-click the **RT CompactRIO Target** in the **Project Explorer** window and select **New»Targets and Devices** from the shortcut menu to display the **Add Targets and Devices** dialog box.
5. Click the **New target or device** radio button, select the appropriate chassis under **CompactRIO Chassis**, and click the **OK** button. LabVIEW adds a chassis item to the project.
 **Note** If you selected the cRIO-9072/9074 Integrated controller in step 4, you must select the matching cRIO-9072/9074 Integrated chassis in this step.
6. Right-click the chassis item in the **Project Explorer** window and select **New»FPGA Target** from the shortcut menu. If the chassis type is supported in Scan Interface mode, the **Deploy CompactRIO Chassis Settings?** dialog box appears. Click the **No** button to return to the project. LabVIEW adds an FPGA target item for the chassis to the project and puts the system into [LabVIEW FPGA Interface mode](#).
7. Right-click the **FPGA Target** in the **Project Explorer** window and select **New»C Series Modules** from the shortcut menu to display the **Add Targets and Devices** dialog box.
8. Click the **New target or device** radio button, select **C Series Module**, and click the **OK** button to display the **New C Series Module** dialog box.
9. Select the appropriate C Series module from the **Module Type** pull-down menu and click the **OK** button. LabVIEW adds a

module item and FPGA I/O items for the module I/O to the project.

10. Repeat steps 7 through 9 to add additional C Series modules to the project.
11. Right-click a module item in the **Project Explorer** window and select **Properties** from the shortcut menu to configure module-specific settings in the **C Series Module Properties** dialog box. Some modules do not have any settings to configure other than the module name and chassis slot location. Click the **Help** button on the **C Series Module Properties** dialog box for information about the module settings.

Configuring a Project for a Single-Board RIO Reconfigurable System (FPGA Interface)

Complete the following steps to add a [Single-Board RIO](#) (sbRIO) Reconfigurable Embedded system to a new or existing project.



Note If you have not yet created a LabVIEW project, you also can use the [FPGA Project Wizard](#) to create a project for the sbRIO system.

Configuring a Project with Connected Hardware

Complete the following steps to configure the project if you have hardware installed. The sbRIO device must be connected to the same subnet as the host computer and powered on. Refer to the sbRIO device manual for information about connecting the device to a network and wiring power to the device. The sbRIO device also must be configured in Measurement & Automation Explorer (MAX).

1. [Create a new project](#) or open an existing project.
2. Right-click the project root 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 the appropriate sbRIO device under **Real-Time Single-Board RIO** and click the **OK** button.
4. Click the **Discover** button in the **Discover C Series Modules?** dialog box that appears. LabVIEW adds items for the sbRIO device, the FPGA target, and all onboard and installed C Series modules to the project. LabVIEW also adds FPGA I/O items to the project for all onboard and installed C Series module I/O.
5. Right-click a module item in the **Project Explorer** window and select **Properties** from the shortcut menu to configure module-specific settings in the **C Series Module Properties** dialog box. Some modules do not have any settings to configure other than the module name and chassis slot location. Click the **Help** button on the **C Series Module Properties** dialog box for information about the module settings.

Configuring a Project with Offline Hardware

Complete the following steps to configure the project if you do not have hardware installed.

1. [Create a new project](#) or open an existing project.
2. Right-click the project root 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. Click the **New target or device** radio button, select the appropriate sbRIO system under **Real-Time Single-Board RIO**, and click the **OK** button. LabVIEW adds an RT target item for the sbRIO device to the project.
4. Right-click **RT Single-Board RIO** in the **Project Explorer** window and select **New»Targets and Devices** from the shortcut menu to display the **Add Targets and Devices** dialog box.
5. Click the **New target or device** radio button, select the appropriate sbRIO system under **Single-Board RIO**, and click the **OK** button. LabVIEW adds an FPGA target item for the sbRIO device and FPGA I/O items for all onboard C Series module I/O to the project.
 **Note** The system you select in this step must be the same system you selected in step 3.
6. Right-click the **FPGA Target** for the sbRIO device in the **Project Explorer** window and select **New»C Series Modules** from the shortcut menu to display the **Add Targets and Devices** dialog box.
7. Click the **New target or device** radio button, select **C Series Module**, and click the **OK** button to display the **New C Series Module** dialog box.
8. Select the appropriate C Series module from the **Module Type** pull-down menu and click the **OK** button. LabVIEW adds a module item and FPGA I/O items for the module I/O to the project.
9. Repeat steps 6 through 8 to add additional C Series modules to the project.
10. Right-click a module item in the **Project Explorer** window and

select **Properties** from the shortcut menu to configure module-specific settings in the **C Series Module Properties** dialog box. Some modules do not have any settings to configure other than the module name and chassis slot location. Click the **Help** button on the **C Series Module Properties** dialog box for information about the module settings.

Converting the Temperature of a CompactRIO Chassis or a Single-Board RIO Device (FPGA Interface)

The [FPGA I/O Node](#) returns binary values for the [chassis temperature](#). You can convert these values into meaningful engineering units. To ensure that the FPGA VI is as efficient as possible, convert the values in the host VI.

Using a VI to Convert Values

Refer to the 910x Temperature Monitor (FPGA) VI in the labview\examples\CompactRIO\Chassis Specific\910x Temperature Monitor\910x Temperature Monitor.lvproj for an example of converting binary values. You can use the 910x Temperature Monitor (FPGA) VI as a subVI in the host VI to convert the values.

- Open example

Using an Equation to Convert Values

You can use the following equation in the host VI to convert binary values to temperature:

$$\text{Temperature} = \text{Binary Value} \times 0.25 \text{ } ^\circ\text{C}$$

where *Binary Value* is the value returned by the FPGA I/O Node.

Downloading to Flash Memory (FPGA Interface)

You can download a VI or bitfile to the flash memory on the FPGA target using one of the following methods:

- Right-click the VI you want to download to flash memory on the FPGA target, select **Download VI to Flash Memory** from the shortcut menu, and click the **Yes** button on the **Confirm Downloading VI to Flash** dialog box that appears.
- Right-click the FPGA target to which you want to download a bitfile in the **Project Explorer** window in LabVIEW and select **RIO Device Setup** from the shortcut menu to display the **RIO Device Setup** dialog box. Refer to the *RIO Device Setup Help*, available by clicking the **Help** button on the **RIO Device Setup** dialog box, for more information about the RIO Device Setup utility.
- Select **Start»All Programs»National Instruments»NI-RIO»RIO Device Setup** to display the **RIO Device Setup** dialog box. When you display the dialog box in this manner, you must select the FPGA target to which you want to download a bitfile from the **Device** pull-down menu. Refer to the *RIO Device Setup Help*, available by clicking the **Help** button on the **RIO Device Setup** dialog box, for more information about the RIO Device Setup utility.

Enabling Sleep Mode on a CompactRIO Chassis or a Single-Board RIO Device (FPGA Interface)

The cRIO-910x Reconfigurable Embedded chassis, the cRIO-9072/9074 Integrated controller and chassis, and the [Single-Board RIO](#) (sbRIO) devices have a sleep mode line. When you enable sleep mode, installed and onboard C Series I/O modules in the chassis or device enter a low-power state. Typically, you cannot communicate with modules when they are in sleep mode. If you try to communicate with a module when it is in sleep mode, LabVIEW returns a module communications error from the [FPGA I/O functions](#). Sleep mode does not affect the CompactRIO chassis or controller or the sbRIO device.



Note The cRIO-9151 R Series Expansion chassis does not have a sleep mode line.

Complete the following steps to enable sleep mode on a cRIO-910x Reconfigurable Embedded chassis or an sbRIO device.

1. [Configure](#) the CompactRIO or sbRIO system.
2. Make sure the **Sleep** I/O item is [added](#) to the system.
3. Right-click the FPGA target and select **New»VI** from the shortcut menu to add a new VI to the system.
4. Place an [FPGA I/O Node](#) on the block diagram of the VI.
5. Click the element section of the FPGA I/O Node and select **Chassis I/O (or Onboard I/O)»Sleep** from the shortcut menu.
6. Right-click the **Sleep** input of the FPGA I/O Node and select **Create»Control** from the shortcut menu. When **Sleep** is TRUE, the system enters sleep mode. When **Sleep** is FALSE, the system enters active mode. The default state of the system is active mode.

Confirming the State of the Sleep Channel

Complete the following steps to read the state of the Sleep channel.

1. Place another FPGA I/O Node function on the block diagram. This FPGA I/O Node must execute after the FPGA I/O Node you added above.
2. Click the element section of the FPGA I/O Node and select **Chassis I/O (or Onboard I/O)»Sleep** from the shortcut menu.
3. Right-click the FPGA I/O Node and select **Change to Read** from the shortcut menu.
4. Right-click the **Sleep** output of the FPGA I/O Node and select **Create»Indicator** from the shortcut menu.

When you run the VI, the FPGA I/O Node reads the state of the Sleep channel.

New C Series Module Dialog Box (FPGA Interface)

Right-click the chassis in the **Project Explorer** window and select **New»C Series Modules** from the shortcut menu to display the [Add Targets and Devices](#) dialog box. Select **New target or device**, select **C Series Module**, and click the **OK** button to display the **New C Series Module** dialog box.

Use this dialog box to select a module name, the type of module, and the chassis slot in which the module is installed.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module you want to add to the chassis in the **Project Explorer** window.
- **Location**—Specifies a slot in the chassis for the C Series module.

New R Series Expansion Chassis Dialog Box (FPGA Interface)

This dialog box appears when you right-click an R Series device in the **Project Explorer** window and select **New»R Series Expansion Chassis** from the shortcut menu. Use this dialog box to enter a name for the cRIO-9151 R Series Expansion chassis and select the connector to which it is installed.

- **Name**—You can use this text box to enter a name for the chassis.
- **Location**—You can use this pull-down menu to select the connector to which the chassis is connected.
- **Discover C Series modules**—Place a checkmark in this checkbox if you want LabVIEW to find all C Series modules installed in the chassis. Do not check this checkbox if you do not have an expansion chassis installed to the connector selected in the **Location** pull-down menu.

NI-RIO FPGA Communications Framework Error Codes (FPGA Interface)

The [FPGA Interface functions](#) can return the following error codes for NI-RIO Devices.

Code	Description
-63198	The system has run out of resources. Close a session and retry the operation.
-63197	An invalid attribute value has been specified.
-63196	An invalid attribute has been specified.
-63195	The handle for device communication is invalid or has been closed. Restart the application.
-63194	The NI-RIO software on the remote system is not compatible with the local NI-RIO software. Upgrade the NI-RIO software on the remote system.
-63193	The requested feature is not supported.
-63192	Either the supplied resource name is invalid as a RIO resource name, or the device was not found. Use MAX to find the proper resource name for the intended device.
-63189	The supplied search pattern is invalid.
-63188	The operation is no longer supported.
-63187	This remote system does not support connections to other remote systems.
-63183	An invalid port was specified. The RIO server port must be between 0 and 65535, where 0 indicates a dynamically assigned port. Port 3580 is reserved and cannot be used.
-63182	An invalid device access setting was specified. RIO device access patterns may contain only alphanumerics, '-', '_', ':', and '*'.
-63181	The supplied alias was not found.
-63180	An invalid alias was specified. RIO aliases may contain only alphanumerics, '-', and '_'.
-63150	An unspecified hardware failure has occurred. The operation

	could not be completed.
-63082	The operation could not complete because another session is accessing the FIFO. Close the other session and retry.
-63081	The caller did not allocate a memory buffer.
-63080	The allocated buffer is too small.
-63073	The specified event did not occur within the specified time period, in milliseconds. Extend the time period, or ignore if the result was expected.
-63072	The specified RIO event has not been enabled for this session. Attempting a Wait on IRQ after an Abort causes this error.
-63071	The specified RIO event has already been enabled for this session.
-63070	The specified event type is invalid.
-63052	Trigger lines are not supported or enabled. For PXI, identify the controller and chassis using MAX.
-63051	The specified trigger line is not reserved in the current session.
-63050	The specified trigger line is already reserved. Consult the MAX Trigger settings or the trigger reservations for each device within the system.
-63043	The session is invalid. The target may have reset or been rebooted. Check the network connection and retry the operation.
-63042	A fault on the network caused the operation to fail.
-63041	The connection to the remote device has been lost due to an error on the remote device. Retry the operation. If the remote device continues to report this error, check its power supply and look for diagnostic messages on the console.
-63040	A connection could not be established to the specified remote device. Ensure that the device is on and accessible over the network, that NI-RIO software is installed, and that the RIO server is running and properly configured. For NI-RIO 2.3 and forward, refer to Software»NI-RIO»NI-RIO Settings under the system in MAX. Prior to NI-RIO 2.3, refer to Software»NI-VISA»VISA Options under the system in MAX.

-63033	Access to the remote system was denied. Use MAX to check the Remote Device Access settings under Software»NI-RIO»NI-RIO Settings on the remote system.
-63031	The operation could not be completed because another session is accessing the device. Close all other sessions and retry.
-63030	Operation failed due to device reconfiguration. Multiple sessions to FPGA devices are not supported. Close the other session and retry this operation. This error code can occur only with LabVIEW 8.2 and earlier versions. The operation could not complete because another session has reconfigured the device.
-63001	DMA from host to FPGA target is not supported for this remote system. Use another method for I/O or change the controller associated with the FPGA target.
63033	Access to the remote system was denied. Use MAX to check the Remote Device Access settings under Software»NI-RIO»NI-RIO Settings on the remote system.
63073	The specified event did not occur within the specified time period, in milliseconds. Extend the time period, or ignore if the result was expected.
63186	The number of open RIO sessions exceeds the recommended limit. For optimal performance, close RIO sessions when you no longer need them.
63188	The operation is no longer supported.
63194	The target software is older than, but compatible with, the host. Upgrade target software to get full functionality.

Programmatically Resetting the CompactRIO or Single-Board RIO System (FPGA Interface)

The CompactRIO cRIO-910x and cRIO-9072/9074 chassis and the [Single-Board RIO](#) (sbRIO) devices have a digital output line you can use to programmatically reset the system.

Complete the following steps to programmatically reset the system.

1. [Configure](#) the CompactRIO or sbRIO system.
2. Make sure the **System Reset** I/O item is [added](#) to the system.
3. Right-click the FPGA target in the **Project Explorer** window and select **New»VI** from the shortcut menu to add a new FPGA VI to the system.
4. Place an [FPGA I/O Node](#) on the block diagram of the FPGA VI.
5. Click the element section of the FPGA I/O Node and select **Chassis I/O (or Onboard I/O)»System Reset** from the shortcut menu.
6. Right-click the **System Reset** input of the FPGA I/O Node and select **Create»Control** from the shortcut menu. When **System Reset** is TRUE, the system resets.

The system FPGA deasserts the **System Reset** line during the system reset. You do not need to wire a FALSE Boolean value to the **System Reset** input or ensure that the **System Reset** input is TRUE for a minimum amount of time.



Caution If the FPGA VI uses the **System Reset** output, and you configure the system to load the FPGA VI when there is a software or hardware reset, make sure that the system is not set up in a way in which the system continually resets before you run the FPGA VI. If the system is in a state in which it continually resets, [contact National Instruments](#).

Programming the CompactRIO Chassis FPGA LED (FPGA Interface)

The cRIO-910x Reconfigurable Embedded chassis and the cRIO-9072/9074 Integrated controller and chassis have an FPGA LED. You can use the FPGA LED to help debug your application or easily retrieve application status. Complete the following steps to configure the FPGA LED.



Note The cRIO-9002/9004 embedded controller does not have an FPGA LED.

1. [Configure](#) the CompactRIO system.
2. Make sure the **FPGA LED** I/O item is [added](#) to the CompactRIO system.
3. Right-click the FPGA target in the **Project Explorer** window and select **New»VI** from the shortcut menu to add a new VI to the CompactRIO system.
4. Place an [FPGA I/O Node](#) on the block diagram of the VI.
5. Click the element section of the FPGA I/O Node and select **Chassis I/O»FPGA LED** from the shortcut menu.
6. Right-click the **FPGA LED** input of the FPGA I/O Node and select **Create»Control** from the shortcut menu. When **FPGA LED** is TRUE, the CompactRIO chassis FPGA LED is lit. When **FPGA LED** is FALSE, the CompactRIO chassis FPGA LED is off.

R Series Expansion Chassis Properties Dialog Box (FPGA Interface)

This dialog box appears when you right-click an cRIO-9151 R Series Expansion chassis in the **Project Explorer** window and select **Properties** from the shortcut menu. Use this dialog box to enter a name for the R Series Expansion chassis and select the connector to which it is installed.

- **Name**—You can use this text box to enter a new name for the chassis.
- **Location**—You can use this pull-down menu to select the connector to which the chassis is connected.

Reading from CompactRIO Channels (FPGA Interface)

Complete the following steps to read data from a CompactRIO channel.

1. [Configure](#) the CompactRIO system.
2. Make sure the input channel is [added](#) to the CompactRIO system.
3. Right-click the FPGA target in the **Project Explorer** window and select **New»VI** from the shortcut menu to add a new FPGA VI to the CompactRIO system.
4. Place an [FPGA I/O Node](#) on the block diagram of the FPGA VI.
5. Click the element section of the FPGA I/O Node and select **Module»Module/Channel** from the shortcut menu, where *Module* represents a C Series module and *Module/Channel* is the channel from which you want to read. If you configure an FPGA I/O Node for multiple channels, the I/O Node [accesses the channels simultaneously or sequentially](#) depending on the type of module and other factors.



Note LabVIEW can read CompactRIO digital input lines as ports for some I/O modules. Refer to the reference topic for the module you are using for more information. You can find the reference topic for the module by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Digital Input Modules»NI 9xxx**.

Reading the Temperature of a CompactRIO Chassis or a Single-Board RIO Device (FPGA Interface)

The cRIO-910x Reconfigurable Embedded chassis, the cRIO-9072/9074 Integrated controller and chassis, and the [Single-Board RIO](#) (sbRIO) devices have an onboard temperature sensor. Complete the following steps to read the temperature of the chassis or device.

1. [Configure](#) the CompactRIO or sbRIO system.
2. Make sure the **Chassis Temperature I/O** item is [added](#) to the system.
3. Right-click the FPGA target in the **Project Explorer** window and select **New»VI** from the shortcut menu to add a new VI to the system.
4. Place an [FPGA I/O Node](#) on the block diagram of the VI.
5. Click the element section of the FPGA I/O Node and select **Chassis I/O (or Onboard I/O)»Chassis Temperature** from the shortcut menu.
6. Right-click the **Chassis Temperature** output of the FPGA I/O Node and select **Create»Indicator** from the shortcut menu.

The **Chassis Temperature** output returns 16-bit values for the temperature. You can [convert this value](#) into meaningful engineering units.

Simultaneously Reading From or Writing to Multiple CompactRIO Channels (FPGA Interface)

You can simultaneously read from or write to multiple channels on a single simultaneous module such as the [NI 9215](#), [NI 9225](#), [NI 9229](#), [NI 9239](#), [NI 9263](#), [NI 9264](#), and [NI 9265](#). Configure a single [FPGA I/O Node](#) with the channels you want to simultaneously sample or update. If you configure a single FPGA I/O Node with channels from different modules, FPGA I/O execution begins at the same time for all the modules, but the actual sampling or update times may differ between modules depending on the types of modules, the number of channels used on each module, the arbitration settings, and whether or not the modules are idle when the I/O Node begins execution. If you configure an FPGA I/O Node with multiple channels on a non-simultaneous module, the function accesses the channels sequentially.

Understanding Power-On and Startup Output States for CompactRIO Output Modules (FPGA Interface)

The power-on output state is the state that a C Series output module is in when power is applied to the module. Refer to the [NI 9xxx Operating Instructions and Specifications](#) for the module for the power-on output state. You cannot configure the power-on output state.

The startup output state is the state that a CompactRIO channel is in after you load and run an FPGA VI that communicates with the output module. You can configure startup output states.

Configuring Startup Output States

Use the [FPGA I/O Node](#) analog or digital output options to change startup output states. You can develop the FPGA VI so that the output function executes when the FPGA VI starts running.



Note If you insert an NI 9263 or NI 9265 module while the CompactRIO chassis is powered on and an FPGA VI is running, the analog output channels reset to the power-on output state, and the FPGA VI continues running and communicating with the analog output channels.

Understanding Output States

Action	State of the Output Channels
The chassis with the module powers on.	Power-on output state
An FPGA VI loads.	Startup output state
An Invoke Method function configured with the Abort method executes on the host VI.	Power-on output state
A Close FPGA VI Reference function that is configured to Close and Reset executes. Close and Reset is the default action for the Close FPGA VI Reference function.	Power-on output state
A Close FPGA VI Reference function that is configured to Close executes.	Last state that was written
You click the Abort button on the host VI or select Operate»Stop while LabVIEW is targeted to an FPGA device and the FPGA VI is running with Interactive Front Panel Communication .	Last state that was written
You remove and reinsert one of the following modules: NI 9263 , NI 9265 , NI 9476 , NI 9477 , or NI 9478 .	Power-on output state
You remove and reinsert an NI 9264 module and you have not loaded the FPGA VI.	Power-on output state
You remove and reinsert an NI 9264 module after you loaded the FPGA VI.	Last state that was written
You remove and reinsert one of the following modules: NI 9401 , NI 9402 , NI 9403 , NI 9472 , NI 9474 , NI 9475 , NI 9481	Last state that was

or [NI 9485](#). National Instruments recommends that you do not replace a module that is a different type with one of these modules after the FPGA VI is loaded. Replacing a module that is a different type than the module with which you are replacing it after the FPGA VI is loaded can place the output channels in an undefined or unexpected state.

written

Using the RIO Device I/O Control (FPGA Interface)

Place the RIO Device I/O control on the front panel of a VI. Use the I/O control pull-down menu to select a RIO device.

Selecting a RIO Device in a Project VI Targeted to My Computer

If the VI is part of a project and is targeted to My Computer, the I/O control pull-down menu shows the following sections from top to bottom.

- RIO devices in the project
- Local RIO devices and local RIO aliases
- Browse, which opens the [Browse RIO Devices](#) dialog box
- Recently used RIO devices

Selecting a RIO Device in a Project VI Targeted to an RT Controller

If the VI is part of a project and is targeted to an RT controller, the I/O control pull-down menu shows the following section.

- RIO devices local to the controller

Selecting a RIO Device in a Non-Project VI

If the VI is not part of a project, the I/O control pull-down menu shows the following sections from top to bottom.

- Local RIO devices and local RIO aliases
- Browse, which opens the [Browse RIO Devices](#) dialog box
- Recently used RIO devices

Writing to CompactRIO Channels (FPGA Interface)

Complete the following steps to write to CompactRIO output channels.

1. [Configure](#) the CompactRIO system.
2. Make sure the output channel is [added](#) to the CompactRIO system.
3. Right-click the FPGA target in the **Project Explorer** window and select **New»VI** from the shortcut menu to add a new FPGA VI to the CompactRIO system.
4. Place an [FPGA I/O Node](#) on the block diagram of the FPGA VI.
5. Click the element section of the FPGA I/O Node and select **Module»Module/Channel** from the shortcut menu, where *Module* represents a C Series module and *Module/Channel* is the channel to which you want to write. If you configure an FPGA I/O Node for multiple channels, the I/O Node [accesses the channels simultaneously or sequentially](#) depending on the type of module and other factors.



Note LabVIEW can write to CompactRIO digital output lines as ports for some I/O modules. Refer to the reference topic for the module you are using for more information. You can find the reference topic for the module by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Digital Output Modules»NI 9xxx**.

Confirming the State of Digital Output Channels

For digital channels, you also can read the digital output channel. Complete the following steps to read from the digital output channels.

1. Place another FPGA I/O Node function on the block diagram. This FPGA I/O Node must execute after the FPGA I/O Node you added above.
2. Click the element section of the FPGA I/O Node and select ***Module»Module/Channel*** from the shortcut menu, where *Module* represents a C Series module and *Module/Channel* is the channel you want to read.
3. Right-click the new FPGA I/O Node and select **Change to Read** from the shortcut menu.

Converting and Calibrating CompactRIO Analog Input Values (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the **C Series Module Properties** dialog box for an analog input module if you want the [FPGA I/O Node](#) to return calibrated, **fixed-point** data for the module. If you set the **Calibration Mode** to **Raw**, the FPGA I/O Node returns uncalibrated, binary values for the following analog input modules:

- [NI 9201](#)
- [NI 9203](#)
- [NI 9205](#)
- [NI 9206](#)
- [NI 9215](#)
- [NI 9217](#)
- [NI 9221](#)
- [NI 9225](#)
- [NI 9229](#)
- [NI 9233](#)
- [NI 9234](#)
- [NI 9235](#)
- [NI 9236](#)
- [NI 9237](#)
- [NI 9239](#)

If you set the **Calibration Mode** to **Raw**, you must convert the binary values into meaningful engineering units and apply calibration constants to achieve more accurate results. You must convert and calibrate these values in the host VI.



Note After you convert the binary values to engineering units for the NI 9217, you can also [convert the nominal resistance values into temperature values](#).



Note The binary data returned by the [NI 9211](#), [NI 9213](#), or [NI 9219](#) is already calibrated. You must use different equations to [convert values for the NI 9211](#), [convert values for the NI 9213](#), or [convert values for the NI 9219](#) into engineering units.

Using a VI to Convert and Calibrate Values

Refer to the Binary to Nominal VI in the labview\examples\CompactRIO\Basic IO\Analog Raw Host Calibration\AI Raw Host Calibration\AI Raw Host Calibration - cRIO.lvproj for an example of converting binary analog input values to calibrated engineering units. You can use the Binary to Nominal VI as a subVI in the host VI.

■ Open example

For the NI 9205, you can also refer to the Binary to Nominal Polynomial (Host) VI in the labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Linearization Coefficients\NI 9205 Linearization Coefficients.lvproj for an example of using the straight-line conversion algorithm along with the NI-MCal linearization correction to convert binary analog input values to calibrated engineering units. You can use the Binary to Nominal Polynomial (Host) VI as a subVI in the host VI.

■ Open NI 9205 example

For the NI 9206, you can also refer to the Binary to Nominal Polynomial (Host) VI in the labview\examples\CompactRIO\Module Specific\NI 9206\NI 9206 Linearization Coefficients\NI 9206 Linearization Coefficients.lvproj for an example of using the straight-line conversion algorithm along with the NI-MCal linearization correction to convert binary analog input values to calibrated engineering units. You can use the Binary to Nominal Polynomial (Host) VI as a subVI in the host VI.

■ Open NI 9206 example

Using an Equation to Convert and Calibrate Values

You can use the following equation in the host VI to convert binary analog input values to calibrated engineering units:

$$\text{Input Engineering Units} = (\text{Binary Value} \times \text{LSB Weight} - \text{Offset})^*$$

where *Binary Value* is the signed or unsigned value returned by the FPGA I/O Node

LSB Weight is the value returned by the LSB Weight property

Offset is the value returned by the Offset property.

The units of *LSB Weight* and *Offset* differ per module. Refer to the reference topic for the module you are using for the *LSB Weight* and *Offset* units. You can find the reference topic for the module by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Analog Input Modules»NI 9xxx**.

To convert to calibrated engineering units, use the [FPGA I/O Property Node](#) to read the LSB Weight and Offset properties. If you do not want to read the LSB Weight and Offset values from the module, you can convert to uncalibrated engineering units by using the following values for *Offset* and *LSB Weight*:

$$\text{Offset} = 0^\dagger$$

$$\text{LSB Weight} = \text{Typical Input Span} \div 2^{\text{ADC Resolution}}$$

where *Typical Input Span* is the value for the module in the table below

ADC Resolution is the ADC resolution value in the [NI 9xxx Operating Instructions and Specifications](#) for the module.

Module	Typical Input Span
NI 9201	21.06 V
NI 9203	21.56 mA (unipolar), 43.12 mA (bipolar)
NI 9205	20.8 V
NI 9206	21.5 V
NI 9215	20.8 V
NI 9217	1000 Ω
NI 9221	125 V

NI 9225	850 V
NI 9229	125.28 V
NI 9233	10.8 V
NI 9234	10.2 V
NI 9235	52.6 mV/V
NI 9236	52.6 mV/V
NI 9237	50 mV/V
NI 9239	21.04 V

*When converting and calibrating data acquired from the NI 9203 in ± 20 mA range, the equation is:

$$\text{Input Engineering Units} = ((\text{Binary Value} - 32768) \times \text{LSB Weight} - \text{Offset})$$

†When calculating engineering units for the NI 9203 in ± 20 mA range, *Offset* = 20 mA.

Detecting Out-of-Range Channels for CompactRIO Analog Input Channels (FPGA Interface)

You can detect an out-of-range CompactRIO analog input channel when you [read](#) the channel. If you set the **Calibration Mode** to **Calibrated** in the **C Series Module Properties** dialog box and the [FPGA I/O Node](#) returns a value greater than the minimum operating range value, the channel might be out of range. Refer to the [NI 9xxx Operating Instructions and Specifications](#) for the module for more information about module specifications.

If you set the **Calibration Mode** to **Raw** in the **C Series Module Properties** dialog box and a channel is out of range, the FPGA I/O Node returns the full-scale binary value, as shown in the table below.

Module	Full-Scale Binary Value
NI 9201	-2,048 (0xF800) or 2,047 (0x7FF)
NI 9203	0 (0x0000) or 65,535 (0xFFFF)
NI 9205 *	-32,768 (0x8000) or 32,767 (0x7FFF)
NI 9206 *	-32,768 (0x8000) or 32,767 (0x7FFF)
NI 9211	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9213	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9215	-32,768 (0x8000) or 32,767 (0x7FFF)
NI 9217	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9219	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9221	-2,048 (0xF800) or 2,047 (0x7FF)
NI 9225	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9229	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9233	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9234	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9235	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9236	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
NI 9237	-8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)

NI 9239	–8,388,608 (0xFF800000) or 8,388,607 (0x7FFFFFFF)
----------------	---

*For the NI 9205 and NI 9206, a channel might be out of range before the FPGA I/O Node reaches the minimum or maximum full-scale binary value.

NI 9201 (FPGA Interface)

CompactRIO 8-Channel, ± 10 V, 12-Bit Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9201 has AI channels 0 to 7.

You can [configure the minimum time between conversions](#) and [understand scanning](#) for these channels.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in nV/LSB for the channel. Use this value to convert and calibrate NI 9201 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9201 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9201/9221 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9201 and NI 9221 (FPGA Interface)

Right-click an [NI 9201](#) or [NI 9221](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 16 bits and an integer word length of 5 bits for the NI 9201 and 7 bits for the NI 9221. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Minimum Time Between Conversions**—Specifies the [minimum time between conversions](#) in μs .

Configuring the Minimum Time Between Conversions for the NI 9201/9221 (FPGA Interface)

You can configure the minimum time between conversions for the [NI 9201/9221](#) in the [C Series Module Properties](#) dialog box.

Complete the following steps to configure the minimum time between conversions for the NI 9201/9221.

1. **Configure** the CompactRIO system, and add an NI 9201/9221.
2. Right-click the NI 9201/9221 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Enter a value between 0 and 10 μs in increments of 25 ns in the **Minimum Time Between Conversions** text box.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.



Note The default minimum time between conversions for the NI 9201 is 2 μs , and the default minimum time between conversions for the NI 9221 is 1.25 μs . The accuracy specifications in the [NI 9201/9221 Operating Instructions and Specifications](#) are based on these default values. Refer to the following tables for examples of how the value you enter in the **Minimum Time Between Conversions** text box affects the actual time between conversions and the accuracy of the module.

Refer to the following table if you are using a chassis other than the cRIO-9151 R Series Expansion chassis.

Module	Sampling Data from a Single or Multiple Channels?	Minimum Time Between Conversions*	Able to Achieve Accuracy Specified Between Conversions
NI 9201	Single	$\geq 1.25 \mu\text{s}$	Yes
	Single	$< 1.25 \mu\text{s}$	No
	Multiple	$\geq 2 \mu\text{s}$	Yes

		< 2 μs	No
NI 9221	Single	$\geq 1.25 \mu\text{s}$	Yes
		< 1.25 μs	No
	Multiple	$\geq 1.25 \mu\text{s}$	Yes
		< 1.25 μs	No

Refer to the following table if you are using the cRIO-9151 R Series Expansion chassis.

Module	Sampling Data from a Single or Multiple Channels?	Minimum Time Between Conversions*	Able to A Specified Between Con
NI 9201	Single	$\geq 2.1 \mu\text{s}$	Yes
		< 2.1 μs	No
	Multiple	$\geq 2.1 \mu\text{s}$	Yes
		< 2.1 μs and $\geq 2 \mu\text{s}$	No
		< 2 μs	No
NI 9221	Single	$\geq 2.1 \mu\text{s}$	Yes
		< 2.1 μs	No
	Multiple	$\geq 2.1 \mu\text{s}$	Yes
		< 2.1 μs and $\geq 1.25 \mu\text{s}$	No

	< 1.25 μ s	No
--	----------------	----

* The value you set in the **C Series Module Properties** dialog box.

** The actual time between conversions depends on how you develop the FPGA VI. Yes indicates that it is possible to write an FPGA VI that can sustain the minimum time between conversions specified in the **C Series Module Properties** dialog box. No indicates that the minimum time between conversions specified in the **C Series Module Properties** dialog box is too low and you may not be able to write an FPGA VI that can sustain the specified time.

Understanding NI 9201/9221 Scanning (FPGA Interface)

To scan the channels of the [NI 9201/9221](#), configure an [FPGA I/O Node](#) with the channels you want to acquire from the module. The channels are scanned in numerical order. They are not scanned in the order of appearance in an FPGA I/O Node.

The first time an FPGA I/O Node configured with channels on an NI 9201/9221 module executes, the NI 9201/9221 module performs two setup conversions before converting the first channel. The module stores the channel list from the most recently executed FPGA I/O Node. The module does not repeat the setup conversions for subsequent FPGA I/O Node reads that are configured with the same channel list. The module performs the setup conversions only if the channel list changes.

Refer to the Analog Input VI in the `labview\examples\CompactRIO\Basic IO\Analog Input` directory for an example of handling the setup conversions in the FPGA VI block diagram.

■ [Open example](#)

NI 9203 (FPGA Interface)

CompactRIO 8-Channel, ± 20 mA, 16-Bit Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x , where x is the number of the channel. The NI 9203 has AI channels 0 to 7.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Set Input Range	Sets the input range for a channel as either 0–20 mA or ± 20 mA. This method overwrites the value you configure in the C Series Module Properties dialog box.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight (± 20 mA range)	Returns the LSB weight in pA/LSB for the channel. Use this value to convert and calibrate NI 9203 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box, and you set the input range to ± 20 mA.
LSB Weight (4–20 mA range)	Returns the LSB weight in pA/LSB for the channel. Use this value to convert and calibrate NI 9203 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box, and you set the input range to 0–20 mA.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Offset (± 20 mA range)	Returns the calibration offset in pA for the channel. Use this value to convert and calibrate NI 9203 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box, and you set the input range to ± 20 mA.
Offset (4–20 mA range)	Returns the calibration offset in pA for the channel. Use this value to convert and calibrate NI 9203 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box, and you set the input range to 0–20 mA.
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9203 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9203 (FPGA Interface)

Right-click an [NI 9203](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of amps. The fixed-point data is signed, with a word length of 21 bits and an integer word length of –4 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Minimum Time Between Conversions**—Specifies the minimum time between conversions in μ s.
- **Channel Configuration**—Specifies the input range for each channel.
 - **Channels**—Specifies the channel(s) for which you want to select the input range.
 - **Input Range**—Specifies the input range for the selected channel(s) as either 0–20 mA or \pm 20 mA.

NI 9205 (FPGA Interface)

CompactRIO 32-Channel Single-Ended/16-Channel Differential, ± 200 mV to ± 10 V, 16-Bit Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9205 has AI channels 0 to 31. The channel is sampled using the currently configured terminal mode and voltage range. You can configure these settings at edit time using the C Series Module Properties dialog box or at run time using the Set Terminal Mode or Set Voltage Range methods.
DIO	Digital input channel used to access the module digital line, which is labeled PFIO. This terminal is not available for an onboard NI 9205 on a Single-Board RIO (sbRIO) device.
Trig	Returns the output of the module trigger circuitry. Use the Set Triggers method to configure the trigger circuit.
DO0	Digital output channel 0. This terminal is not available for an onboard NI 9205 on an sbRIO device.

Refer to the [Conversion Timing](#) topic for information about the order and timing of conversions for these I/O channels.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any I/O methods.

Method	Description
IO Sample	<p>Acquires a single sample from the module. The channel number, terminal mode, and voltage range are all configurable at run time.</p> <p> Note National Instruments suggests that you do not configure the VI to run an IO Sample method and another FPGA I/O Method Node in parallel when both are targeted to the same C Series module. Similarly, National Instruments suggests that you do not configure the VI to run multiple IO Sample methods in parallel when the IO Sample methods are targeted to the same C Series module.</p> <p>Refer to the 9205 Basic IO VI and the NI 9205 Advanced IO VI in the labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Basic IO and labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Advanced IO directories for examples related to this method.</p> <ul style="list-style-type: none"> • Basic IO <ul style="list-style-type: none"> ■ Open example • Advanced IO <ul style="list-style-type: none"> ■ Open example
Set Terminal Mode	<p>Sets the terminal mode for a channel as RSE (referenced single-ended), NRSE (non-referenced single-ended), or DIFF (differential). This method overwrites the value you configure in the C Series Module Properties dialog box.</p>
Set Triggers	<p>Sets the trigger output to one of the following values:</p> <ul style="list-style-type: none"> • None

- AI Below Low
- AI Above High
- AI Inside Region
- AI Below Low with Hysteresis
- AI Above High with Hysteresis
- DI Trigger



Note The onboard NI 9205 on sbRIO devices does not support DI triggering.

Refer to the 9205 Basic Triggering VI and the 9205 Advanced Triggering VI in the labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Basic Triggering and labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Advanced Triggering directories for examples related to this method.

- Basic Triggering
 - Open example
- Advanced Triggering
 - Open example

Set Voltage Range

Sets the input range for a channel as ± 10 V, ± 5 V, ± 1 V, or ± 200 mV. This method overwrites the value you configure in the **C Series Module Properties** dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Linearization Coefficient x	<p>(Optional) NI-MCal linearization coefficient x, where x is the number of the linearization coefficient. The NI 9205 has linearization coefficients 3 to 0. You can use these coefficients in calibration to compensate for some types of measurement nonlinearity inherent in this device.</p> <p>Refer to the NI 9205 Advanced IO VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Advanced IO</code> directory for an example of this concept.</p> <p>■ Open example</p>
LSB Weight (± 10 V range)	Returns the LSB weight in pV/LSB for the ± 10 V range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
LSB Weight (± 1 V range)	Returns the LSB weight in pV/LSB for the ± 1 V range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
LSB Weight (± 200 mV range)	Returns the LSB weight in pV/LSB for the ± 200 mV range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
LSB Weight (± 5 V range)	Returns the LSB weight in pV/LSB for the ± 5 V range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Module ID	Returns the module ID .
Offset (± 10 V range)	Returns the calibration offset in nV for the ± 10 V range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module

	Properties dialog box.
Offset (± 1 V range)	Returns the calibration offset in nV for the ± 1 V range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset (± 200 mV range)	Returns the calibration offset in nV for the ± 200 mV range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset (± 5 V range)	Returns the calibration offset in nV for the ± 5 V range. Use this value to convert and calibrate NI 9205 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9205 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9205 and NI 9206 (FPGA Interface)

Right-click an [NI 9205](#) or [NI 9206](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 26 bits and an integer word length of 5 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Minimum Time Between Conversions**—Specifies the [minimum time between conversions](#) in μs .
- **Channel Configuration**—Specifies the input range for each channel.
 - **Channels**—Specifies the channel(s) for which you want to select the input range.
 - **Input Range**—Specifies the input range for the selected channel(s) as $\pm 10\text{ V}$, $\pm 5\text{ V}$, $\pm 1\text{ V}$, or $\pm 200\text{ mV}$.
 - **Terminal Mode**—Specifies the terminal mode for the selected channel(s) as RSE (referenced single-ended), NRSE (non-referenced single-ended), or DIFF (differential).

IO Sample Method (FPGA Interface)

This module method acquires a single sample from the module. The channel number, terminal mode, and voltage range are all configurable at run time. You use this module method by selecting it in an [FPGA I/O Method Node](#) that is configured for the appropriate device and/or channel. [Details](#) [Examples](#)



Note National Instruments suggests that you do not configure the VI to run an IO Sample method and another FPGA I/O Method Node in parallel when both are targeted to the same C Series module. Similarly, National Instruments suggests that you do not configure the VI to run multiple IO Sample methods in parallel when the IO Sample methods are targeted to the same C Series module.



Configuration [i+2] contains encoded configuration information that gets loaded into the module conversion pipeline. This configuration information controls the data to be sampled two iterations into the future. Refer to the [Conversion Timing](#) topic for more information. Refer to the [Configuration Encoding](#) table below for an example of how the configuration information is encoded.



Data [i] Returns the data from the current sample.



error in describes error conditions that occur before this VI or function runs. The default is `no error`. If an error occurred before this VI or function runs, the VI or function passes the **error in** value to **error out**. This VI or function runs normally only if no error occurred before this VI or function runs. If an error occurs while this VI or function runs, it runs normally and sets its own error status in **error out**. Use **error in** and **error out** to check errors and to specify execution order by wiring **error out** from one node to **error in** of the next node.



status is TRUE (X) if an error occurred before this VI or function ran or FALSE (checkmark) to indicate a warning or that no error occurred before this VI or function ran. The default is FALSE.



code is the error code number identifying an error. The default is 0. If status is TRUE, code is a nonzero [error code](#).

If status is FALSE, code is 0 or a warning code.

 **source** always contains an empty string because strings are not supported in LabVIEW FPGA.

 **error out** contains error information. If **error in** indicates that an error occurred before this VI or function ran, **error out** contains the same error information. Otherwise, it describes the error status that this VI or function produces. Right-click the **error out** indicator on the front panel and select **Explain Error** from the shortcut menu for more information about the error.

 **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

 **code** is the error code number identifying an error. The default is 0. If status is TRUE, code is a nonzero [error code](#). If status is FALSE, code is 0 or a warning code.

 **source** always contains an empty string because strings are not supported in LabVIEW FPGA.

Configuration Encoding Table

The following table describes how to construct the Configuration [i+2] value to write to the IO Sample method to perform a particular operation. The **Operation** column lists the types of operations that can be performed using the IO Sample method. The **Configuration Bit Fields** column lists the bit positions within the **Configuration [i+2]** number that need to be set. The **Values** column specifies the binary value to apply to the specified **Configuration Bit Fields**.

Operation	Configuration Bit Fields	Values
AI Read	15:13	001b
	12:11	01b for channels 0-15 10b for channels 16-31
	10:8	000b for channels 0, 8, 16, or 24 001b for channels 1, 9, 17, or 25 ... 110b for channels 6, 14, 22, or 30 111b for channels 7, 15, 23, or 31
	7:6	00b
	5:4	00b for NRSE mode 11b for RSE or DIFF modes
	3:2	11b for channels 8-15 or 24-31 01b for DIFF mode on channels 0-7 or 16-23 10b for RSE or NRSE mode on channels 0-7 or 16-23
	DI0 Read	15:0
DO0 Write	15:1	0100000000000000b
	0	Binary value to write to DO0

Using This Method

The IO Sample method provides an efficient and flexible interface to the module. You can use this method to acquire a single sample from any of the channels on the module, at any range, and with any available input mode.

When this method is executed, the module performs a single conversion on the next channel present in the conversion pipeline on the module. The data from this conversion is returned via the **Data [i]** method output. At the same time that the conversion data is read from the module, the new configuration information specified by the **Configuration [i+2]** input is loaded into the configuration pipeline on the module. The pipeline is two samples deep. So, the configuration information specified on one execution of the IO Sample method determines which channel will be sampled by the IO Sample method two iterations into the future. Refer to the [Conversion Timing](#) topic for more details.

Examples

Refer to the 9205 Basic IO VI and the NI 9205 Advanced IO VI in the labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Basic IO and labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Advanced IO directories for examples of the IO concepts discussed above.

- Basic IO
 - Open example
- Advanced IO
 - Open example

Set Terminal Mode Method (FPGA Interface)

Overrides the default setting provided in the [C Series Module Properties](#) dialog box. You use this module method by selecting it in an [FPGA I/O Method Node](#) that is configured for the appropriate device and/or channel.



AI Channel is the channel you want to set the terminal mode for.



Terminal Mode is an enumeration of the following trigger modes:

- **RSE** is a single-ended measurement between any of the 32 individual channels and the module ground.
- **NRSE** is a single-ended measurement between any of the 32 individual channels and the AISENSE line into the module.
- **DIFF** is a differential measurement between two of the 32 individual channels, with the name of the differential channel matching the channel tied to the positive side of the differential measurement. Channels AI0 to AI7, and AI16 to AI23 may be set to DIFF configuration.



error in describes error conditions that occur before this VI or function runs. The default is no error. If an error occurred before this VI or function runs, the VI or function passes the **error in** value to **error out**. This VI or function runs normally only if no error occurred before this VI or function runs. If an error occurs while this VI or function runs, it runs normally and sets its own error status in **error out**. Use **error in** and **error out** to check errors and to specify execution order by wiring **error out** from one node to **error in** of the next node.



status is TRUE (X) if an error occurred before this VI or function ran or FALSE (checkmark) to indicate a warning or that no error occurred before this VI or function ran. The default is FALSE.



code is the error or warning code. The default is 0. If **status** is TRUE, **code** is a nonzero [error code](#). If **status** is FALSE, **code** is 0 or a warning code.



source always contains an empty string because strings are not supported in LabVIEW FPGA.



error out contains error information. If **error in** indicates that an error occurred before this VI or function ran, **error out** contains the same error information. Otherwise, it describes the error status that this VI or function produces. Right-click the **error out** indicator on the front panel and select **Explain Error** from the shortcut menu for more information about the error.



status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.



code is the error or warning code. If **status** is TRUE, **code** is a nonzero **error code**. If **status** is FALSE, **code** is 0 or a warning code.



source always contains an empty string because strings are not supported in LabVIEW FPGA.

Set Triggers Method (FPGA Interface)

This module method controls the data returned by the trigger digital input line. You use this module method by selecting it in an [FPGA I/O Method Node](#) that is configured for the appropriate device and/or channel. [Details](#)
[Examples](#)



Trigger Mode is an enumeration of the following trigger mode options:

- **None** disables the trigger function.
- **AI Below Low** detects when the analog signal is below the low threshold you specify.
- **AI Above High** detects when the analog signal is above the high threshold you specify.
- **AI Inside Region** detects when the analog signal is between the high and low thresholds you specify.
- **AI Below Low with Hysteresis** asserts the trigger when the signal starts above the high threshold you specify and then crosses below the low threshold you specify. The trigger deasserts when the signal crosses above the high threshold you specify.
- **AI Above High with Hysteresis** asserts the trigger when the signal starts below the low threshold you specify and then crosses above the high threshold you specify. The trigger deasserts when the signal crosses below the low threshold you specify.
- **DI Trigger** detects when the digital signal on PFI0 is logic high.



Note The onboard NI 9205 on [Single-Board RIO](#) (sbRIO) devices does not support DI triggering.



AI High Threshold defines the upper threshold for the analog trigger circuit.



AI Low Threshold defines the lower threshold for the analog trigger circuit.



error in describes error conditions that occur before this VI or function runs. The default is no error. If an error occurred before

this VI or function runs, the VI or function passes the **error in** value to **error out**. This VI or function runs normally only if no error occurred before this VI or function runs. If an error occurs while this VI or function runs, it runs normally and sets its own error status in **error out**. Use **error in** and **error out** to check errors and to specify execution order by wiring **error out** from one node to **error in** of the next node.

 **status** is TRUE if an error occurred. If **status** is TRUE, the VI does not perform any operations.

 **code** is the error code number identifying an error. The default is 0. If status is TRUE, code is a nonzero [error code](#). If status is FALSE, code is 0 or a warning code.

 **source** always contains an empty string because strings are not supported in LabVIEW FPGA.

 **error out** contains error information. If **error in** indicates that an error occurred before this VI or function ran, **error out** contains the same error information. Otherwise, it describes the error status that this VI or function produces. Right-click the **error out** indicator on the front panel and select **Explain Error** from the shortcut menu for more information about the error.

 **status** is TRUE if an error occurred. If **status** is TRUE, the VI does not perform any operations.

 **code** is the error code number identifying an error. The default is 0. If status is TRUE, code is a nonzero [error code](#). If status is FALSE, code is 0 or a warning code.

 **source** always contains an empty string because strings are not supported in LabVIEW FPGA.

Using This Method

Use the Set Triggers Method to configure the trigger circuit on the module. Once the trigger circuit is configured, use an FPGA I/O Node configured to access the Trig channel to monitor the output of the trigger circuit. The trigger circuit does not automatically initiate any conversions on the module. It is up to the FPGA VI to monitor the Trig line, and then initiate a conversion using an FPGA I/O Node or the IO Sample method when the appropriate trigger event occurs.

Perform the following steps to trigger an acquisition using the Set Triggers Method:

1. Ensure that the module pipeline is primed with the next two channels that you want to sample after the trigger event. This step will minimize the delay between the trigger event and the AI channels getting sampled. For analog triggering modes, this step also configures the analog trigger circuitry to monitor the desired channel. The AI trigger circuit monitors the channel waiting in the module pipeline for the next conversion.

If you are using FPGA I/O Nodes to perform the acquisition, this step can be accomplished by executing an FPGA I/O Node configured to match the FPGA I/O Node that you will be executing after the trigger event. If you are using one of the analog triggering modes along with the FPGA I/O Nodes, the AI trigger circuit will operate on the lowest numbered channel present in the FPGA I/O Node.

Refer to the [IO Sample](#) method and [Conversion Timing](#) topics for more information on managing the module pipeline using the IO Sample method. If you are using one of the analog trigger modes along with the IO Sample method, the AI trigger circuit will operate on the channel waiting in the pipeline to be converted by the next IO Sample method.

2. Execute the [Set Triggers](#) method with the appropriate trigger mode selected. The trigger mode circuitry is implemented in the hardware as you specify in this method. Refer to the [NI 9205 Operating Instructions and Specifications](#) or [NI 9206 Operating Instructions and Specifications](#) for more information about the

trigger modes.

3. Monitor the Trig channel in a loop until you see the appropriate trigger condition occur.
4. Execute an FPGA I/O Node or IO Sample method function to acquire data in response to the trigger event.

Examples

Refer to the 9205 Basic Triggering VI and the 9205 Advanced Triggering VI in the labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Basic Triggering and labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Advanced Triggering directories for examples of the triggering concepts discussed above.

- Basic Triggering
 - Open example
- Advanced Triggering
 - Open example

Set Voltage Range Method (FPGA Interface)

This module method overrides the default setting provided in the [C Series Module Properties](#) dialog box. You use this module method by selecting it in an [FPGA I/O Method Node](#) that is configured for the appropriate device and/or channel.

 **AI Channel** is the channel you want to set the voltage range for.

 **Voltage Range** is an enumeration of the following voltage ranges:

- ± 10 V
- ± 5 V
- ± 1 V
- ± 200 mV

 **error in** describes error conditions that occur before this VI or function runs. The default is no error. If an error occurred before this VI or function runs, the VI or function passes the **error in** value to **error out**. This VI or function runs normally only if no error occurred before this VI or function runs. If an error occurs while this VI or function runs, it runs normally and sets its own error status in **error out**. Use **error in** and **error out** to check errors and to specify execution order by wiring **error out** from one node to **error in** of the next node.

 **status** is TRUE if an error occurred. If **status** is TRUE, the VI does not perform any operations.

 **code** is the error code number identifying an error. The default is 0. If status is TRUE, code is a nonzero [error code](#). If status is FALSE, code is 0 or a warning code.

 **source** always contains an empty string because strings are not supported in LabVIEW FPGA.

 **error out** contains error information. If **error in** indicates that an error occurred before this VI or function ran, **error out** contains the same error information. Otherwise, it describes the error status that this VI or function produces. Right-click the **error out** indicator on the front panel and select **Explain Error** from the shortcut menu for more information about the error.

 **status** is TRUE if an error occurred. If **status** is TRUE, the

VI does not perform any operations.

 **code** is the error code number identifying an error. The default is 0. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

 **source** always contains an empty string because strings are not supported in LabVIEW FPGA.

Conversion Timing for the NI 9205/9206 (FPGA Interface)

Example

The [NI 9205](#) and [NI 9206](#) modules implement a two-element deep pipeline for access to the AI, DIO, and DOO channels. This pipeline results in maximum sample rate and maximum sample quality. The Trig channel has a parallel data path that bypasses the pipeline.

When using the [FPGA I/O Node](#) to sample channels, the pipeline is automatically managed by the FPGA I/O Node, and the channels within the FPGA I/O Node are sampled in numerical order regardless of the order they appear in the node.

If the first two channel requests in the FPGA I/O Node do not match the two channel requests stored in the module pipeline, there will be a delay before the first channel sample occurs. This delay is caused by the FPGA I/O Node automatically updating the module channel sample pipeline, which takes two channel sample cycles.

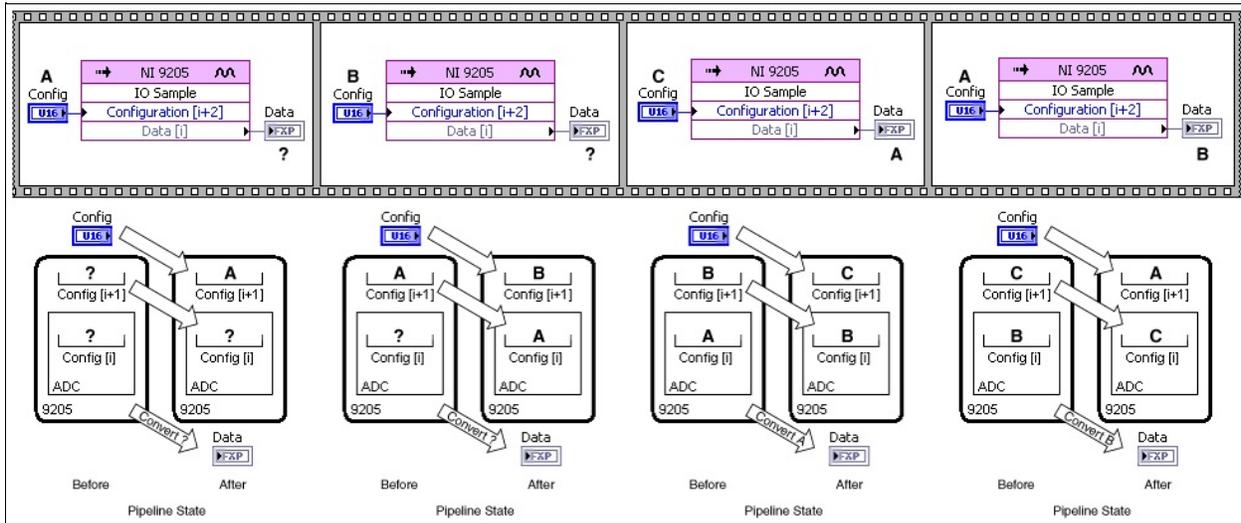
To minimize the need for the pipeline updates, each FPGA I/O Node leaves the module pipeline primed to repeat itself. If you use only one FPGA I/O Node in a looping structure, only the first iteration of the loop will incur the two-cycle time delay. All subsequent iterations operate with no delay.

When using the [IO Sample](#) method, you must take steps to manage this pipeline in the VIs.

To read one channel, that channel configuration must be requested two cycles before the time it is to be sampled. The U16 configuration value for an AI channel sample contains the channel number, channel input range, and terminal mode. The U16 configuration value also can represent an access to the DIO or DOO channels, which is treated the same as an AI operation.

The following diagram illustrates how the pipeline works within a sequence structure. A typical application would use a loop structure to iterate through a scan list, which is a predefined list of configurations, continuously. The sequence structure and art elements show how data moves into the NI 9205/9206 pipeline and then is converted. Note that the IO Sample method first converts the configuration that is in the ADC,

and then shifts the configuration values through the pipeline to prepare for the next time LabVIEW calls the IO Sample method.



The simplest form of pipeline management is to force the pipeline on every set of samples. This assumes that the pipeline is always wrong at the first sample, and sends all IO Sample requests in order. The last two data points need some configuration to be requested, but it does not matter, in this use case, what configuration that is.

A better form of pipeline management emulates the behavior of the FPGA I/O Node. Instead of always assuming that the pipeline is wrong, you choose the last two configuration values based on the first two samples you want to include in the next iteration. This pipeline management system can be accomplished in two parts: initialization and sample iteration.



Tip National Instruments recommends you use the IO Sample method for the following reasons:

- Smaller on FPGA—With large channel count, the IO Sample method is more efficient with FPGA resources, and thus requires less space on the FPGA for the same application.
- Sample flexibility—The IO Sample method gives you more flexibility to change channel range, terminal mode, and channel order at run time and at full sample rate.
- Application timing flexibility—The IO Sample method returns all sample data as soon as it is available. The

FPGA I/O Node will gather data for all requested samples and then return all at once.

The **Minimum Time Between Conversions** control you can set in the [C Series Module Properties](#) dialog box determines the shortest possible time between any two conversions. For channels sampled within the same FPGA I/O Node, the time you set determines the exact time between conversions. For channels sampled within separate FPGA I/O Nodes or for conversions caused by looping on an FPGA I/O Node, the time you set may be less than the actual time between conversions. However, the minimum time you set is never greater than the time between conversions. If the application tries to execute an FPGA I/O Node or IO Sample method faster than the specified minimum time between conversions, the conversion is delayed until the minimum time you set is satisfied.

The default minimum time between conversions for the NI 9205/9206 is 8 μ s. The accuracy specifications in the [NI 9205 Operating Instructions and Specifications](#) or [NI 9206 Operating Instructions and Specifications](#) are based on this default value. If you set the minimum time between conversions to at least 8 μ s, the accuracy of the module is not affected. If you set the minimum time between conversions to less than 8 μ s, the accuracy of the module degrades if you sample data from multiple channels.

Example

Refer to the Advanced IO example in the
labview\examples\CompactRIO\Module Specific\NI 9205\NI 9205 Advanced
IO directory for an example of conversion timing on the NI 9205 and NI
9206 C Series Modules.

■ Open example

NI 9206 (FPGA Interface)

CompactRIO 32-Channel Single-Ended/16-Channel Differential, ± 200 mV to ± 10 V, 16-Bit Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x , where x is the number of the channel. The NI 9206 has AI channels 0 to 31. The channel is sampled using the currently configured terminal mode and voltage range. You can configure these settings at edit time using the C Series Module Properties dialog box or at run time using the Set Terminal Mode or Set Voltage Range methods.
DI0	Digital input channel used to access the module digital line, which is labeled PFIO.
Trig	Returns the output of the module trigger circuitry. Use the Set Triggers method to configure the trigger circuit.
DO0	Digital output channel 0.

Refer to the [Conversion Timing](#) topic for information about the order and timing of conversions for these I/O channels.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any I/O methods.

Method	Description
IO Sample	<p>Acquires a single sample from the module. The channel number, terminal mode, and voltage range are all configurable at run time.</p> <p> Note National Instruments suggests that you do not configure the VI to run an IO Sample method and another FPGA I/O Method Node in parallel when both are targeted to the same C Series module. Similarly, National Instruments suggests that you do not configure the VI to run multiple IO Sample methods in parallel when the IO Sample methods are targeted to the same C Series module.</p> <p>Refer to the 9206 Basic IO VI and the NI 9206 Advanced IO VI in the labview\examples\CompactRIO\Module Specific\NI 9206\NI 9206 Basic IO and labview\examples\CompactRIO\Module Specific\NI 9206\NI 9206 Advanced IO directories for examples related to this method.</p> <ul style="list-style-type: none"> • Basic IO <ul style="list-style-type: none"> ■ Open example • Advanced IO <ul style="list-style-type: none"> ■ Open example
Set Terminal Mode	<p>Sets the terminal mode for a channel as RSE (referenced single-ended), NRSE (non-referenced single-ended), or DIFF (differential). This method overwrites the value you configure in the C Series Module Properties dialog box.</p>
Set Triggers	<p>Sets the trigger output to one of the following values:</p>

- None
- AI Below Low
- AI Above High
- AI Inside Region
- AI Below Low with Hysteresis
- AI Above High with Hysteresis
- DI Trigger

Refer to the 9206 Basic Triggering VI and the 9206 Advanced Triggering VI in the labview\examples\CompactRIO\Module Specific\NI 9206\NI 9206 Basic Triggering and labview\examples\CompactRIO\Module Specific\NI 9206\NI 9206 Advanced Triggering directories for examples related to this method.

- Basic Triggering
 - Open example
- Advanced Triggering
 - Open example

Set Voltage Range

Sets the input range for a channel as ± 10 V, ± 5 V, ± 1 V, or ± 200 mV. This method overwrites the value you configure in the **C Series Module Properties** dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Linearization Coefficient x	<p>(Optional) NI-MCal linearization coefficient x, where x is the number of the linearization coefficient. The NI 9206 has linearization coefficients 3 to 0. You can use these coefficients in calibration to compensate for some types of measurement nonlinearity inherent in this device.</p> <p>Refer to the NI 9206 Advanced IO VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9206\NI 9206 Advanced IO</code> directory for an example of this concept.</p> <p>■ Open example</p>
LSB Weight (±10 V range)	Returns the LSB weight in pV/LSB for the ±10 V range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
LSB Weight (±1 V range)	Returns the LSB weight in pV/LSB for the ±1 V range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
LSB Weight (±200 mV range)	Returns the LSB weight in pV/LSB for the ±200 mV range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
LSB Weight (±5 V range)	Returns the LSB weight in pV/LSB for the ±5 V range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Module ID	Returns the module ID .
Offset (±10 V range)	Returns the calibration offset in nV for the ±10 V range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module

	Properties dialog box.
Offset (± 1 V range)	Returns the calibration offset in nV for the ± 1 V range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset (± 200 mV range)	Returns the calibration offset in nV for the ± 200 mV range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset (± 5 V range)	Returns the calibration offset in nV for the ± 5 V range. Use this value to convert and calibrate NI 9206 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9206 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9211 (FPGA Interface)

CompactRIO 4-Channel, ± 80 mV, 24-Bit Thermocouple Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
TCx	Thermocouple input channel x , where x is the number of the channel. The NI 9211 has TC channels 0 to 3.
CJC	Cold-junction compensation channel. For the best accuracy, read the CJC channel in the same FPGA I/O Node as the thermocouple input channels. You must convert the CJC data to temperature.
Autozero	Autozero channel. For the best accuracy, read the Autozero channel in the same FPGA I/O Node as the thermocouple input channels.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9211 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9211 (FPGA Interface)

Right-click an [NI 9211](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 24 bits and an integer word length of –2 bits. If you select **Calibrated**, you must [convert](#) convert the fixed-point CJC data to binary CJC data and then convert the binary CJC data to temperature. Select **Raw** if you want the FPGA I/O Node to return calibrated, binary data from the module. If you select **Raw**, you must [convert](#) the binary thermocouple values to voltage and convert the binary CJC data to temperature. The default is **Calibrated**.

Converting NI 9211 Data (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9211](#) if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. If you set the **Calibration Mode** to **Calibrated**, you must convert the fixed-point CJC data to binary CJC data and then convert the binary CJC data to temperature. Set the **Calibration Mode** to **Raw** if you want the FPGA I/O Node to return calibrated, binary data from the module. If you set the **Calibration Mode** to **Raw**, you must convert the binary thermocouple values to voltage and convert the binary CJC data to temperature. You must convert these values in the host VI.

Using a VI to Convert Data to Temperature

Refer to the Convert to Temperature (NI 9211) polymorphic VI in the labview\examples\CompactRIO\Module Specific\NI 9211\NI 9211 Support Files.llb for an example of converting calibrated or raw data to temperature. You can use the Convert to Temperature (NI 9211) VI as a subVI in the host VI.

- Open example

Using an Equation to Convert Binary Thermocouple Values to Voltage

You can use the following equation in the host VI to convert binary thermocouple values to voltage:

$$\text{Voltage} = \text{Binary Value} \times 80 \text{ mV} \div 8,388,607$$

where *Binary Value* is the value returned by the FPGA I/O Node.

Using an Equation to Convert Fixed-Point CJC Data to Binary CJC Data

You can use the following equation in the host VI to convert fixed-point CJC data to binary CJC data:

$$\text{Binary CJC Data} = (\text{Fixed-Point CJC Data}) \div (0.160 \div (2^{24} - 1))$$

where *Fixed-Point CJC Data* is the value returned by the FPGA I/O Node.

Using Equations to Convert Binary CJC Data to Temperature

You can use the following equations in the host VI to convert binary CJC data to temperature:

Calculate the resistance of the thermistor:

$$R_T = (10000 \times \text{Binary CJC Data}) \div (2^{23} - \text{Binary CJC Data})$$

Calculate the CJC temperature:

$$T = [1 \div [A + B(\ln(R_T)) + C(\ln(R_T))^3]] - 273.85$$

where T = temperature in °C

$$A = 1.2873851 \times 10^{-3}$$

$$B = 2.3575235 \times 10^{-4}$$

$$C = 9.4978060 \times 10^{-8}$$

Refer to the National Institute of Standards and Technology (NIST) Monograph 175 thermocouple reference tables for more information about converting and adjusting thermocouple values.

NI 9213 (FPGA Interface)

CompactRIO 16-Channel, ± 78 mV, 24-Bit Thermocouple Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
TCx	Thermocouple input channel x , where x is the number of the channel. The NI 9213 has TC channels 0 to 15.
Autozero	Autozero channel. For the best accuracy, read the Autozero channel in the same FPGA I/O Node as the thermocouple input channels.
CJC	Cold-junction compensation channel. For the best accuracy, read the CJC channel in the same FPGA I/O Node as the thermocouple input channels. You must convert the CJC data to temperature.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Check Status	<p>Returns Booleans for each channel that indicate whether the channel is out of range or has an open thermocouple. This method determines the Boolean values for the Out of Range and Open Thermocouple outputs when the FPGA I/O Node reads the channels. A value of TRUE for any channel is cached until the Check Status method executes.</p> <ul style="list-style-type: none">• Force Status Read—When the value of this input is FALSE, the method returns the cached status information since the last time that the Check Status method executed. When the value is TRUE, the method forces the FPGA I/O Node to read all channels and then the method updates the status information. Forcing a status read can introduce jitter into an analog input loop.• Out of Range—Returns an array of Boolean values. A value of TRUE in any index indicates that the channel sharing a number with that index exceeded the common-mode voltage range at some point after the last time that the Check Status method executed.• Open Thermocouple—Returns an array of Boolean values. A value of TRUE in any index indicates that the channel sharing a number with that index detected an open thermocouple on the channel at some point after the last time that the Check Status method executed.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Conversion Time	Sets the time it takes to acquire one point of data from one or more channels in a single FPGA I/O Node. You can select High Speed or High Resolution . Refer to the <i>NI 9213 Operating Instructions and Specifications</i> for more information about the High Speed and High Resolution conversion times. This property overwrites the value you configure in the C Series Module Properties dialog box.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the *NI 9213 Operating Instructions and Specifications* to learn about module specifications and how to use the module. The operating instructions ship with the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9213 (FPGA Interface)

Right-click an [NI 9213](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 24 bits and an integer word length of –2 bits. If you select **Calibrated**, you must [convert](#) the CJC data from voltage to temperature. Select **Raw** if you want the FPGA I/O Node to return calibrated, binary data from the module. If you select **Raw**, you must [convert](#) the binary thermocouple and CJC values to voltage and then convert the CJC data from voltage to temperature. The default is **Calibrated**.
- **Conversion Time**—Specifies the time it takes to acquire one point of data from one or more channels in a single FPGA I/O Node. You can select **High Speed** or **High Resolution**. The default is **High Resolution**. Refer to the *NI 9213 Operating Instructions and Specifications* for more information about the **High Speed** and **High Resolution** conversion times.

Converting NI 9213 Data (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9213](#) if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. If you set the **Calibration Mode** to **Calibrated**, you must convert the CJC data from voltage to temperature. Set the **Calibration Mode** to **Raw** if you want the FPGA I/O Node to return calibrated, binary data from the module. If you set the **Calibration Mode** to **Raw**, you must convert the binary thermocouple and CJC values to voltage and then convert the CJC data from voltage to temperature. You must convert these values in the host VI.

Using a VI to Convert Data to Temperature

Refer to the NI 9213 Convert to Temperature polymorphic VI in the labview\examples\CompactRIO\Module Specific\NI 9213\NI 9213 Getting Started\NI 9213 Getting Started.lvproj for an example of converting calibrated or raw data to temperature. You can use the NI 9213 Convert to Temperature VI as a subVI in the host VI.

- Open example

Using an Equation to Convert Binary Values to Voltage

You can use the following equation in the host VI to convert the binary thermocouple and CJC values to voltage:

$$\text{Voltage} = \text{Binary Value} \times 78.125 \text{ mV} \div 8,388,607$$

where *Binary Value* is the value returned by the FPGA I/O Node.

Using Equations to Convert CJC Data from Voltage to Temperature

You can use the following equations in the host VI to convert CJC data from volts to temperature:

Calculate the resistance of the thermistor:

$$R_T = (10000 \times \text{CJC Data} \times 32) \div (2.5 - \text{CJC Data} \times 32)$$

Calculate the CJC temperature:

$$T = [1 \div [A + B(\ln(R_T)) + C(\ln(R_T))^3]] - 273.15 - 1$$

where T = temperature in °C

$$A = 1.2873851 \times 10^{-3}$$

$$B = 2.3575235 \times 10^{-4}$$

$$C = 9.4978060 \times 10^{-8}$$

Refer to the National Institute of Standards and Technology (NIST) Monograph 175 thermocouple reference tables for more information about converting and adjusting thermocouple values.

NI 9215 (FPGA Interface)

CompactRIO 4-Channel, ± 10 V, 16-Bit Simultaneous Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [simultaneously read from](#) multiple channels on the NI 9215.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x , where x is the number of the channel. The NI 9215 has AI channels 0 to 3.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in nV/LSB for the channel. Use this value to convert and calibrate NI 9215 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9215 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9215 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9215 (FPGA Interface)

Right-click an [NI 9215](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data for the module in units of volts. The fixed-point data is signed, with a word length of 20 bits and an integer word length of 5 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data for the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.

NI 9217 (FPGA Interface)

CompactRIO 4-Channel, 24-Bit, 100 Ω RTD Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
RTDx	RTD input channel x, where x is the number of the channel. The NI 9217 has RTD channels 0 to 3.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in pΩ/LSB for the channel. Use this value to convert and calibrate NI 9217 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in μΩ for the channel. Use this value to convert and calibrate NI 9217 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Conversion Time	Sets the time in milliseconds it takes to acquire conversion data. This property overwrites the value you configure in the C Series Module Properties dialog box.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9217 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9217 (FPGA Interface)

Right-click an [NI 9217](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of ohms. The fixed-point data is signed, with a word length of 24 bits and an integer word length of 10 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Conversion Time**—Specifies the time it takes to acquire one point of data from all channels. You can select **200 ms** or **2.5 ms**.

Converting Nominal Values to Temperature Values for the NI 9217 (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9217](#) if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data for the module in units of ohms. If you set the **Calibration Mode** to **Raw**, the FPGA I/O Node returns uncalibrated, binary values for the module. After you [convert these binary values to nominal values](#), you can convert the nominal resistance values into temperature values. You must convert these values in the host VI.

Using a VI to Convert Values

Refer to the RTD to Temp VI in the labview\examples\CompactRIO\Module Specific\NI 9217\NI 9217 Getting Started\NI 9217 Getting Started.lvproj for an example of converting nominal resistance values to temperature. You can use the RTD to Temp VI as a subVI in the host VI to convert nominal resistance values to temperature.

- Open example

Using an Equation to Measure Temperature

You can use a linearization curve known as the [Callendar-Van Dusen equation](#) in the host VI to measure the temperature of RTDs.

NI 9219 (FPGA Interface)

CompactRIO 4-Channel, 24-Bit Universal Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Chx	Channel x, where x is the number of the channel. The NI 9219 has channels 0 to 3.
CJCx	Cold-junction compensation channel x, where x is the number of the channel. The NI 9219 has CJC channels 0 to 3. If a channel is in Thermocouple mode, you can read CJC data for the channel. You must convert the CJC data to temperature.

You can [read TEDS information](#) from the NI 9219.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Voltage Range	Sets the range of a channel in Voltage mode.
Resistance Range	Sets the range of a channel in Resistance mode.
RTD Range	Sets the range of a channel in RTD mode.
Quarter Bridge Range	Sets the range of a channel in Quarter Bridge mode.
Full Bridge Range	Sets the range of a channel in Full Bridge mode.
Digital Threshold	Sets the threshold of a channel in Digital In mode.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Conversion Time	Sets the time it takes to acquire one point of data from all channels. You can select High Speed , Best 60 Hz Rejection , Best 50 Hz Rejection , or High Resolution . Refer to the NI 9219 Operating Instructions and Specifications for more information about these conversion times. This property overwrites the value you configure in the C Series Module Properties dialog box.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Excitation Protection

The NI 9219 excitation circuit is protected from overcurrent and overvoltage fault conditions. The circuit is automatically disabled in the event of a fault condition. Whenever possible, a channel automatically recovers after the fault is removed. You must wire error terminals on the FPGA I/O Property Nodes to receive notification of overcurrent and overvoltage faults. LabVIEW returns [error 65544 or 65548](#) if there is an overcurrent or overvoltage fault on at least one channel. If a warning occurs, only the channel(s) with the fault are affected and all other channels on the module continue to function properly without interruption. If an error occurs, the module is unable to recover from the fault condition and you must restart the module after the fault is removed.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9219 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Accessing TEDS Information from the NI 9219 (FPGA Interface)

You can access Transducer Electronic Data Sheet (TEDS) information from TEDS-compatible transducers connected to the channels of an [NI 9219](#). You must enable TEDS support for the module before you can read TEDS information. You can access TEDS information only from the host VI.

Enabling TEDS Support for the NI 9219

Complete the following steps to enable TEDS support for the NI 9219.

1. [Configure](#) the CompactRIO system, and add an NI 9219.
2. Right-click the NI 9219 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Place a checkmark in the **Enable TEDS Support** checkbox.
4. Click the **OK** button.

The FPGA VI must have an FPGA I/O Node or an FPGA I/O Property Node that communicates with the NI 9219. If it does not, the LabVIEW FPGA Compile Server optimizes the VI when you compile it, and the host VI cannot communicate with the NI 9219.

Reading TEDS Information in the Host VI

After you develop the FPGA VI and open a reference to it in the host VI, complete the following steps to read TEDS information from the NI 9219.

1. Place an [Invoke Method](#) function on the block diagram.
2. 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.
3. Right-click **Method** on the Invoke Method function and select [Read TEDS](#).
4. If you are using the NI 9219 with an R Series device, right-click the **Connector** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9219.
5. Right-click the **Slot** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9219. Valid slot values are 1 through N , where N is the number of slots in the chassis.
6. Right-click the **Channel** input of the Invoke Method function and select **Create»Constant** from the shortcut menu. Configure the constant for the channel on the NI 9219 for which you want to read TEDS information. Valid channel values are 0 through 3.
7. Place a TEDS_Parse Binary TEDS VI, available on the **TEDS** palette, on the block diagram.

■ Place ■ Find
8. Wire the **TEDS Binary** output of the Invoke Method function to the **binary TEDS array** input of the TEDS_Parse Binary TEDS VI.
9. Wire the **v0.9 (TEDS)** output of the Invoke Method function to the **v0.9 TEDS (F)** input of the TEDS_Data to Table VI.
10. Wire the **TEDS data out** output of the TEDS_Parse Binary VI to the **TEDS data in** input of the TEDS_Data to Table VI.

When you run the VI, LabVIEW reads the TEDS from the sensor, and the LabVIEW TEDS Toolkit parses the TEDS information and outputs it as a

table.



Note You must enable error handling in the FPGA VI to receive consistent error messages in the host VI.

Refer to the NI 9219 TEDS VI in the labview\examples\CompactRIO\Module Specific\NI 9219\NI 9219 TEDS directory for an example of accessing TEDS information.

■ Open example

C Series Module Properties Dialog Box for the NI 9219 (FPGA Interface)

Right-click an [NI 9219](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module. Refer to the table below for information about the units and encoding of the fixed-point data depending on the channel mode. If you select **Calibrated** and an NI 9219 channel is in Thermocouple mode, you must [convert](#) the CJC data in the host VI. Select **Raw** if you want the FPGA I/O Node to return calibrated, binary data from the module. If you select **Raw**, you must [convert](#) the analog input values in the host VI. The default is **Calibrated**.

Channel Mode	Units of Fixed-Point Data	Integer Type	Word Length	Integer Word Length
Voltage	Volts	Signed	32 bits	7 bits
Current	Amps	Signed	32 bits	–1 bits
Resistance and RTD	Ohms	Unsigned	32 bits	14 bits
Thermocouple	Volts	Signed	32 bits	–1 bits
Quarter	Volts/volt	Unsigned	32 bits	14 bits

Bridge				
Half Bridge	Volts/volt	Signed	32 bits	7 bits
Full Bridge	Volts/volt	Signed	32 bits	-1 bits

- **Channels**—Specifies the channel(s) for which you want to select the mode and range.
- **Selected Channel(s) Settings**—Specifies the mode and range for each channel.
 - **Channel Mode**—Sets the mode for the selected channel(s).
 - **Range**—Sets the range for the selected channel(s).
 - **Threshold**—[Sets the Digital In threshold](#) for the selected channel(s). This option is available only if you select **Digital In** for the channel mode.
- **Conversion Time**—Specifies the time it takes to acquire one point of data from all channels. You can select **High Speed**, **Best 60 Hz Rejection**, **Best 50 Hz Rejection**, or **High Resolution**. Refer to the [NI 9219 Operating Instructions and Specifications](#) for more information about these conversion times.
- **Enable TEDS Support**—Place a checkmark in this checkbox if you want to enable [TEDS support](#) in the FPGA and host VIs for this module.

Converting NI 9219 Data (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9219](#) if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module. If you set the **Calibration Mode** to **Calibrated** and an NI 9219 channel is in Thermocouple mode, you must convert the fixed-point CJC data to binary CJC data and then convert the binary CJC data to temperature. Set the **Calibration Mode** to **Raw** if you want the FPGA I/O Node to return binary values from the module that are calibrated based on the [range](#) of the selected mode. If you set the **Calibration Mode** to **Raw**, you must convert the binary values to meaningful engineering units. If you set the **Calibration Mode** to **Raw** and an NI 9219 channel is in Thermocouple mode, you also must convert the binary CJC data to temperature. You must convert these values in the host VI.

Using a VI to Convert and Adjust Binary Values

Refer to the NI 9219 Binary to Nominal VI in the `labview\examples\CompactRIO\Module Specific\NI 9219\NI 9219 Scaling Utility` directory for an example of converting and adjusting binary input values. You can use the polymorphic NI 9219 Binary to Nominal VI as a subVI in the host VI to convert and adjust binary input values.

Using an Equation to Convert Binary Values

You can use the following equation in the host VI to convert the binary input values to engineering units for each channel:

$$\text{Engineering Units}^* = \text{Binary Value} \times (\text{Range High} - \text{Range Low}) \div 2^{24}$$

where *Binary Value* is the value returned by the FPGA I/O Node

Range High is the upper value of the input range †

Range Low is the lower value of the input range.

*Engineering units are equivalent to the units of the mode input range in the [NI 9219 Operating Instructions and Specifications](#).

†Refer to the *NI 9219 Operating Instructions and Specifications* for the input ranges for each mode. For modes that have only one input range value, use 0 as the lower value of the input range.

Digital In

Digital In mode returns a Boolean value, where FALSE is equivalent to any voltage below the specified [threshold](#) and TRUE is equivalent to any voltage above the specified threshold.

Open Contact

Open Contact mode returns a Boolean value, where FALSE indicates a closed circuit and TRUE indicates a open circuit.

Using an Equation to Convert Fixed-Point CJC Data

You can use the following equation in the host VI to convert fixed-point CJC data to binary CJC data:

$$\text{Binary CJC Data} = (\text{Fixed-Point CJC Data}) \div (0.250 \div (2^{24} - 1))$$

Using Equations to Convert Binary CJC Data

You can use the following equations in the host VI to convert binary CJC data to temperature:

Calculate the resistance of the thermistor:

$$R_T = 10000 \div [(2^{16} \div \text{Binary CJC Data}) - 1]$$

Calculate the CJC temperature:

$$T = [1 \div [A + B(\ln(R_T)) + C(\ln(R_T))^3]] - 274.65$$

where T = temperature in °C

$$A = 1.2873851 \times 10^{-3}$$

$$B = 2.3575235 \times 10^{-4}$$

$$C = 9.4978060 \times 10^{-8}$$

Refer to the National Institute of Standards and Technology (NIST) Monograph 175 thermocouple reference tables for more information about converting and adjusting thermocouple values.

Configuring Modes and Ranges for the NI 9219 (FPGA Interface)

You can configure the mode and range for each channel on the NI 9219 at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the range for each channel at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Range** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring Modes and Ranges Using the C Series Module Properties Dialog Box

Complete the following steps to configure the channel(s) mode and range using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9219.
2. Right-click the NI 9219 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Select the channel(s) for which you want to configure the mode and range from the **Channels** table. You can select more than one channel by holding the <Ctrl> or <Shift> key when selecting channels.
4. Select the mode for the channel(s) from the **Channel Mode** pull-down menu. If you select **Digital In** mode, skip the following steps and **set the threshold**. If you select **Open Contact** mode, skip the following step.
5. Select the range from the **Range** pull-down menu for the selected channel(s).
6. Click the **OK** button.
7. Select **File»Save All** in the **Project Explorer** window.

Configuring Ranges Using the FPGA I/O Property Node

Complete the following steps to configure the channel range using the FPGA I/O Property Node.

1. [Create FPGA I/O items](#) for the channel of the NI 9219 for which you want to configure a **Range** property. The channel must be set to Voltage mode, Resistance mode, RTD mode, Quarter Bridge mode, or Full Bridge mode.
2. Place an FPGA I/O Property Node on the block diagram and [configure](#) it for the NI 9219 channel for which you want to configure the range.
3. Click the **Property** section and select the **Range** property from the shortcut menu.
4. Right-click the **Range** input and select **Create»Control** from the shortcut menu.
5. On the front panel of the VI, select a rate from the **Range** pull-down menu.

You can change the channel range at run time by writing to the control from the host VI. Refer to the NI 9219 Getting Started VI in the `labview\examples\CompactRIO\Module Specific\NI 9219\NI 9219 Getting Started` directory for an example of changing a channel range using the FPGA I/O Property Node.

■ [Open example](#)

Configuring the Digital In Threshold for the NI 9219 (FPGA Interface)

Channels in Digital In mode on the NI 9219 have a 0–60 V unipolar threshold. The default value of the Digital In mode threshold is 1.5 V. You can configure the threshold at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the threshold at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Digital Threshold** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring the Digital In Threshold Using the C Series Module Properties Dialog Box

Complete the following steps to configure the Digital In threshold for the NI 9219 using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9219.
2. Right-click the NI 9219 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Select the channel(s) for which you want to configure the Digital In threshold from the **Channels** table. You can select more than one channel by holding the <Ctrl> or <Shift> key when selecting channels.
4. Select **Digital In** from the **Channel Mode** pull-down menu.
5. Enter a value between 0 and 60 in the **Threshold** text box for the selected channel(s).
6. Click the **OK** button.
7. Select **File»Save All** in the **Project Explorer** window.

Configuring the Digital In Threshold Using the FPGA I/O Property Node

Complete the following steps to configure the Digital In threshold using the FPGA I/O Property Node.

1. [Create FPGA I/O items](#) for the channel of the NI 9219 for which you want to configure the **Digital Threshold** property. The channel must be set to Digital In mode.
2. Place an FPGA I/O Property Node on the block diagram and [configure](#) it for the NI 9219 channel for which you want to configure the Digital In threshold.
3. Click the **Property** section and select **Digital Threshold** from the shortcut menu.
4. Right-click the **Digital Threshold** input and select **Create»Control** from the shortcut menu.
5. On the front panel of the VI, enter a binary value in the **Digital Threshold** control. You can use the following equation in the host VI to calculate the threshold value:

$$\text{Threshold (binary)} = \text{Threshold (volts)} \times 2^{24} \div 60$$

You can change the Digital In threshold at run time by writing to the control from the host VI. Refer to the NI 9219 Digital Threshold VI in the labview\examples\CompactRIO\Module Specific\NI 9219\NI 9219 Digital Threshold directory for an example of configuring the NI 9219 Digital In threshold using the FPGA I/O Property Node.

■ Open example

NI 9221 (FPGA Interface)

CompactRIO 8-Channel, ± 60 V, 12-Bit Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9221 has AI channels 0 to 7.

You can [configure the minimum time between conversions](#) and [understand scanning](#) for these channels.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in nV/LSB for the channel. Use this value to convert and calibrate NI 9221 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9221 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9201/9221 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9225 (FPGA Interface)

CompactRIO 3-Channel, 300 V_{rms}, 24-Bit Simultaneous Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [simultaneously read from or write to](#) multiple channels on the NI 9225. You can also [synchronize](#) an NI 9225 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9225 has AI channels 0 to 2. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	Gives access to the onboard clock in the LabVIEW block diagram. The Onboard Clock frequency is 12.8 MHz. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the Onboard Clock of the NI 9225 to access this channel.
Start	<p>Channel that controls when the NI 9225 starts acquiring data. If TRUE is written to the Start channel, the NI 9225 starts acquiring data. When the NI 9225 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9225 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9225\NI 9225 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9225 stops acquiring data. If TRUE is written to the Stop channel, the NI 9225 stops acquiring data. When the NI 9225 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9225 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9225\NI 9225 Getting Started directory for an example of using the Start and</p>

Stop channels.

■ [Open example](#)

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in pV/LSB for the channel. Use this value to convert and calibrate NI 9225 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in uV for the channel. Use this value to convert and calibrate NI 9225 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9225 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9225 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the [NI 9225 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Acquiring Data from an NI 9225/9229/9239 (FPGA Interface)

The [NI 9225/9229/9239](#) has analog input channels that are sampled simultaneously at the [data rate](#) for which you have configured the module. Use the **Start** and **Stop** channels of the NI 9225/9229/9239 to put the module in and out of acquisition mode. In acquisition mode, the NI 9225/9229/9239 can only acquire data. You can use the [FPGA I/O Node](#) to read the data from the module. You cannot perform other operations with the NI 9225/9229/9239, such as accessing properties, while the module is in acquisition mode.



Note You must [create FPGA I/O items](#) for the NI 9225/9229/9239 before you can configure the items using the FPGA I/O Node.

Putting the Module in Acquisition Mode

Configure an FPGA I/O Node with the **Start** channel of the NI 9225/9229/9239. Wire a Boolean constant set to TRUE to the **Start** input of the FPGA I/O Node to send a synchronization pulse to the module. The module starts acquiring data at the data rate you configure.

Reading Data from the Module

When the module starts acquiring data, you can use an FPGA I/O Node to read data from the module. You can connect the AI output of the FPGA I/O Node to various types of functions, including an [FPGA Memory function](#) or an [FPGA FIFO function](#). If you read from multiple channels on the module, place the channels in the same FPGA I/O Node to ensure that the VI reads the data synchronously.

Because the NI 9225/9229/9239 internally acquires data at a specified rate, the FPGA I/O Node does not return data until new data has been acquired by the module. If the NI 9225/9229/9239 did not start acquiring data or stops acquiring data while an FPGA I/O Node is waiting for data from the module, the FPGA I/O Node returns a timeout error.

Exiting Acquisition Mode

Configure an FPGA I/O Node with the **Stop** channel of the NI 9225/9229/9239. Write a TRUE to the **Stop** input. The module is no longer in acquisition mode and now you can access NI 9225/9229/9239 properties.

Examples

Refer to the NI 9225 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9225\NI 9225 Getting Started directory for an example of reading from the NI 9225.

- Open NI 9225 example

Refer to the NI 9229 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9229\NI 9229 Getting Started directory for an example of reading from the NI 9229.

- Open NI 9229 example

Refer to the NI 9239 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9239\NI 9239 Getting Started directory for an example of reading from the NI 9239.

- Open NI 9239 example

C Series Module Properties Dialog Box for the NI 9225, NI 9229, and NI 9239 (FPGA Interface)

Right-click an [NI 9225](#), [NI 9229](#), or [NI 9239](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 24 bits and an integer word length of 10 bits for the NI 9225, 7 bits for the NI 9229, and 5 bits for the NI 9239. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Master Timebase Source**—Specifies the [master timebase source](#) that the module uses.
- **Export Onboard Clock**—Place a checkmark in this checkbox if you want to make this module accessible as a master timebase source to other modules.
- **Data Rate**—Specifies the rate at which the module acquires data.

Configuring the Master Timebase Source for the NI 9225/9229/9239 (FPGA Interface)

By default, the [NI 9225/9229/9239](#) module uses the internal 12.8 MHz Onboard Clock as the master timebase source. In a system with multiple NI 9225/9229/923x modules, you can configure one of the NI 9225/9229/923x modules as the master timebase source (master) and configure the other NI 9225/9229/923x modules to use that master timebase source (slaves). Sharing the same master timebase source enables you to [synchronize multiple NI 9225/9229/923x modules](#). The NI 9225/9229/9239 divides the master timebase source to acquire data at the [data rate](#) you configure.



Note The cRIO-9151 R Series Expansion chassis does not support synchronizing multiple NI 9225/9229/923x modules.

Configuring the Onboard Clock as the Master Timebase Source

Complete the following steps to configure the internal 12.8 MHz Onboard Clock as the master timebase source.

1. [Configure](#) the CompactRIO system, and add an NI 9225/9229/9239.
2. Right-click the NI 9225/9229/9239 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring a Master Module to Share the Master Timebase Source with Slaves

Complete the following steps to configure the master timebase source for a master module and slaves.

1. Configure the CompactRIO system, and add an NI 9225/9229/9239.
2. Right-click the NI 9225/9229/9239 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Place a checkmark in the **Export Onboard Clock** checkbox.
5. Right-click the NI 9225/9229/9239 you want to configure as a slave in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
6. Select the name of the master module from the **Master Timebase Source** pull-down menu.
7. Repeat steps 5 and 6 for each slave module you want to configure.
8. Click the **OK** button.
9. Select **File»Save All** in the **Project Explorer** window.

Configuring the NI 9225/9229/9239 Data Rate (FPGA Interface)

You can configure the data rate at which the [NI 9225/9229/9239](#) module acquires and returns data at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the data rate at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Data Rate** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring the Data Rate Using the C Series Module Properties Dialog Box

Complete the following steps to configure the data rate for the NI 9225/9229/9239 using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9225/9229/9239.
2. Right-click the NI 9225/9229/9239 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select the rate from the **Data Rate** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring the Data Rate Using the FPGA I/O Property Node

Complete the following steps to configure the data rate using the FPGA I/O Property Node.

1. Place an FPGA I/O Property Node on the block diagram and **configure** it for the NI 9225/9229/9239.
2. Click the **Property** section and select **Data Rate** from the shortcut menu.
3. Right-click the **Data Rate** input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select a rate from the **Data Rate** pull-down menu.

You can change the data rate at run time by writing to the control from the host VI. Refer to the NI 9225 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9225\NI 9225 Getting Started directory for an example of configuring the NI 9225 data rate using the FPGA I/O Property Node.

■ Open NI 9225 example

Refer to the NI 9229 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9229\NI 9229 Getting Started directory for an example of configuring the NI 9229 data rate using the FPGA I/O Property Node.

■ Open NI 9229 example

Refer to the NI 9239 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9239\NI 9239 Getting Started directory for an example of configuring the NI 9239 data rate using the FPGA I/O Property Node.

■ Open NI 9239 example



Note The **Data Rate** property returns [error 65538](#) if the module is acquiring data. You must use the **Stop** channel to stop acquiring data before you can write properties to the modules. Refer to the [NI 9225](#), [NI 9229](#), and [NI 9239](#) help topics for more information about module properties.

Synchronizing Multiple NI 9225/9229/923x Modules (FPGA Interface)

You can synchronize multiple NI 9225/9229/923x modules that are connected to the same FPGA device. To synchronize multiple NI 9225/9229/923x modules, the modules must use the same master timebase source, the modules must start acquisition mode at the same time, and a single [FPGA I/O Node](#) function must read the synchronous data. You must [create FPGA I/O items](#) for the NI 9225/9229/923x before you can configure the items using the FPGA I/O Node. Develop the FPGA VI to meet the following guidelines.

-  **Note** [Synchronizing the NI 9234 module with NI 9225/9229/9233/9235/9236/9237/9239 modules](#) depends on the frequency of the shared master timebase.
-  **Note** The cRIO-9151 R Series Expansion chassis does not support synchronizing multiple NI 9225/9229/923x modules.

Sharing a Master Timebase Source

Configure the modules to [share the same master timebase source](#).

Starting the Synchronized Acquisition

Configure an FPGA I/O Node with **Start** channels for the NI 9225/9229/923x modules you want to synchronize and wire a Boolean constant set to TRUE to each **Start** channel.



Note Ensure that all I/O channels are in the same FPGA I/O Node. Otherwise, the FPGA I/O Node will not return synchronized data.

Acquiring Data from Synchronized NI 9225/9229/923x Modules with the Same Data Rate

Configure an FPGA I/O Node with all of the channels from which you want to synchronously sample.



Note Ensure that all I/O channels are in the same FPGA I/O Node. Otherwise, the FPGA I/O Node will not return synchronized data.

Refer to the Synchronizing NI 923x Modules (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 923x\Synchronizing NI 923x Modules directory for an example of synchronizing multiple modules with the same data rate.

■ Open example

Acquiring Data from Synchronized NI 9225/9229/923x Modules with Different Data Rates

If you synchronize NI 9225/9229/923x modules that are [configured for different data rates](#), create a [separate loop](#) for each data rate in the FPGA VI. In each loop, configure an FPGA I/O Node with all of the channels that are configured for the data rate of that loop. If you place NI 9225/9229/923x channels that are configured for different data rates in the same loop, LabVIEW returns an overrun warning from the FPGA I/O Node ([error 65539](#)) when you run the VI. There is a delay before the FPGA I/O Node returns the first data point. The length of the delay depends on the data rate of the NI 9225/9229/923x.

Understanding Loop Timing for the NI 9225/9229/923x (FPGA Interface)

The NI 9225/9229/923x is internally timed. Do not use the [Loop Timer](#) or [Wait](#) functions in a loop with an [FPGA I/O Node](#) that acquires data from an NI 9225/9229/923x. When you create a loop that reads data from an NI 9225/9229/923x, make sure the loop does not execute slower than the data rate of the NI 9225/9229/923x. If the loop execution time is slower than the NI 9225/9229/923x data rate, the FPGA I/O Node returns an overrun warning and continues to read NI 9225/9229/923x data. The overrun warning means that the data the FPGA I/O Node returns is valid, but the function missed one or more data points since the last time it read data from the module. The function returns the overrun warning when all of the following conditions are true:

- The NI 9225/9229/923x is in acquisition mode.
- An FPGA I/O Node that is acquiring data from the NI 9225/9229/923x executes at least once after you put the module in acquisition mode.
- The FPGA I/O Node did not read one or more data points since the previous time the function executed.

If the application acquires multiple buffers of data from an NI 9225/9229/923x and the timing relationship between them is not important, you can ignore the overrun warning returned with the first point of each buffer.

To avoid overrun warnings, develop the FPGA VI to meet the following guidelines:

- **Reading from multiple NI 9225/9229/923x modules in the same loop**—Use one FPGA I/O Node to read the NI 9225/9229/923x channels. You also must configure the NI 9225/9229/923x modules to share a master timebase source and have the same data rate.
- **Reading from the NI 9225/9229/923x and a different analog input module in the same loop**—If the rate at which you can acquire data from the other C Series module is as fast or faster than the data rate configured for the NI 9225/9229/923x, you can read from both modules in the same loop. If you use the same

FPGA I/O Node to read data from the modules, the FPGA I/O Node does not return data for the other module until the NI 9225/9229/923x acquires data. However, if the module has a slower data rate than the NI 9225/9229/923x and you read from the modules in the same loop, the FPGA I/O Node for the NI 9225/9229/923x returns an overrun warning and continues reading data. To avoid missing data, you can either change the data rate of the NI 9225/9229/923x or read from each module in a different loop.

NI 9229 (FPGA Interface)

CompactRIO 4-Channel, ± 60 V, 24-Bit Simultaneous Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [simultaneously read from or write to](#) multiple channels on the NI 9229. You also can [synchronize](#) an NI 9229 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9229 has AI channels 0 to 3. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	Gives access to the onboard clock in the LabVIEW block diagram. The Onboard Clock frequency is 12.8 MHz. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the Onboard Clock of the NI 9229 to access this channel.
Start	<p>Channel that controls when the NI 9229 starts acquiring data. If TRUE is written to the Start channel, the NI 9229 starts acquiring data. When the NI 9229 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9229 Getting Started VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9229\NI 9229 Getting Started</code> directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9229 stops acquiring data. If TRUE is written to the Stop channel, the NI 9229 stops acquiring data. When the NI 9229 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9229 Getting Started VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9229\NI 9229 Getting Started</code> directory for an example of using the Start and</p>

Stop channels.

■ [Open example](#)

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in pV/LSB for the channel. Use this value to convert and calibrate NI 9229 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9229 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9229 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9229 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the [NI 9229/9239 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9233 (FPGA Interface)

CompactRIO 4-Channel, ± 5 V, 24-Bit IEPE Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [synchronize](#) an NI 9233 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9233 has AI channels 0 to 3. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
12.8 MHz Timebase	12.8 MHz internal clock of the NI 9233. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the 12.8 MHz timebase of the NI 9233 to access this channel. If you do not export the 12.8 MHz timebase and try to access the 12.8 MHz Timebase channel, LabVIEW returns a code generation error when you try to compile the FPGA VI.
Start	<p>Channel that controls when the NI 9233 starts acquiring data. If TRUE is written to the Start channel, the NI 9233 starts acquiring data. When the NI 9233 is acquiring data, you must write TRUE to the Stop channel before you can access properties or TEDS information for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9233 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9233\NI 9233 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9233 stops acquiring data. If TRUE is written to the Stop channel, the NI 9233 stops acquiring data. When the NI 9233 is acquiring data, you must write TRUE to the Stop channel before you can access properties or TEDS information for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9233 Getting Started VI in the</p>

labview\examples\CompactRIO\Module Specific\NI 9233\NI 9233
Getting Started directory for an example of using the **Start** and
Stop channels.

■ Open example

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in pV/LSB for the channel. Use this value to convert and calibrate NI 9233 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9233 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9233 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **12.8 MHz Timebase** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9233 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the [NI 9233 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Accessing TEDS Information from the NI 9233 (FPGA Interface)

You can access Transducer Electronic Data Sheet (TEDS) information from TEDS-compatible transducers connected to the channels of an [NI 9233](#). You must enable TEDS support for the module before you can read TEDS information. You can access TEDS information only from the host VI.

Enabling TEDS Support for the NI 9233

Complete the following steps to enable TEDS support for the NI 9233.

1. [Configure](#) the CompactRIO system, and add an NI 9233.
2. Right-click the NI 9233 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Place a checkmark in the **Enable TEDS Support** checkbox.
4. Click the **OK** button.

The FPGA VI must have an FPGA I/O Node that communicates with the NI 9233. If it does not, the LabVIEW FPGA Compile Server optimizes the VI when you compile it, and the host VI cannot communicate with the NI 9233.

Reading TEDS Information in the Host VI

After you develop the FPGA VI and open a reference to it in the host VI, complete the following steps to read TEDS information from the NI 9233.



Note The host VI cannot access TEDS information while the module is in acquisition mode. The NI 9233 must [exit acquisition mode](#) before the host VI can access TEDS information.

1. Place an [Invoke Method](#) function on the block diagram.
2. 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.
3. Right-click **Method** on the Invoke Method function and select [Read TEDS](#).
4. If you are using the NI 9233 with an R Series device, right-click the **Connector** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9233.
5. Right-click the **Slot** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9233. Valid slot values are 1 through N , where N is the number of slots in the chassis.
6. Right-click the **Channel** input of the Invoke Method function and select **Create»Constant** from the shortcut menu. Configure the constant for the channel on the NI 9233 for which you want to read TEDS information. Valid channel values are 0 through 3.
7. Place a TEDS_Parse Binary TEDS VI, available on the **TEDS** palette, on the block diagram.

■ Place ■ Find
8. Wire the **TEDS Binary** output of the Invoke Method function to the **binary TEDS array** input of the TEDS_Parse Binary TEDS VI.
9. Wire the **v0.9 (TEDS)** output of the Invoke Method function to the **v0.9 TEDS (F)** input of the TEDS_Data to Table VI.
10. Wire the **TEDS data out** output of the TEDS_Parse Binary VI to

the **TEDS data in** input of the TEDS_Data to Table VI.

When you run the VI, LabVIEW reads the TEDS from the sensor, and the LabVIEW TEDS Toolkit parses the TEDS information and outputs it as a table.



Note You must enable error handling in the FPGA VI to receive consistent error messages in the host VI.

Refer to the NI 9233 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9233\NI 9233 Getting Started directory for an example of accessing TEDS information.

■ Open example

Acquiring Data from an NI 9233 (FPGA Interface)

The [NI 9233](#) has four analog input channels that are sampled simultaneously at the [data rate](#) for which you have configured the module. Use the **Start** and **Stop** channels of the NI 9233 to put the module in and out of acquisition mode. In acquisition mode, the NI 9233 can only acquire data. You can use the [FPGA I/O Node](#) to read the data from the module. You cannot perform other operations with the NI 9233, such as accessing properties or TEDS information, while the module is in acquisition mode.



Note You must [create FPGA I/O items](#) for the NI 9233 before you can configure the items using the FPGA I/O Node.

Putting the Module in Acquisition Mode

Configure an FPGA I/O Node with the **Start** channel of the NI 9233. Wire a Boolean constant set to TRUE to the **Start** input of the FPGA I/O Node to send a synchronization pulse to the module. The module starts acquiring data at the data rate you configure.

Reading Data from the Module

When the module starts acquiring data, you can use an FPGA I/O Node to read data from the module. You can connect the AI output of the FPGA I/O Node to various types of functions, including an [FPGA Memory function](#) or an [FPGA FIFO function](#). If you read from multiple channels on the module, place the channels in the same FPGA I/O Node to ensure that the VI reads the data synchronously.

Because the NI 9233 internally acquires data at a specified rate, the FPGA I/O Node does not return data until new data has been acquired by the module. If the NI 9233 did not start acquiring data or stops acquiring data while an FPGA I/O Node is waiting for data from the module, the FPGA I/O Node returns a timeout error.

Exiting Acquisition Mode

Configure an FPGA I/O Node with the **Stop** channel of the NI 9233. Write a TRUE to the **Stop** input. The module is no longer in acquisition mode and now you can access NI 9233 properties and TEDS information.

Example

Refer to the NI 9233 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9233\NI 9233 Getting Started directory for an example of reading from the NI 9233.

■ Open example

C Series Module Properties Dialog Box for the NI 9233 (FPGA Interface)

Right-click an [NI 9233](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 24 bits and an integer word length of 4 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Master Timebase Source**—Specifies the [master timebase source](#) that the module uses.
- **Export 12.8 MHz Timebase**—Place a checkmark in this checkbox if you want to make this module accessible as a master timebase source to other modules.
- **Data Rate**—Specifies the rate at which the module acquires data.
- **Enable TEDS Support**—Place a checkmark in this checkbox if you want to enable [TEDS support](#) in the FPGA and host VIs for this module.

Configuring the Master Timebase Source for the NI 9233 (FPGA Interface)

By default, the [NI 9233](#) module uses the internal Onboard Clock as the master timebase source. In a system with multiple NI 923x modules, you can configure one of the NI 923x modules as the master timebase source (master) and configure the other NI 923x modules to use that master timebase source (slaves). Sharing the same master timebase source enables you to [synchronize multiple NI 923x modules](#). The NI 9233 divides the master timebase source to acquire data at the [data rate](#) you configure.



Note The cRIO-9151 R Series Expansion chassis does not support synchronizing multiple NI 9225/9229/923x modules.

Configuring the Onboard Clock as the Master Timebase Source

Complete the following steps to configure the Onboard Clock as the master timebase source.

1. [Configure](#) the CompactRIO system, and add an NI 9233.
2. Right-click the NI 9233 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Select **<12.8 MHz Timebase>** from the **Master Timebase Source** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring a Master Module to Share the Master Timebase Source with Slaves

Complete the following steps to configure the master timebase source for a master module and slaves.

1. Configure the CompactRIO system, and add an NI 9233.
2. Right-click the NI 9233 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select **<12.8 MHz Timebase>** from the **Master Timebase Source** pull-down menu.
4. Place a checkmark in the **Export 12.8 MHz Timebase** checkbox.
5. Right-click the NI 9233 you want to configure as a slave in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
6. Select the name of the master module from the **Master Timebase Source** pull-down menu.
7. Repeat steps 5 and 6 for each slave module you want to configure.
8. Click the **OK** button.
9. Select **File»Save All** in the **Project Explorer** window.

Configuring the NI 9233 Data Rate (FPGA Interface)

You can configure the data rate at which the [NI 9233](#) module acquires and returns data at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the data rate at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Data Rate** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring the Data Rate Using the C Series Module Properties Dialog Box

Complete the following steps to configure the data rate for the NI 9233 using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9233.
2. Right-click the NI 9233 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select the rate from the **Data Rate** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring the Data Rate Using the FPGA I/O Property Node

Complete the following steps to configure the data rate using the FPGA I/O Property Node.

1. Place an FPGA I/O Property Node on the block diagram and **configure** it for the NI 9233.
2. Click the **Property** section and select **Data Rate** from the shortcut menu.
3. Right-click the **Data Rate** input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select a rate from the **Data Rate** pull-down menu.

You can change the data rate at run time by writing to the control from the host VI. Refer to the NI 9233 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9233\NI 9233 Getting Started directory for an example of configuring the data rate using the FPGA I/O Property Node.

■ Open example



Note The **Data Rate** property returns **error 65538** if the module is acquiring data. You must use the **Stop channel** to stop acquiring data before you can write properties to the modules.

NI 9234 (FPGA Interface)

CompactRIO 4-Channel, ± 5 V, 51.2 KS/s, 24-Bit Software Selectable
IEPE and AC/DC Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [synchronize](#) an NI 9234 with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9234 has AI channels 0 to 3. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	13.1072 MHz internal clock of the NI 9234. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the onboard clock of the NI 9234 to access this channel. If you do not export the onboard clock and try to access the Onboard Clock channel, LabVIEW returns a code generation error when you try to compile the FPGA VI.
Start	<p>Channel that controls when the NI 9234 starts acquiring data. If TRUE is written to the Start channel, the NI 9234 starts acquiring data. When the NI 9234 is acquiring data, you must write TRUE to the Stop channel before you can access properties or TEDS information for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9234 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9234\NI 9234 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9234 stops acquiring data. If TRUE is written to the Stop channel, the NI 9234 stops acquiring data. When the NI 9234 is acquiring data, you must write TRUE to the Stop channel before you can access properties or TEDS information for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9234 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9234\NI 9234</p>

Getting Started directory for an example of using the **Start** and **Stop** channels.

■ Open example

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Input Configuration	Sets the input configuration of the corresponding channel to one of three modes: AC coupled, DC coupled, or IEPE AC coupled.
LSB Weight	Returns the LSB weight in pV/LSB for the channel. Use this value to convert and calibrate NI 9234 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9234 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9234 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9234 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the [NI 9234 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Accessing TEDS Information from the NI 9234 (FPGA Interface)

You can access Transducer Electronic Data Sheet (TEDS) information from TEDS-compatible transducers connected to the channels of an [NI 9234](#). You must enable TEDS support for the module before you can read TEDS information. You can access TEDS information only from the host VI.

Enabling TEDS Support for the NI 9234

Complete the following steps to enable TEDS support for the NI 9234.

1. [Configure](#) the CompactRIO system, and add an NI 9234.
2. Right-click the NI 9234 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Place a checkmark in the **Enable TEDS Support** checkbox.
4. Click the **OK** button.

The FPGA VI must have an FPGA I/O Node that communicates with the NI 9234. If it does not, the LabVIEW FPGA Compile Server optimizes the VI when you compile it, and the host VI cannot communicate with the NI 9234.

Reading TEDS Information in the Host VI

After you develop the FPGA VI and open a reference to it in the host VI, complete the following steps to read TEDS information from the NI 9234.



Note The host VI cannot access TEDS information while the module is in acquisition mode. The NI 9234 must [exit acquisition mode](#) before the host VI can access TEDS information.

1. Place an [Invoke Method](#) function on the block diagram.
2. 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.
3. Right-click **Method** on the Invoke Method function and select [Read TEDS](#).
4. If you are using the NI 9234 with an R Series device, right-click the **Connector** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9234.
5. Right-click the **Slot** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9234. Valid slot values are 1 through N , where N is the number of slots in the chassis.
6. Right-click the **Channel** input of the Invoke Method function and select **Create»Constant** from the shortcut menu. Configure the constant for the channel on the NI 9234 for which you want to read TEDS information. Valid channel values are 0 through 3.
7. Place a TEDS_Parse Binary TEDS VI, available on the **TEDS** palette, on the block diagram.

■ Place ■ Find
8. Wire the **TEDS Binary** output of the Invoke Method function to the **binary TEDS array** input of the TEDS_Parse Binary TEDS VI.
9. Wire the **v0.9 (TEDS)** output of the Invoke Method function to the **v0.9 TEDS (F)** input of the TEDS_Data to Table VI.
10. Wire the **TEDS data out** output of the TEDS_Parse Binary VI to

the **TEDS data in** input of the TEDS_Data to Table VI.

When you run the VI, LabVIEW reads the TEDS from the sensor, and the LabVIEW TEDS Toolkit parses the TEDS information and outputs it as a table.



Note You must enable error handling in the FPGA VI to receive consistent error messages in the host VI.

Refer to the NI 9234 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9234\NI 9234 Getting Started directory for an example of accessing TEDS information.

■ Open example

Acquiring Data from an NI 9234 (FPGA Interface)

The [NI 9234](#) has four analog input channels that are sampled simultaneously at the [data rate](#) for which you have configured the module. Use the **Start** and **Stop** channels of the NI 9234 to put the module in and out of acquisition mode. In acquisition mode, the NI 9234 can only acquire data. You can use the [FPGA I/O Node](#) to read the data from the module. You cannot perform other operations with the NI 9234, such as accessing properties or TEDS information, while the module is in acquisition mode.



Note You must [create FPGA I/O items](#) for the NI 9234 before you can configure the items using the FPGA I/O Node.

Putting the Module in Acquisition Mode

Configure an FPGA I/O Node with the **Start** channel of the NI 9234. Wire a Boolean constant set to TRUE to the **Start** input of the FPGA I/O Node to send a synchronization pulse to the module. The module starts acquiring data at the data rate you configure.

Reading Data from the Module

When the module starts acquiring data, you can use an FPGA I/O Node to read data from the module. You can connect the AI output of the FPGA I/O Node to various types of functions, including an [FPGA Memory function](#) or an [FPGA FIFO function](#). If you read from multiple channels on the module, place the channels in the same FPGA I/O Node to ensure that the VI reads the data synchronously.

Because the NI 9234 internally acquires data at a specified rate, the FPGA I/O Node does not return data until new data has been acquired by the module. If the NI 9234 did not start acquiring data or stops acquiring data while an FPGA I/O Node is waiting for data from the module, the FPGA I/O Node returns a timeout error.

Exiting Acquisition Mode

Configure an FPGA I/O Node with the **Stop** channel of the NI 9234. Write a TRUE to the **Stop** input. The module is no longer in acquisition mode and now you can access NI 9234 properties and TEDS information.

Example

Refer to the NI 9234 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9234\NI 9234 Getting Started directory for an example of reading from the NI 9234.

■ Open example

C Series Module Properties Dialog Box for the NI 9234 (FPGA Interface)

Right-click an [NI 9234](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts. The fixed-point data is signed, with a word length of 24 bits and an integer word length of 4 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Master Timebase Source**—Specifies the [master timebase source](#) that the module uses.
- **Export Onboard Clock**—Place a checkmark in this checkbox if you want to make this module accessible as a master timebase source to other modules.
- **Data Rate**—Specifies the rate at which the module acquires data.
- **Enable TEDS Support**—Place a checkmark in this checkbox if you want to enable [TEDS support](#) in the FPGA and host VIs for this module.
- **Channels**—Specifies the channel(s) for which you want to select the input configuration.
- **Selected Channel(s) Settings**—Specifies the input configuration

for each channel.

- **Input Configuration**—Sets the input configuration for the selected channel(s) to one of three modes: AC coupled, DC coupled, or IEPE AC coupled.

Configuring the Master Timebase Source for the NI 9234 (FPGA Interface)

By default, the [NI 9234](#) module uses the internal Onboard Clock as the master timebase source. In a system with multiple NI 923x modules, you can configure one of the NI 923x modules as the master timebase source (master) and configure the other NI 923x modules to use that master timebase source (slaves). Sharing the same master timebase source enables you to [synchronize multiple NI 923x modules](#). The NI 9234 divides the master timebase source to acquire data at the [data rate](#) you configure.



Note The cRIO-9151 R Series Expansion chassis does not support synchronizing multiple NI 9225/9229/923x modules.

Configuring the Onboard Clock as the Master Timebase Source

Complete the following steps to configure the Onboard Clock as the master timebase source.

1. [Configure](#) the CompactRIO system, and add an NI 9234.
2. Right-click the NI 9234 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring a Master Module to Share the Master Timebase Source with Slaves

Complete the following steps to configure the master timebase source for a master module and slaves.

1. Configure the CompactRIO system, and add an NI 9234.
2. Right-click the NI 9234 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Place a checkmark in the **Export Onboard Clock** checkbox.
5. Right-click the NI 9234 you want to configure as a slave in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
6. Select the name of the master module from the **Master Timebase Source** pull-down menu.
7. Repeat steps 5 and 6 for each slave module you want to configure.
8. Click the **OK** button.
9. Select **File»Save All** in the **Project Explorer** window.

Configuring the NI 9234 Channel Input Modes (FPGA Interface)

You can configure the input mode for each channel of the [NI 9234](#) at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the input mode for each channel at run time using the [FPGA I/O Property Node](#). The execution of an FPGA I/O Property Node that is configured with an **Input Configuration** property overwrites the value you configured in the **C Series Module Properties** dialog box for the corresponding channel.

Configuring the Channel Input Modes Using the C Series Module Properties Dialog Box

Complete the following steps to configure the channel input modes using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9234.
2. Right-click the NI 9234 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select the channel(s) for which you want to configure the input mode from the **Channels** table. You can select more than one channel by holding the <Ctrl> or <Shift> key when selecting channels.
4. Select the input mode for the selected channel(s) from the **Input Configuration** pull-down menu.
5. Click the **OK** button.
6. Select **File»Save All** in the **Project Explorer** window.

Configuring the Channel Input Modes Using the FPGA I/O Property Node

Complete the following steps to configure the channel input modes using the FPGA I/O Property Node.

1. [Create FPGA I/O items](#) for the channels of the NI 9234 for which you want to configure the **Input Configuration** property.
2. Place an FPGA I/O Property Node on the block diagram and [configure](#) it for the NI 9234 channel for which you want to configure the input mode.
3. Click the **Property** section and select **Input Configuration** from the shortcut menu.
4. Right-click the **Input Configuration** input and select **Create»Control** from the shortcut menu.
5. On the front panel of the VI, select an input mode for the channel from the **Input Configuration** pull-down menu.

You can change the channel input modes at run time by writing to the control from the host VI. Refer to the NI 9234 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9234\NI 9234 Getting Started directory for an example of configuring the input modes using the FPGA I/O Property Node.

■ Open example



Note The **Input Configuration** property returns [error 65538](#) if the module is acquiring data. You must use the [Stop channel](#) to stop acquiring data before you can write properties to the modules.

Configuring the NI 9234 Data Rate (FPGA Interface)

You can configure the data rate at which the [NI 9234](#) module acquires and returns data at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the data rate at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Data Rate** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring the Data Rate Using the C Series Module Properties Dialog Box

Complete the following steps to configure the data rate for the NI 9234.

1. [Configure](#) the CompactRIO system, and add an NI 9234.
2. Right-click the NI 9234 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select the rate from the **Data Rate** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring the Data Rate Using the FPGA I/O Property Node

Complete the following steps to configure the data rate using the FPGA I/O Property Node.

1. Place an FPGA I/O Property Node on the block diagram and configure it for the NI 9234.
2. Right-click the **Property** section and select **Data Rate** from the shortcut menu.
3. Right-click the **Data Rate** input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select a rate from the **Data Rate** pull-down menu.

You can change the data rate at run time by writing to the control from the host VI. Refer to the NI 9234 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9234\NI 9234 Getting Started directory for an example of configuring the data rate using the FPGA I/O Property Node.

■ Open example



Note The **Data Rate** property returns [error 65538](#) if the module is acquiring data. You must use the [Stop channel](#) to stop acquiring data before you can write properties to the modules.

Synchronizing the NI 9234 Module with NI 9225/9229/9233/9235/9236/9237/9239 Modules (FPGA Interface)

The [NI 9225/9229/9233/9235/9236/9237/9239](#) module has a 12.8 MHz onboard oscillator, while the [NI 9234](#) has a 13.1072 MHz onboard oscillator. All of these modules have delta-sigma ADCs that derive their sampling rates from the frequency of these onboard oscillators. An NI 9234 that uses its own 13.1072 MHz timebase can sample at a maximum rate of 51.2 kS/s ($13.1072 \text{ MHz}/256$), while an NI 9225/9229/9233/9237/9239 that uses its own 12.8 MHz timebase can sample at a maximum rate of 50 kS/s ($12.8 \text{ MHz}/256$), and an NI 9235/9236 that uses its own 12.8 MHz timebase can sample at a maximum rate of 10 kS/s ($12.8 \text{ MHz}/(256 \times 5)$).

Synchronizing the NI 9234 with the NI 9225/9229/9233/9235/9236/9237/9239 depends on the frequency of the shared master timebase. The NI 9234 can sample at a maximum rate of 51.2 kS/s when an NI 9234 is sourcing the master timebase. However, the NI 9234 can sample only at a maximum rate of 50 kS/s when an NI 9225/9229/9233/9235/9236/9237/9239 is sourcing the master timebase. It is important to note that when an NI 9235/9236 is sourcing the master timebase, the NI 9234 can still sample at a maximum rate of 50 kS/s. Furthermore, the NI 9225/9229/9233/9237/9239 can sample at a maximum rate of 50 kS/s when an NI 9225/9229/9233/9235/9236/9237/9239 is sourcing the master timebase. The NI 9235/9236 can sample at a maximum rate of 10 kS/s when an NI 9235/9236 is sourcing the master timebase. The possible sample rates of the NI 9225/9229/9233/9235/9236/9237/9239 change slightly when an NI 9234 is sourcing the master timebase. The NI 9225/9229/9233/9237/9239 can sample at a maximum rate of 51.2 kS/s when an NI 9234 is sourcing the master timebase, and the NI 9235/9236 can sample at a maximum rate of 10.24 kS/s when an NI 9234 is sourcing the master timebase.

The data rate in the [C Series Module Properties](#) dialog box automatically updates to reflect the actual data rate of the module when you change the master timebase source. However, the **Data Rate** property controls or constants on the block diagram do not automatically update when you

change the master timebase source. Therefore, you might need to delete and recreate controls or constants for any **Data Rate** property on the block diagram after changing the master timebase source.

Refer to the *NI 9xxx Operating Instructions and Specifications* for the module for more information about understanding data rates. The latest versions of the operating instructions are online at ni.com/manuals.

NI 9235 (FPGA Interface)

CompactRIO 8-Channel, 24-Bit Quarter-Bridge Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [synchronize](#) an NI 9235 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9235 has AI channels 0 to 7. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	12.8 MHz internal clock of the NI 9235. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the Onboard Clock of the NI 9235 to access this channel.
Start	<p>Channel that controls when the NI 9235 starts acquiring data. If TRUE is written to the Start channel, the NI 9235 starts acquiring data. When the NI 9235 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9235 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9235\NI 9235 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9235 stops acquiring data. If TRUE is written to the Stop channel, the NI 9235 stops acquiring data. When the NI 9235 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9235 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9235\NI 9235 Getting Started directory for an example of using the Start and Stop channels.</p>

■ Open example

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in units of 100 fV/V per LSB for the channel. 100 fV is equal to 10^{-13} volts. Use this value to convert and calibrate NI 9235 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in units of 100 nV/V for the channel. Use this value to convert and calibrate NI 9235 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Shunt Cal Enable	Controls the shunt calibration switch for each channel. Refer to the NI 9235 Getting Started VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9235\NI 9235 Getting Started</code> directory for an example of using the Shunt Cal Enable property. <input type="checkbox"/> Open example

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9235 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9235 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the *NI 9235/9236 Operating Instructions and Specifications* to learn about module specifications and how to use the module. The operating instructions ship with the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Acquiring Data from an NI 9235/9236 (FPGA Interface)

The [NI 9235](#) or [NI 9236](#) has eight analog input channels that are sampled simultaneously at the [data rate](#) for which you have configured the module. Use the **Start** and **Stop** channels of the NI 9235/9236 to put the module in and out of acquisition mode. In acquisition mode, the NI 9235/9236 can only acquire data. You can use the [FPGA I/O Node](#) to read the data from the module. You cannot perform other operations with the NI 9235/9236, such as accessing properties, while the module is in acquisition mode.



Note You must [create FPGA I/O items](#) for the NI 9235/9236 before you can configure the items using the FPGA I/O Node.

Putting the Module in Acquisition Mode

Configure an FPGA I/O Node with the **Start** channel of the NI 9235/9236. Wire a Boolean constant set to TRUE to the **Start** input of the FPGA I/O Node to send a synchronization pulse to the module. The module starts acquiring data at the data rate you configure.

Reading Data from the Module

When the module starts acquiring data, you can use an FPGA I/O Node to read data from the module. You can connect the AI output of the FPGA I/O Node to various types of functions, including an [FPGA Memory function](#) or an [FPGA FIFO function](#). If you read from multiple channels on the module, place the channels in the same FPGA I/O Node to ensure that the VI reads the data synchronously.

Because the NI 9235/9236 internally acquires data at a specified rate, the FPGA I/O Node does not return data until new data has been acquired by the module. If the NI 9235/9236 did not start acquiring data or stops acquiring data while an FPGA I/O Node is waiting for data from the module, the FPGA I/O Node returns a timeout error.

Exiting Acquisition Mode

Configure an FPGA I/O Node with the **Stop** channel of the NI 9235/9236. Write a TRUE to the **Stop** input. The module is no longer in acquisition mode and now you can access NI 9235/9236 properties.

Example

Refer to the NI 9235 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9235\NI 9235 Getting Started directory for an example of reading from the NI 9235.

- Open NI 9235 example

Refer to the NI 9236 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9236\NI 9236 Getting Started directory for an example of reading from the NI 9236.

- Open NI 9236 example

C Series Module Properties Dialog Box for the NI 9235 and NI 9236 (FPGA Interface)

Right-click an [NI 9235](#) or [NI 9236](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts/volt. The fixed-point data is signed, with a word length of 24 bits and an integer word length of –4 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Master Timebase Source**—Specifies the [master timebase source](#) that the module uses.
- **Export Onboard Clock**—Place a checkmark in this checkbox if you want to make this module accessible as a master timebase source to other modules.
- **Data Rate**—Specifies the rate at which the module acquires data.

Configuring the Master Timebase Source for the NI 9235/9236 (FPGA Interface)

By default, the [NI 9235](#) or [NI 9236](#) module uses the internal Onboard Clock as the master timebase source. In a system with multiple NI 923x modules, you can configure one of the NI 923x modules as the master timebase source (master) and configure the other NI 923x modules to use that master timebase source (slaves). Sharing the same master timebase source enables you to [synchronize multiple NI 923x modules](#). The NI 9235/9236 divides the master timebase source to acquire data at the [data rate](#) you configure.



Note The cRIO-9151 R Series Expansion chassis does not support synchronizing multiple NI 9225/9229/923x modules.

Configuring the Onboard Clock as the Master Timebase Source

Complete the following steps to configure the Onboard Clock as the master timebase source.

1. [Configure](#) the CompactRIO system, and add an NI 9235/9236.
2. Right-click the NI 9235/9236 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring a Master Module to Share the Master Timebase Source with Slaves

Complete the following steps to configure the master timebase source for a master module and slaves.

1. Configure the CompactRIO system, and add an NI 9235/9236.
2. Right-click the NI 9235/9236 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Place a checkmark in the **Export Onboard Clock** checkbox.
5. Right-click the NI 9235/9236 you want to configure as a slave in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
6. Select the name of the master module from the **Master Timebase Source** pull-down menu.
7. Repeat steps 5 and 6 for each slave module you want to configure.
8. Click the **OK** button.
9. Select **File»Save All** in the **Project Explorer** window.

Configuring the NI 9235/9236 Data Rate (FPGA Interface)

You can configure the data rate at which the [NI 9235](#) or [NI 9236](#) module acquires and returns data at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the data rate at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Data Rate** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring the Data Rate Using the C Series Module Properties Dialog Box

Complete the following steps to configure the data rate for the NI 9235/9236 using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9235/9236.
2. Right-click the NI 9235/9236 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select the rate from the **Data Rate** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring the Data Rate Using the FPGA I/O Property Node

Complete the following steps to configure the data rate using the FPGA I/O Property Node.

1. Place an FPGA I/O Property Node on the block diagram and [configure](#) it for the NI 9235/9236.
2. Click the **Property** section and select **Data Rate** from the shortcut menu.
3. Right-click the **Data Rate** input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select a rate from the **Data Rate** pull-down menu.

You can change the data rate at run time by writing to the control from the host VI. Refer to the NI 9235 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9235\NI 9235 Getting Started directory for an example of configuring the NI 9235 data rate using the FPGA I/O Property Node.

■ Open example

Refer to the NI 9236 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9236\NI 9236 Getting Started directory for an example of configuring the NI 9236 data rate using the FPGA I/O Property Node.

■ Open example



Note The **Data Rate** property returns [error 65538](#) if the module is acquiring data. You must use the [Stop channel](#) to stop acquiring data before you can write properties to the modules.

NI 9236 (FPGA Interface)

CompactRIO 8-Channel, 24-Bit Quarter-Bridge Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [synchronize](#) an NI 9236 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9236 has AI channels 0 to 7. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	12.8 MHz internal clock of the NI 9236. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the Onboard Clock of the NI 9236 to access this channel.
Start	<p>Channel that controls when the NI 9236 starts acquiring data. If TRUE is written to the Start channel, the NI 9236 starts acquiring data. When the NI 9236 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9236 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9236\NI 9236 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9236 stops acquiring data. If TRUE is written to the Stop channel, the NI 9236 stops acquiring data. When the NI 9236 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9236 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9236\NI 9236 Getting Started directory for an example of using the Start and Stop channels.</p>

■ Open example

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in units of 100 fV/V per LSB for the channel. 100 fV is equal to 10^{-13} volts. Use this value to convert and calibrate NI 9236 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in units of 100 nV/V for the channel. Use this value to convert and calibrate NI 9236 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Shunt Cal Enable	Controls the shunt calibration switch for each channel. Refer to the NI 9236 Getting Started VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9236\NI 9236 Getting Started</code> directory for an example of using the Shunt Cal Enable property. <input type="checkbox"/> Open example

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9236 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9236 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the *NI 9235/9236 Operating Instructions and Specifications* to learn about module specifications and how to use the module. The operating instructions ship with the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9237 (FPGA Interface)

CompactRIO 4-Channel, 24-Bit Half/Full-Bridge Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [synchronize](#) an NI 9237 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9237 has AI channels 0 to 3. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	12.8 MHz internal clock of the NI 9237. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the Onboard Clock of the NI 9237 to access this channel.
Start	<p>Channel that controls when the NI 9237 starts acquiring data. If TRUE is written to the Start channel, the NI 9237 starts acquiring data. When the NI 9237 is acquiring data, you must write TRUE to the Stop channel before you can access properties or TEDS information for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9237 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9237\NI 9237 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9237 stops acquiring data. If TRUE is written to the Stop channel, the NI 9237 stops acquiring data. When the NI 9237 is acquiring data, you must write TRUE to the Stop channel before you can access properties or TEDS information for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9237 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9237\NI 9237 Getting Started directory for an example of using the Start and Stop channels.</p>

■ Open example

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Half-Bridge Enable	Controls the half-bridge completion option for each channel.
LSB Weight	Returns the LSB weight in units of 100 fV/V per LSB for the channel. 100 fV is equal to 10^{-13} volts. Use this value to convert and calibrate NI 9237 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in units of 10 nV/LSB for the channel. Use this value to convert and calibrate NI 9237 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset Cal Enable	Enables offset calibration. This disconnects both signal input pins and forces the channel inputs to zero.
Shunt Cal Enable	Controls the shunt calibration switch for each channel.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9237 acquires data.
Excitation Voltage	Sets the excitation voltage level. All channels share the same excitation voltage.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9237 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the [NI 9237 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Accessing TEDS Information from the NI 9237 (FPGA Interface)

You can access Transducer Electronic Data Sheet (TEDS) information from TEDS-compatible transducers connected to the channels of an [NI 9237](#). You must enable TEDS support for the module before you can read TEDS information. You can access TEDS information only from the host VI.

Enabling TEDS Support for the NI 9237

Complete the following steps to enable TEDS support for the NI 9237.

1. [Configure](#) the CompactRIO system, and add an NI 9237.
2. Right-click the NI 9237 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Place a checkmark in the **Enable TEDS Support** checkbox.
4. Click the **OK** button.

The FPGA VI must have an FPGA I/O Node or an FPGA I/O Property Node that communicates with the NI 9237. If it does not, the LabVIEW FPGA Compile Server optimizes the VI when you compile it, and the host VI cannot communicate with the NI 9237.

Reading TEDS Information in the Host VI

After you develop the FPGA VI and open a reference to it in the host VI, complete the following steps to read TEDS information from the NI 9237.



Note The host VI cannot access TEDS information while the module is in acquisition mode. The NI 9237 must [exit acquisition mode](#) before the host VI can access TEDS information.

1. Place an [Invoke Method](#) function on the block diagram.
2. 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.
3. Right-click **Method** on the Invoke Method function and select [Read TEDS](#).
4. If you are using the NI 9237 with an R Series device, right-click the **Connector** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9237.
5. Right-click the **Slot** input of the Invoke Method function and select **Create»Constant** or **Create»Control** from the shortcut menu. Configure the constant or control for the slot of the NI 9237. Valid slot values are 1 through N , where N is the number of slots in the chassis.
6. Right-click the **Channel** input of the Invoke Method function and select **Create»Constant** from the shortcut menu. Configure the constant for the channel on the NI 9237 for which you want to read TEDS information. Valid channel values are 0 through 3.
7. Place a TEDS_Parse Binary TEDS VI, available on the **TEDS** palette, on the block diagram.

■ Place ■ Find
8. Wire the **TEDS Binary** output of the Invoke Method function to the **binary TEDS array** input of the TEDS_Parse Binary TEDS VI.
9. Wire the **v0.9 (TEDS)** output of the Invoke Method function to the **v0.9 TEDS (F)** input of the TEDS_Data to Table VI.
10. Wire the **TEDS data out** output of the TEDS_Parse Binary VI to

the **TEDS data in** input of the TEDS_Data to Table VI.

When you run the VI, LabVIEW reads the TEDS from the sensor, and the LabVIEW TEDS Toolkit parses the TEDS information and outputs it as a table.



Note You must enable error handling in the FPGA VI to receive consistent error messages in the host VI.

Refer to the NI 9237 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9237\NI 9237 Getting Started directory for an example of accessing TEDS information.

■ Open example

Acquiring Data from an NI 9237 (FPGA Interface)

The [NI 9237](#) has four analog input channels that are sampled simultaneously at the [data rate](#) for which you have configured the module. Use the **Start** and **Stop** channels of the NI 9237 to put the module in and out of acquisition mode. In acquisition mode, the NI 9237 can only acquire data. You can use the [FPGA I/O Node](#) to read the data from the module. You cannot perform other operations with the NI 9237, such as accessing properties or TEDS information, while the module is in acquisition mode.



Note You must [create FPGA I/O items](#) for the NI 9237 before you can configure the items using the FPGA I/O Node.

Putting the Module in Acquisition Mode

Configure an FPGA I/O Node with the **Start** channel of the NI 9237. Wire a Boolean constant set to TRUE to the **Start** input of the FPGA I/O Node to send a synchronization pulse to the module. The module starts acquiring data at the data rate you configure.

Reading Data from the Module

When the module starts acquiring data, you can use an FPGA I/O Node to read data from the module. You can connect the AI output of the FPGA I/O Node to various types of functions, including an [FPGA Memory function](#) or an [FPGA FIFO function](#). If you read from multiple channels on the module, place the channels in the same FPGA I/O Node to ensure that the VI reads the data synchronously.

Because the NI 9237 internally acquires data at a specified rate, the FPGA I/O Node does not return data until new data has been acquired by the module. If the NI 9237 did not start acquiring data or stops acquiring data while an FPGA I/O Node is waiting for data from the module, the FPGA I/O Node returns a timeout error.

Exiting Acquisition Mode

Configure an FPGA I/O Node with the **Stop** channel of the NI 9237. Write a TRUE to the **Stop** input. The module is no longer in acquisition mode and now you can access NI 9237 properties and TEDS information.

Example

Refer to the NI 9237 Getting Started (FPGA) VI in the labview\examples\CompactRIO\Module Specific\NI 9237\NI 9237 Getting Started directory for an example of reading from the NI 9237.

■ Open example

C Series Module Properties Dialog Box for the NI 9237 (FPGA Interface)

Right-click an [NI 9237](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to return calibrated, [fixed-point](#) data from the module in units of volts/volt. The fixed-point data is signed, with a word length of 24 bits and an integer word length of –4 bits. Select **Raw** if you want the FPGA I/O Node to return uncalibrated, binary data from the module. If you select **Raw**, you must [convert and calibrate](#) the analog input values in the host VI. The default is **Calibrated**.
- **Master Timebase Source**—Specifies the [master timebase source](#) that the module uses.
- **Export Onboard Clock**—Place a checkmark in this checkbox if you want to make this module accessible as a master timebase source to other modules.
- **Data Rate**—Specifies the rate at which the module acquires data.
- **Enable TEDS Support**—Place a checkmark in this checkbox if you want to enable [TEDS support](#) in the FPGA and host VIs for this module.
- **Excitation Voltage**—Specifies the excitation voltage for the module to output to bridges, or specifies external excitation.
- **Enable Half-Bridge Completion**—Enables half-bridge

completion for individual channels.

Configuring the Master Timebase Source for the NI 9237 (FPGA Interface)

By default, the [NI 9237](#) module uses the internal Onboard Clock as the master timebase source. In a system with multiple NI 923x modules, you can configure one of the NI 923x modules as the master timebase source (master) and configure the other NI 923x modules to use that master timebase source (slaves). Sharing the same master timebase source enables you to [synchronize multiple NI 923x modules](#). The NI 9237 divides the master timebase source to acquire data at the [data rate](#) you configure.



Note The cRIO-9151 R Series Expansion chassis does not support synchronizing multiple NI 9225/9229/923x modules.

Configuring the Onboard Clock as the Master Timebase Source

Complete the following steps to configure the Onboard Clock as the master timebase source.

1. [Configure](#) the CompactRIO system, and add an NI 9237.
2. Right-click the NI 9237 in the **Project Explorer** window and select **Properties** to display the [C Series Module Properties](#) dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring a Master Module to Share the Master Timebase Source with Slaves

Complete the following steps to configure the master timebase source for a master module and slaves.

1. Configure the CompactRIO system, and add an NI 9237.
2. Right-click the NI 9237 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select **<Onboard Clock>** from the **Master Timebase Source** pull-down menu.
4. Place a checkmark in the **Export Onboard Clock** checkbox.
5. Right-click the NI 9237 you want to configure as a slave in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
6. Select the name of the master module from the **Master Timebase Source** pull-down menu.
7. Repeat steps 5 and 6 for each slave module you want to configure.
8. Click the **OK** button.
9. Select **File»Save All** in the **Project Explorer** window.

Configuring the NI 9237 Data Rate (FPGA Interface)

You can configure the data rate at which the [NI 9237](#) module acquires and returns data at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the data rate at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Data Rate** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring the Data Rate Using the C Series Module Properties Dialog Box

Complete the following steps to configure the data rate for the NI 9237 using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9237.
2. Right-click the NI 9237 in the **Project Explorer** window and select **Properties** to display the **C Series Module Properties** dialog box.
3. Select the rate from the **Data Rate** pull-down menu.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring the Data Rate Using the FPGA I/O Property Node

Complete the following steps to configure the data rate using the FPGA I/O Property Node.

1. Place an FPGA I/O Property Node on the block diagram and **configure** it for the NI 9237.
2. Click the **Property** section and select **Data Rate** from the shortcut menu.
3. Right-click the **Data Rate** input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select a rate from the **Data Rate** pull-down menu.

You can change the data rate at run time by writing to the control from the host VI. Refer to the NI 9237 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9237\NI 9237 Getting Started directory for an example of configuring the data rate using the FPGA I/O Property Node.

■ Open example



Note The **Data Rate** property returns **error 65538** if the module is acquiring data. You must use the **Stop channel** to stop acquiring data before you can write properties to the modules.

NI 9239 (FPGA Interface)

CompactRIO 4-Channel, ± 10 V, 24-Bit Simultaneous Analog Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.



Note You can [simultaneously read from or write to](#) multiple channels on the NI 9239. You also can [synchronize](#) an NI 9239 module with other NI 9225/9229/923x modules.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AIx	Analog input channel x, where x is the number of the channel. The NI 9239 has AI channels 0 to 3. Do not access AI channels on multiple modules in the same FPGA I/O Node if the modules are not synchronized or do not use the same data rate.
Onboard Clock	Gives access to the onboard clock in the LabVIEW block diagram. The Onboard Clock frequency is 12.8 MHz. Use the FPGA I/O Node in a single-cycle Timed Loop to access this channel. You must export the Onboard Clock of the NI 9239 to access this channel.
Start	<p>Channel that controls when the NI 9239 starts acquiring data. If TRUE is written to the Start channel, the NI 9239 starts acquiring data. When the NI 9239 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Start channel, no operation is performed.</p> <p>Refer to the NI 9239 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9239\NI 9239 Getting Started directory for an example of using the Start and Stop channels.</p> <p>■ Open example</p>
Stop	<p>Channel that controls when the NI 9239 stops acquiring data. If TRUE is written to the Stop channel, the NI 9239 stops acquiring data. When the NI 9239 is acquiring data, you must write TRUE to the Stop channel before you can access properties for the module. If FALSE is written to the Stop channel, no operation is performed.</p> <p>Refer to the NI 9239 Getting Started VI in the labview\examples\CompactRIO\Module Specific\NI 9239\NI 9239 Getting Started directory for an example of using the Start and</p>

Stop channels.

■ [Open example](#)

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in pV/LSB for the channel. Use this value to convert and calibrate NI 9239 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert and calibrate NI 9239 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Data Rate	Sets the rate at which the NI 9239 acquires data.
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

You can use the **Onboard Clock** channel in the single-cycle Timed Loop. You cannot use the other channels on the NI 9239 with the single-cycle Timed Loop. Refer to the [Understanding Loop Timing for the NI 9225/9229/923x](#) topic for information about loop timing for this module.

Hardware Documentation

Refer to the [NI 9229/9239 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9263 (FPGA Interface)

CompactRIO 4-Channel, ± 10 V, 16-Bit Simultaneous Analog Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.



Note You can [simultaneously write to](#) multiple channels on the NI 9263.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AOx	Analog output channel x, where x is the number of the channel. The NI 9263 has AO channels 0 to 3.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any I/O methods.

You should use these three methods in conjunction with each other. The Write Data method stores data values before writing the data to a channel, the Update method writes the stored data values to the output channel, and the Wait for Update method waits until the Update method updates the output channels.

Refer to the NI 9264 Methods VI in the labview\examples\CompactRIO\Module Specific\NI 9264\NI 9264 Methods directory for an example of using these methods.

■ Open example

Method	Input/Output	Description
Update	Overflow	Returns a Boolean value. A value of TRUE indicates an overflow warning. The overflow warning means that the Update method is trying to run while the Write Data method is running, typically in a different loop. Either the Write Data method is writing data to the module too quickly or the Update method is running too slowly.
	Underflow	Returns a Boolean value. A value of TRUE indicates an underflow warning. The underflow warning means that the Write Data method has not written new data to the module since the last time the Update method ran. Either the Write Data method is writing data to the module too slowly or the Update method is running too quickly.
Wait for Update	Timeout (Ticks)	Specifies in FPGA clock ticks how

		long the Wait for Update method waits for the Update method to update the output channels. A value of 0 causes the Wait for Update method to timeout immediately, a negative value causes the Wait for Update method to wait indefinitely, and a positive value causes the Wait for Update method to wait for that number of clock ticks before timing out.
	Timeout Occurred?	Returns a Boolean value. A value of TRUE indicates that a timeout occurred.
Write Data	Channel	Specifies the channel to which you want to write data.
	Data	Specifies the data you want to write to the channel.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in nV/LSB for the channel. Use this value to convert NI 9263 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert NI 9263 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9263 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9263 (FPGA Interface)

Right-click an [NI 9263](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to accept [fixed-point](#) data in units of volts when writing to the module. The fixed-point data is signed, with a word length of 20 bits and an integer word length of 5 bits. Select **Raw** if you want the FPGA I/O Node to accept calibrated, binary data when writing to the module. If you select **Raw**, you must [convert](#) the analog output values in the host VI before you write them to the module. The default is **Calibrated**.

Converting Voltage Values to Binary Values for the NI 9263 (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9263](#) if you want the [FPGA I/O Node](#) to accept [fixed-point](#) data in units of volts when writing to the module. If you set the **Calibration Mode** to **Raw**, the FPGA I/O Node accepts only binary values when writing to the module. You must convert output voltage values to binary values before you write them to the module. You must convert these values in the host VI.

Using a VI to Convert Voltage to Binary

Refer to the Nominal to Binary VI in the labview\examples\CompactRIO\Basic IO\Analog Raw Host Calibration\AO Raw Host Calibration\AO Raw Host Calibration - cRIO.lvproj for an example of converting analog output values to binary values. You can use the Nominal to Binary VI as a subVI in the host VI.

■ Open example

Using an Equation to Convert Voltage to Binary

You can use the following equation in the host VI to convert the analog output values to binary values:

$$\text{Binary Value} = (\text{Voltage Value} \times 10^9 - \text{Offset}) \div \text{LSB Weight},$$

where *Binary Value* is the value you write to the FPGA I/O Node

Voltage Value is the voltage in V that you want the channel to output

Offset is the value returned by the Offset property

LSB Weight is the value returned by the LSB Weight property.

National Instruments recommends using calibrated values for analog output. To convert calibrated voltage values, use the [FPGA I/O Property Node](#) to read the LSB Weight and Offset properties. If you do not want to read the LSB Weight and Offset values from the module, you can convert uncalibrated voltage values by using the following values for *Offset* and *LSB Weight*:

$$\text{Offset} = 0$$

$$\text{LSB Weight} = 21.4 \text{ V} \div 2^{\text{DAC Resolution}} \times 10^9$$

where *DAC Resolution* is the DAC resolution value in the [NI 9263 Operating Instructions and Specifications](#).

NI 9264 (FPGA Interface)

CompactRIO 16-Channel, ± 10 V, 16-Bit Simultaneous Analog Voltage Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.



Note You can [simultaneously write to](#) multiple channels on the NI 9264.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AOx	Analog output channel x, where x is the number of the channel. The NI 9264 has AO channels 0 to 15.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any I/O methods.

You should use these three methods in conjunction with each other. The Write Data method stores data values before writing the data to a channel, the Update method writes the stored data values to the output channel, and the Wait for Update method waits until the Update method updates the output channels.

Refer to the NI 9264 Methods VI in the labview\examples\CompactRIO\Module Specific\NI 9264\NI 9264 Methods directory for an example of using these methods.

■ Open example

Method	Input/Output	Description
Update	Overflow	Returns a Boolean value. A value of TRUE indicates an overflow warning. The overflow warning means that the Update method is trying to run while the Write Data method is running, typically in a different loop. Either the Write Data method is writing data to the module too quickly or the Update method is running too slowly.
	Underflow	Returns a Boolean value. A value of TRUE indicates an underflow warning. The underflow warning means that the Write Data method has not written new data to the module since the last time the Update method ran. Either the Write Data method is writing data to the module too slowly or the Update method is running too quickly.
Wait for Update	Timeout (Ticks)	Specifies in FPGA clock ticks how

		long the Wait for Update method waits for the Update method to update the output channels. A value of 0 causes the Wait for Update method to timeout immediately, a negative value causes the Wait for Update method to wait indefinitely, and a positive value causes the Wait for Update method to wait for that number of clock ticks before timing out.
	Timeout Occurred?	Returns a Boolean value. A value of TRUE indicates that a timeout occurred.
Write Data	Channel	Specifies the channel to which you want to write data.
	Data	Specifies the data you want to write to the channel.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in nV/LSB for the channel. Use this value to convert NI 9264 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in nV for the channel. Use this value to convert NI 9264 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9264 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9264 (FPGA Interface)

Right-click an [NI 9264](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to accept [fixed-point](#) data in units of volts when writing to the module. The fixed-point data is signed, with a word length of 20 bits and an integer word length of 5 bits. Select **Raw** if you want the FPGA I/O Node to accept calibrated, binary data when writing to the module. If you select **Raw**, you must [convert](#) the analog output values in the host VI before you write them to the module. The default is **Calibrated**.

Converting Voltage Values to Binary Values for the NI 9264 (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9264](#) if you want the [FPGA I/O Node](#) to accept [fixed-point](#) data in units of volts when writing to the module. If you set the **Calibration Mode** to **Raw**, the FPGA I/O Node accepts only binary values when writing to the module. You must convert output voltage values to binary values before you write them to the module. You must convert these values in the host VI.

Using a VI to Convert Voltage to Binary

Refer to the Nominal to Binary VI in the labview\examples\CompactRIO\Basic IO\Analog Raw Host Calibration\AO Raw Host Calibration\AO Raw Host Calibration - cRIO.lvproj for an example of converting analog output values to binary values. You can use the Nominal to Binary VI as a subVI in the host VI.

■ Open example

Using an Equation to Convert Voltage to Binary

You can use the following equation in the host VI to convert the analog output values to binary values:

$$\text{Binary Value} = (\text{Voltage Value} \times 10^9 - \text{Offset}) \div \text{LSB Weight},$$

where *Binary Value* is the value you write to the FPGA I/O Node

Voltage Value is the voltage in V that you want the channel to output

Offset is the value returned by the Offset property

LSB Weight is the value returned by the LSB Weight property.

National Instruments recommends using calibrated values for analog output. To convert calibrated voltage values, use the [FPGA I/O Property Node](#) to read the LSB Weight and Offset properties. If you do not want to read the LSB Weight and Offset values from the module, you can convert uncalibrated voltage values by using the following values for *Offset* and *LSB Weight*:

$$\text{Offset} = 0$$

$$\text{LSB Weight} = 21 \text{ V} \div 2^{\text{DAC Resolution}} \times 10^9$$

where *DAC Resolution* is the DAC resolution value in the [NI 9264 Operating Instructions and Specifications](#).

NI 9265 (FPGA Interface)

CompactRIO 4-Channel, 0–20 mA, 16-Bit Simultaneous Analog Current Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.



Note You can [simultaneously write to](#) multiple channels on the NI 9265.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
AOx	Analog output channel x, where x is the number of the channel. The NI 9265 has AO channels 0 to 3.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any I/O methods.

You should use the Update, Wait for Update, and Write Data methods in conjunction with each other. The Write Data method stores data values before writing the data to a channel, the Update method writes the stored data values to the output channel, and the Wait for Update method waits until the Update method updates the output channels.

Refer to the NI 9264 Methods VI in the labview\examples\CompactRIO\Module Specific\NI 9264\NI 9264 Methods directory for an example of using these three methods.

■ Open example

Method	Input/Output	Description
Check Output Status	Force Status Read	When the value of this input is FALSE, the method returns the cached status information since the last status read. When the value is TRUE, the method returns the current status information from the module. Forcing a status read can introduce jitter into an analog output loop.
	Any Fault	Returns a Boolean value. A value of TRUE indicates a power-supply fault or open current loop on at least one channel.
	Power-Supply Fault	Returns a Boolean value. A value of TRUE indicates that the external power supply is out of the expected range.
	Open Current Loop	Returns an array of Boolean values. A value of TRUE in any index indicates that the channel sharing a number with that index has an open current loop and is

		configured to output a nonzero current value.
Update	Overflow	Returns a Boolean value. A value of TRUE indicates an overflow warning. The overflow warning means that the Update method is trying to run while the Check Output Status or Write Data method is running, typically in a different loop. The Update method cannot run while either the Check Output Status or Write Data method is running. If the Update method is running at the same time as the Write Data method, then either the Write Data method is writing data to the module too quickly or the Update method is running too slowly.
	Underflow	Returns a Boolean value. A value of TRUE indicates an underflow warning. The underflow warning means that the Write Data method has not written new data to the module since the last time the Update method ran. Either the Write Data method is writing data to the module too slowly or the Update method is running too quickly.
Wait for Update	Timeout (Ticks)	Specifies in FPGA clock ticks how long the Wait for Update method waits for the Update method to update the output channels. A value of 0 causes the Wait for Update method to timeout immediately, a negative value causes the Wait for Update method to wait indefinitely,

		and a positive value causes the Wait for Update method to wait for that number of clock ticks before timing out.
	Timeout Occurred?	Returns a Boolean value. A value of TRUE indicates that a timeout occurred.
Write Data	Channel	Specifies the channel to which you want to write data.
	Data	Specifies the data you want to write to the channel.

I/O Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
LSB Weight	Returns the LSB weight in pA/LSB for the channel. Use this value to convert NI 9265 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.
Offset	Returns the calibration offset in pA for the channel. Use this value to convert NI 9265 data if you set the Calibration Mode to Raw in the C Series Module Properties dialog box.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9265 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9265 (FPGA Interface)

Right-click an [NI 9265](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Calibration Mode**—Sets the calibration mode for the C Series module. Select **Calibrated** if you want the [FPGA I/O Node](#) to accept [fixed-point](#) data in units of amps when writing to the module. The fixed-point data is unsigned, with a word length of 20 bits and an integer word length of –5 bits. Select **Raw** if you want the FPGA I/O Node to accept calibrated, binary data when writing to the module. If you select **Raw**, you must [convert](#) the analog output values in the host VI before you write them to the module. The default is **Calibrated**.

Converting Current Values to Binary Values for the NI 9265 (FPGA Interface)

Set the **Calibration Mode** to **Calibrated** in the [C Series Module Properties](#) dialog box for the [NI 9265](#) if you want the [FPGA I/O Node](#) to accept [fixed-point](#) data in units of amps when writing to the module. If you set the **Calibration Mode** to **Raw**, the FPGA I/O Node accepts only binary values when writing to the module. You must convert output current values to binary values before you write them to the module. You must convert these values in the host VI.

Using a VI to Convert Current to Binary

Refer to the Nominal to Binary VI in the labview\examples\CompactRIO\Basic IO\Analog Raw Host Calibration\AO Raw Host Calibration\AO Raw Host Calibration - cRIO.lvproj for an example of converting analog output values to binary values. You can use the Nominal to Binary VI as a subVI in the host VI.

■ Open example

Using an Equation to Convert Current to Binary

You can use the following equation in the host VI to convert the analog output values to binary values:

$$\text{Binary Value} = (\text{Current Value} \times 10^9 - \text{Offset}) \div \text{LSB Weight}$$

where *Binary Value* is the value you write to the FPGA I/O Node

Current Value is the current in mA that you want the channel to output

Offset is the value returned by the Offset property

LSB Weight is the value returned by the LSB Weight property.

National Instruments recommends using calibrated values for analog output. To convert calibrated current values, use the [FPGA I/O Property Node](#) to read the LSB Weight and Offset properties. If you do not want to read the LSB Weight and Offset values from the module, you can convert uncalibrated current values by using the following values for *Offset* and *LSB Weight*:

$$\text{Offset} = 0$$

$$\text{LSB Weight} = 20.6 \text{ mA} \div 2^{\text{DAC Resolution}} \times 10^9$$

where *DAC Resolution* is the DAC resolution value in the [NI 9265 Operating Instructions and Specifications](#).

NI 9401 (FPGA Interface)

CompactRIO 8-Channel, TTL Digital Input/Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device. When you write to an NI 9401 channel, the FPGA I/O Node does not automatically enable the channel for output.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIOx	Digital input/output channel x, where x is the number of the channel. The NI 9401 has DIO channels 0 to 7.
DIO3:0	Digital port consisting of channels 0 through 3. Channel 3 is returned in the MSB, and channel 0 is returned in the LSB.
DIO7:4	Digital port consisting of channels 4 through 7. Channel 7 is returned in the MSB, and channel 4 is returned in the LSB.
DIO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

You can configure the arbitration settings for digital output channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). Digital input channels of this device support only the **Never Arbitrate** option for arbitration. You cannot configure arbitration settings for the digital input channels of this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Rising Edge	Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method

waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Check Status	Returns a Boolean value that indicates whether the module is ready.  Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.
Set Line Direction	Sets the direction of one port to input or output.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Configuration](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital I/O before a loop containing digital I/O starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready. The module also might ignore or delay digital output operations if it is not ready.

While the module is performing digital I/O within a single-cycle Timed Loop, do not perform property reads, use the Set Line Direction method, or remove the module from the chassis. Doing any of these actions causes the module to be unable to perform digital I/O and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 Mhz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9401 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9401 (FPGA Interface)

Right-click an [NI 9401](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Initial Line Direction**—Sets the [initial line direction](#) for each digital port to digital input or digital output. The default is digital input.
- **Advanced**—Launches the [Advanced Configuration](#) dialog box.

Configuring the Initial Line Direction for the NI 9401 (FPGA Interface)

Each digital port on the [NI 9401](#) is initially configured as a digital input. You can configure the initial line direction for each port on the NI 9401 at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the line direction for each port at run time using the [FPGA I/O Method Node](#). The execution of an I/O Method Node that is configured with a Set Line Direction method overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring Line Direction Using the C Series Module Properties Dialog Box

Complete the following steps to configure the line direction of each digital port using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9401.
2. Right-click the NI 9401 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Select the direction for each port from the **Initial Line Direction** pull-down menus.
4. Click the **OK** button.
5. Select **File»Save All** in the **Project Explorer** window.

Configuring Line Direction Using the FPGA I/O Method Node

Complete the following steps to configure the line direction of each digital port using the FPGA I/O Method Node.

1. Place an FPGA I/O Method Node on the block diagram and **configure** it for the NI 9401.
2. Click the **Method** section and select the **Set Line Direction** method from the shortcut menu.
3. Right-click each digital port input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select the direction for each port from the digital port pull-down menus.

NI 9402 (FPGA Interface)

CompactRIO 4-Channel, LVTTTL Digital Input/Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIOx	Digital input/output channel x, where x is the number of the channel. The NI 9402 has DIO channels 0 to 3.
DIO3:0	Digital port consisting of channels 0 through 3. Channel 3 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

You can configure the arbitration settings for digital output channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). Digital input channels of this device support only the **Never Arbitrate** option for arbitration. You cannot configure arbitration settings for the digital input channels of this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node (FPGA Module) topic for a description of this method.
Set Output Enable	Sets the line direction of the digital channel or the DIO3:0 digital port. Refer to the FPGA I/O Method Node (FPGA Module) topic for more information on this method.
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low

	<p>level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.</p>
Wait on Rising Edge	<p>Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.</p>

Module Method

Use the [FPGA I/O Method Node](#) to access the following method for this device.

Method	Description
Check Status	Returns a Boolean value that indicates whether the module is ready.  Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box.

The NI 9402 supports the following output synchronizing register options:

- **Number of Synchronizing Registers for Output Data when used in SCTL**
- **Number of Synchronizing Registers for Output Enable when used in SCTL**

These two options support the same functionality as the **Number of Synchronizing Registers for Output Data** and **Number of Synchronizing Registers for Output Enable** options described in the [Advanced Code Generation FPGA I/O Properties Page \(FPGA Module\)](#) topic, with the exception that you can use these options only in a single-cycle Timed Loop. You can implement either 0 or 1 synchronizing registers inside the single-cycle Timed Loop, however, if you configure 0 synchronizing registers outside of the single-cycle Timed Loop, the FPGA VI implements 1 synchronizing register by default.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital I/O before a loop containing digital I/O starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready. The module also might ignore or delay digital output operations if it is not ready.

While the module is performing digital I/O within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing any of these actions causes the module to be unable to perform digital I/O and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the *NI 9402 Operating Instructions and Specifications* to learn about module specifications and how to use the module. The operating instructions ship with the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9402 (FPGA Interface)

Right-click an [NI 9402](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Channels**—Specifies the channel(s) for which you want to select the line direction.
- **Selected Channel(s) Settings**—Specifies the line direction for each channel.
 - **Direction**—[Sets the line direction](#) for the selected channel(s) to digital input or digital output. The default is digital input.

Configuring the Line Direction for the NI 9402 (FPGA Interface)

Each digital channel on the [NI 9402](#) is initially configured as a digital input. You can configure the initial line direction for each channel on the NI 9402 at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the line direction for each channel at run time using the [FPGA I/O Method Node](#). The execution of an I/O Method Node that is configured with a Set Output Enable method overwrites the values you configured in the **C Series Module Properties** dialog box. In addition, the execution of an FPGA I/O Node configured for output automatically configures the line for output and overwrites the values you configured in the **C Series Module Properties** dialog box or using the **Set Output Enable** method.

Configuring Line Direction Using the C Series Module Properties Dialog Box

Complete the following steps to set the line direction of channels using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9402.
2. Right-click the NI 9402 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Select the channel(s) for which you want to configure the line direction from the **Channels** table. You can select more than one channel by holding the <Ctrl> or <Shift> key when selecting channels.
4. Select the direction for the channel(s) from the **Direction** pull-down menu.
5. Click the **OK** button.
6. Select **File»Save All** in the **Project Explorer** window.

Configuring Line Direction of One Channel Using the FPGA I/O Method Node

Complete the following steps to set the line direction of a channel using the FPGA I/O Method Node.

1. [Create FPGA I/O items](#) for the channel of the NI 9402 for which you want to configure the line direction.
2. Place an FPGA I/O Method Node on the block diagram and [configure](#) it for this channel.
3. Click the **Method** section and select the **Set Output Enable** method from the shortcut menu.
4. Right-click the **Enable** input and select **Create»Control** from the shortcut menu.
5. On the front panel of the VI, click the **Enable** Boolean control to set it to TRUE if you want to set the line direction of the channel to digital output.

Configuring Line Direction of Multiple Channels Using the FPGA I/O Method Node

Complete the following steps to set the line direction of multiple channels using the FPGA I/O Method Node.

1. [Create FPGA I/O items](#) for the DIO3:0 digital port of the NI 9402.
2. Place an FPGA I/O Method Node on the block diagram and [configure](#) it for the DIO3:0 digital port.
3. Click the **Method** section and select the **Set Output Enable** method from the shortcut menu.
4. Right-click the **Enable** input and select **Create»Control** from the shortcut menu. The **Enable** control appears as an unsigned 8-bit integer. Each bit in the integer represents the line direction of one channel of the NI 9402.
5. On the front panel of the VI, use the **Enable** control to enter the line direction for each channel in the port. Change a bit to 1 to set the line direction of the corresponding channel to digital output. Leave a bit as 0 to set the line direction of the corresponding channel to digital input. Refer to the table below for examples of what to enter in the **Enable** control.

Channel Line Direction Configuration	Enable Control (Floating Point)	Enable Control (Binary)	Enable Control (Hex)
Change all channels to input	0	0000	0x00
Change all channels to output	15	1111	0x0F
Change channel 0 to output	1	0001	0x01

NI 9403 (FPGA Interface)

CompactRIO 32-Channel, TTL Digital Input/Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIOx	Digital input/output channel x , where x is the number of the channel. The NI 9403 has DIO channels 0 to 31.
DIO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.
DIO15:8	Digital port consisting of channels 8 through 15. Channel 15 is returned in the MSB, and channel 8 is returned in the LSB.
DIO23:16	Digital port consisting of channels 16 through 23. Channel 23 is returned in the MSB, and channel 16 is returned in the LSB.
DIO31:24	Digital port consisting of channels 24 through 31. Channel 31 is returned in the MSB, and channel 24 is returned in the LSB.
DIO31:0	Digital port consisting of channels 0 through 31. Channel 31 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) and [Never Arbitrate](#) options for arbitration. The default arbitration setting is **Arbitrate if Multiple Requestors Only**.

If you are sure that the design of the FPGA VI will never allow more than one digital function to execute at the same time, even on different channels, you can place a checkmark in the **Disable Arbitration** checkbox on the [C Series Module Properties](#) dialog box to disable arbitration and reduce the amount of FPGA logic used by VIs. If more than one digital function could execute simultaneously in the FPGA VI, leave the box unchecked.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any module methods.

Method	Description
Set Output Enable	Sets the line direction of the digital channel or port. If the I/O item is a digital line, Enable requires a Boolean data type. If the I/O item is a digital port, Enable requires a numeric data type.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9403 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9403 (FPGA Interface)

Right-click an [NI 9403](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Channels**—Specifies the channel(s) for which you want to select the line direction.
- **Selected Channel(s) Settings**—Specifies the line direction for each channel.
 - **Direction**—[Sets the line direction](#) for the selected channel(s) to digital input or digital output. The default is digital input.
- **Disable Arbitration**—Place a checkmark in this checkbox if you want to change the arbitration setting to [Never Arbitrate](#) and reduce the amount of FPGA logic used by VIs. Check this box only if you are sure that the design of the FPGA VI will never allow more than one digital function to execute at the same time, even on different channels. If more than one digital function could execute simultaneously in the FPGA VI, leave the box unchecked to keep the default [Arbitrate if Multiple Requestors Only](#) arbitration setting.

Configuring the Line Direction for the NI 9403 (FPGA Interface)

Each digital channel on the [NI 9403](#) is initially configured as a digital input. You can configure the initial line direction for each channel on the NI 9403 at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change the line direction for each channel at run time using the [FPGA I/O Method Node](#). The execution of an I/O Method Node that is configured with a Set Output Enable method overwrites the values you configured in the **C Series Module Properties** dialog box. In addition, the execution of an FPGA I/O Node configured for output automatically configures the line for output and overwrites the values you configured in the **C Series Module Properties** dialog box or using the **Set Output Enable** method.

Configuring Line Direction Using the C Series Module Properties Dialog Box

Complete the following steps to set the line direction of channels using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9403.
2. Right-click the NI 9403 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Select the channel(s) for which you want to configure the line direction from the **Channels** table. You can select more than one channel by holding the <Ctrl> or <Shift> key when selecting channels.
4. Select the direction for the channel(s) from the **Direction** pull-down menu.
5. Click the **OK** button.
6. Select **File»Save All** in the **Project Explorer** window.

Configuring Line Direction of One Channel Using the FPGA I/O Method Node

Complete the following steps to set the line direction of a channel using the FPGA I/O Method Node.

1. [Create FPGA I/O items](#) for the channel of the NI 9403 for which you want to configure the line direction.
2. Place an FPGA I/O Method Node on the block diagram and [configure](#) it for this channel.
3. Click the **Method** section and select the **Set Output Enable** method from the shortcut menu.
4. Right-click the **Enable** input and select **Create»Control** from the shortcut menu.
5. On the front panel of the VI, click the **Enable** Boolean control to set it to TRUE if you want to set the line direction of the channel to digital output.

Configuring Line Direction of Multiple Channels Using the FPGA I/O Method Node

Complete the following steps to set the line direction of multiple channels using the FPGA I/O Method Node.

1. [Create FPGA I/O items](#) for the digital port of the NI 9403 that contains the channels you want to configure.
2. Place an FPGA I/O Method Node on the block diagram and [configure](#) it for this digital port.
3. Click the **Method** section and select the **Set Output Enable** method from the shortcut menu.
4. Right-click the **Enable** input and select **Create»Control** from the shortcut menu. If you configured the FPGA I/O Method Node for the DIO7:0, DIO15:8, DIO23:16, or DIO31:24 digital port, the **Enable** control appears as an unsigned 8-bit integer. If you configured the FPGA I/O Method Node for the DIO31:0 digital port, the **Enable** control appears as an unsigned 32-bit integer. Each bit in the integer represents the line direction of one channel of the NI 9403.
5. On the front panel of the VI, use the **Enable** control to enter the line direction for each channel in the port. Change a bit to 1 to set the line direction of the corresponding channel to digital output. Leave a bit as 0 to set the line direction of the corresponding channel to digital input. Refer to the table below for examples of what to enter in the **Enable** control if you configured the FPGA I/O Method Node for the DIO7:0 digital port.

Channel Line Direction Configuration	Enable Control (Hex)	Enable Control (Binary)
Change all channels to input	0x00	0b00000000
Change all channels to output	0xFF	0b11111111
Change channel 0 to output	0x01	0b00000001
Change channels 0 through 5 to output	0x3F	0b00111111
Change channels 3 and 7 to	0x88	0b10001000

output		
--------	--	--

Refer to the table below for examples of what to enter in the **Enable** control if you configured the FPGA I/O Method Node for the DIO31:0 digital port.

Channel Line Direction Configuration	Enable Control (Hex)	Enable Control (Binary)
Change all channels to input	0x00000000	0b0000000000000000 0000000000000000
Change all channels to output	0xFFFFFFFF	0b1111111111111111 1111111111111111
Change channel 0 to output	0x00000001	0b0000000000000000 0000000000000001
Change channels 0 through 5 to output	0x0000003F	0b0000000000000000 0000000000111111
Change channels 3 and 7 to output	0x00000088	0b0000000000000000 0000000010001000

NI 9411 (FPGA Interface)

CompactRIO 6-Channel, Differential or TTL Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIx	Digital input channel x, where x is the number of the channel. The NI 9411 has DI channels 0 to 5.
DI5:0	Digital port consisting of channels 0 through 5. Channel 5 is returned in bit 5, and channel 0 is returned in bit 0. Bits 6 and 7 return a zero.

Arbitration

This device supports only the [Never Arbitrate](#) option for arbitration. You cannot configure arbitration settings for this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Rising Edge	Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method

waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital input before a loop containing digital input starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready.

While the module is performing digital input within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital input and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9411 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9411, NI 9421, NI 9422, NI 9423, NI 9425, NI 9426, NI 9435, NI 9505, and NI 9802 (FPGA Interface)

Right-click an [NI 9411](#), [NI 9421](#), [NI 9422](#), [NI 9423](#), [NI 9425](#), [NI 9426](#), [NI 9435](#), [NI 9505](#), or [NI 9802](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.

NI 9421 (FPGA Interface)

CompactRIO 8-Channel, 24 V, Sinking Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DI x	Digital input channel x , where x is the number of the channel. The NI 9421 has DI channels 0 to 7.
DI7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Never Arbitrate](#) option for arbitration. You cannot configure arbitration settings for this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Rising Edge	Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method

waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital input before a loop containing digital input starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready.

While the module is performing digital input within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital input and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9421/9423 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9422 (FPGA Interface)

CompactRIO 8-Channel, 24 V, Sinking/Sourcing Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIx	Digital input channel x, where x is the number of the channel. The NI 9422 has DI channels 0 to 7.
DI7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Never Arbitrate](#) option for arbitration. You cannot configure arbitration settings for this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Rising Edge	Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method

waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital input before a loop containing digital input starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready.

While the module is performing digital input within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital input and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9422 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9423 (FPGA Interface)

CompactRIO 8-Channel, 24 V, High-Speed Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DI x	Digital input channel x , where x is the number of the channel. The NI 9423 has DI channels 0 to 7.
DI7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Never Arbitrate](#) option for arbitration. You cannot configure arbitration settings for this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Rising Edge	Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method

waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device.

Method	Description
Check Status	Returns a Boolean value that indicates whether the module is ready.  Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital input before a loop containing digital input starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready.

While the module is performing digital input within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital input and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9421/9423 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9425 (FPGA Interface)

CompactRIO 32-Channel, 24 V, Sinking Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIx	Digital input channel x, where x is the number of the channel. The NI 9425 has DI channels 0 to 31.
DI7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.
DI15:8	Digital port consisting of channels 8 through 15. Channel 15 is returned in the MSB, and channel 8 is returned in the LSB.
DI23:16	Digital port consisting of channels 16 through 23. Channel 23 is returned in the MSB, and channel 16 is returned in the LSB.
DI31:24	Digital port consisting of channels 24 through 31. Channel 31 is returned in the MSB, and channel 24 is returned in the LSB.
DI31:0	Digital port consisting of channels 0 through 31. Channel 31 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9425 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9426 (FPGA Interface)

CompactRIO 32-Channel, 24 V, Sourcing Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIx	Digital input channel x, where x is the number of the channel. The NI 9426 has DI channels 0 to 31.
DI7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.
DI15:8	Digital port consisting of channels 8 through 15. Channel 15 is returned in the MSB, and channel 8 is returned in the LSB.
DI23:16	Digital port consisting of channels 16 through 23. Channel 23 is returned in the MSB, and channel 16 is returned in the LSB.
DI31:24	Digital port consisting of channels 24 through 31. Channel 31 is returned in the MSB, and channel 24 is returned in the LSB.
DI31:0	Digital port consisting of channels 0 through 31. Channel 31 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Methods

This device does not support any methods.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9426 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9435 (FPGA Interface)

CompactRIO 4-Channel, AC/DC Universal Digital Input Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DIx	Digital input channel x, where x is the number of the channel. The NI 9435 has DI channels 0 to 3.
DI3:0	Digital port consisting of channels 0 through 3. Channel 3 is returned in bit 3, and channel 0 is returned in bit 0. Bits 4 through 7 return a zero.

Arbitration

This device supports only the [Never Arbitrate](#) option for arbitration. You cannot configure arbitration settings for this device.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device.

Method	Description
Wait on Any Edge	Pauses the execution of the I/O Method Node until the next falling or rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Any Edge method waits for the next falling or rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Falling Edge	Pauses the execution of the I/O Method Node until the next falling edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Falling Edge method waits for the next falling edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on High Level	Pauses the execution of the I/O Method Node until the digital signal is high. The Timeout input specifies in FPGA clock ticks how long the Wait on High Level method waits for the next high level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Low Level	Pauses the execution of the I/O Method Node until the digital signal is low. The Timeout input specifies in FPGA clock ticks how long the Wait on Low Level method waits for the next low level. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.
Wait on Rising Edge	Pauses the execution of the I/O Method Node until the next rising edge of the digital signal. The Timeout input specifies in FPGA clock ticks how long the Wait on Rising Edge method

waits for the next rising edge. A value of 0 causes the method to timeout immediately, a negative value causes the method to wait indefinitely, and a positive value causes the method to wait for that number of clock ticks before timing out.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital input before a loop containing digital input starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. Digital input operations return invalid data if the module is not ready.

While the module is performing digital input within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital input and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 Mhz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9435 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Advanced Configuration Dialog Box (FPGA Interface)

Click the **Advanced** button on the [C Series Module Properties](#) dialog box for the [NI 9401](#), [NI 9472](#), [NI 9474](#), [NI 9475](#), [NI 9481](#), or [NI 9485](#) to display this dialog box.

Use this dialog box to configure the number of output synchronizing registers for each DO channel in a single-cycle Timed Loop.

This dialog box includes the following components:

- **Channels**—Select the channel for which you want to configure the number of output synchronizing registers.
- **Channel Configuration**—Specifies the number of synchronizing [registers](#) between the DO channel executing on the FPGA target and the FPGA target hardware interface. The FPGA target hardware interface might be a physical I/O connector on the device or a connection to a section of the FPGA that contains circuitry designed by National Instruments. Each synchronizing register executes in one clock cycle.



Caution Select **0** only if you also use the [HDL Interface Node](#) and the HDL code contains its own synchronization registers.

- **0**—Specifies that the FPGA VI uses no synchronizing registers. Do not select this option for most FPGA Module applications.



Note If you select **0** for digital output resources in a single-cycle [Timed Loop](#), you create a combinatorial circuit between the two resources. The combinatorial circuit might cause glitches on the output signal.

- **1**—Specifies that the FPGA VI uses one synchronizing register between the DO channel and the FPGA target hardware interface.

NI 9472 (FPGA Interface)

CompactRIO 8-Channel, 24 V, Sourcing Digital Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DOx	Digital output channel x , where x is the number of the channel. The NI 9472 has DO channels 0 to 7.
DO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

You can configure the arbitration settings for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Configuration](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital output before a loop containing digital output starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. The module might ignore or delay digital output operations if it is not ready.

While the module is performing digital output within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital output and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 Mhz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9472/9474 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9472, NI 9474, NI 9475, NI 9481, and NI 9485 (FPGA Interface)

Right-click an [NI 9472](#), [NI 9474](#), [NI 9475](#), [NI 9481](#), or [NI 9485](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Advanced**—Launches the [Advanced Configuration](#) dialog box.

NI 9474 (FPGA Interface)

CompactRIO 8-Channel, 24 V, High-Speed, Sourcing Digital Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DOx	Digital output channel x , where x is the number of the channel. The NI 9474 has DO channels 0 to 7.
DO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

You can configure the arbitration settings for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Configuration](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital output before a loop containing digital output starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. The module might ignore or delay digital output operations if it is not ready.

While the module is performing digital output within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital output and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9472/9474 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9475 (FPGA Interface)

CompactRIO 8-Channel, 60 V, High-Speed, Sourcing Digital Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DOx	Digital output channel x , where x is the number of the channel. The NI 9475 has DO channels 0 to 7.
DO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

You can configure the arbitration settings for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Configuration](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital output before a loop containing digital output starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. The module might ignore or delay digital output operations if it is not ready.

While the module is performing digital output within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital output and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 Mhz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9475 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9476 (FPGA Interface)

CompactRIO 32-Channel, 24 V, High-Speed, Sourcing Digital Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DOx	Digital output channel x , where x is the number of the channel. The NI 9476 has DO channels 0 to 31.
DO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.
DO15:8	Digital port consisting of channels 8 through 15. Channel 15 is returned in the MSB, and channel 8 is returned in the LSB.
DO23:16	Digital port consisting of channels 16 through 23. Channel 23 is returned in the MSB, and channel 16 is returned in the LSB.
DO31:24	Digital port consisting of channels 24 through 31. Channel 31 is returned in the MSB, and channel 24 is returned in the LSB.
DO31:0	Digital port consisting of channels 0 through 31. Channel 31 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) and [Never Arbitrate](#) options for arbitration. The default arbitration setting is **Arbitrate if Multiple Requestors Only**.

If you are sure that the design of the FPGA VI will never allow more than one digital function to execute at the same time, even on different channels, you can place a checkmark in the **Disable Arbitration** checkbox on the [C Series Module Properties](#) dialog box to disable arbitration and reduce the amount of FPGA logic used by VIs. If more than one digital function could execute simultaneously in the FPGA VI, leave the box unchecked.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Input/Output	Description
Check Output Status	Force Status Read	When the value of this input is FALSE, the method returns the cached status information from the last status read. When the value is TRUE, the method gets the current status information from the module. Forcing a status read can introduce jitter into a digital output loop.
	Any Overcurrent	Returns a Boolean value. A value of TRUE indicates an overcurrent condition on at least one channel.
	Channel Overcurrent	Returns an array of Boolean values. A value of TRUE in any index indicates that the channel sharing a number with that index is in an overcurrent condition.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9476 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9476 and NI 9477 (FPGA Interface)

Right-click an [NI 9476](#) or [NI 9477](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Disable Arbitration**—Place a checkmark in this checkbox if you want to change the arbitration setting to [Never Arbitrate](#) and reduce the amount of FPGA logic used by VIs. Check this box only if you are sure that the design of the FPGA VI will never allow more than one digital output function to execute at the same time, even on different channels. If more than one digital output function could execute simultaneously in the FPGA VI, leave the box unchecked to keep the default [Arbitrate if Multiple Requestors Only](#) arbitration setting.

NI 9477 (FPGA Interface)

CompactRIO 32-Channel, 5–60 V, Sinking Digital Output Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DOx	Digital output channel x , where x is the number of the channel. The NI 9477 has DO channels 0 to 31.
DO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.
DO15:8	Digital port consisting of channels 8 through 15. Channel 15 is returned in the MSB, and channel 8 is returned in the LSB.
DO23:16	Digital port consisting of channels 16 through 23. Channel 23 is returned in the MSB, and channel 16 is returned in the LSB.
DO31:24	Digital port consisting of channels 24 through 31. Channel 31 is returned in the MSB, and channel 24 is returned in the LSB.
DO31:0	Digital port consisting of channels 0 through 31. Channel 31 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) and [Never Arbitrate](#) options for arbitration. The default arbitration setting is **Arbitrate if Multiple Requestors Only**.

If you are sure that the design of the FPGA VI will never allow more than one digital function to execute at the same time, even on different channels, you can place a checkmark in the **Disable Arbitration** checkbox on the [C Series Module Properties](#) dialog box to disable arbitration and reduce the amount of FPGA logic used by VIs. If more than one digital function could execute simultaneously in the FPGA VI, leave the box unchecked.

Methods

This device does not support any methods.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9477 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9478 (FPGA Interface)

CompactRIO 16-Channel, 0–50 V Sinking Digital Output Module with Programmable Current Limits

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
DOx	Digital output channel x , where x is the number of the channel. The NI 9478 has DO channels 0 to 15.
DO7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.
DO15:8	Digital port consisting of channels 8 through 15. Channel 15 is returned in the MSB, and channel 8 is returned in the LSB.
DO15:0	Digital port consisting of channels 0 through 15. Channel 15 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Input/Output	Description
Check Output Status	Limit Selection	Specifies which current limit status information the module reads. Select Active Limit if you want the Channel Overcurrent output to return TRUE for any channels that exceed the current limit threshold specified for the active current limit for that channel. Select Limit A to return TRUE for any channels that exceed the current limit threshold specified for Current Limit A. Select Limit B to return TRUE for any channels that exceed the current limit threshold specified for Current Limit B.
	Module Overtemp	Returns a Boolean value. A value of TRUE indicates that the module exceeded its temperature rating.
	Channel Overcurrent	Returns an array of Boolean values. A value of TRUE in any index indicates that the channel sharing a number with that index exceeded the current limit, as specified by the Limit Selection input, after the last status read. A value of FALSE indicates that the channel did not exceed the current limit after the last status read.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Active Current Limit	Sets the active current limit for each channel. The value in an index of the input array controls the active current limit for the channel sharing a number with that index. Select No Change if you do not want to modify the active current limit for the channel at run time. Select Limit A if you want to set the active current limit for the channel to Current Limit A. Select Limit B if you want to set the active current limit for the channel to Current Limit B. Select No Limit if you want the channel to have no current limit.
Current Limit A	Sets the current limit threshold for Current Limit A in binary values. You can use the following equation to calculate the binary value from amperes: $\text{Binary Value} = (\text{Current Value} / 5.12 \text{ A}) \times 256$ The range of valid current limit values is 0–5.1 A, which corresponds to binary values 0–255.
Current Limit B	Sets the current limit threshold for Current Limit B in binary values. You can use the following equation to calculate the binary value from amperes: $\text{Binary Value} = (\text{Current Value} / 5.12 \text{ A}) \times 256$ The range of valid current limit values is 0–5.1 A, which corresponds to binary values 0–255.
Module ID	Returns the module ID .
Overcurrent Refresh Period	Sets the time in μs it takes a channel to automatically recover if the channel exceeds its active current limit threshold. Values 2–255 enable overcurrent refresh and specify the time in tens of μs . For example, a value of 3 sets an overcurrent refresh period of 30 μs . A value of 0 disables overcurrent refresh so the channel remains disabled after exceeding its active current limit until you write to the channel using an FPGA I/O Node. A value of 1 is not supported and sets an overcurrent refresh period of 20 μs .

Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

Hardware Documentation

Refer to the [NI 9478 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

C Series Module Properties Dialog Box for the NI 9478 (FPGA Interface)

Right-click an [NI 9478](#) C Series module in the **Project Explorer** window and select **Properties** from the shortcut menu to display this dialog box.

Use this dialog box to configure a C Series module.

This dialog box includes the following components:

- **Name**—Specifies the name of the C Series module, which appears in the **Project Explorer** window. LabVIEW assigns a default name to the module based on the slot number. You can use this field to give the module a descriptive name.
- **Module Type**—Specifies the type of C Series module. You cannot change this option.
- **Location**—Specifies a slot in the chassis for the C Series module.
- **Channels**—Specifies the channel(s) for which you want to select the active current limit.
- **Selected Channel(s) Settings**—Specifies the active current limit for each channel.
 - **Active Limit**—[Sets the active current limit](#) for the selected channel(s). Select **Limit A** if you want to set the active current limit for the channel(s) to Current Limit A. Select **Limit B** if you want to set the active current limit for the channel(s) to Current Limit B. Select **No Limit** if you want the channel(s) to have no current limit.
- **Module Settings**—Specifies the current limit threshold settings for the module.
 - **Current Limit A**—Sets the current limit threshold for Current Limit A in amperes. Valid values are 0–5.1 A.
 - **Current Limit B**—Sets the current limit threshold for Current Limit B in amperes. Valid values are 0–5.1 A.
 - **Enable Overcurrent Refresh**—Place a checkmark in this checkbox if you want to enable a channel to automatically recover if the channel exceeds its active current limit threshold. If overcurrent refresh is disabled, the channel remains off after an overcurrent condition

until you write to the channel using an FPGA I/O Node.

- **Overcurrent Refresh Period**—Sets the time in μs it takes a channel to automatically recover if the channel exceeds its active current limit threshold. Valid values are 20–2550 μs . This option is available only if you select **Enable Overcurrent Refresh**.

Configuring Current Limits for the NI 9478 (FPGA Interface)

You can configure the active current limit for each channel and the current limit thresholds at edit time using the [C Series Module Properties](#) dialog box. You can programmatically change these settings at run time using the [FPGA I/O Property Node](#). The execution of an I/O Property Node that is configured with a **Current Limit** property overwrites the value you configured in the **C Series Module Properties** dialog box.

Configuring Current Limits Using the C Series Module Properties Dialog Box

Complete the following steps to configure the active current limit for each channel and the current limit thresholds for the NI 9478 using the **C Series Module Properties** dialog box.

1. **Configure** the CompactRIO system, and add an NI 9478.
2. Right-click the NI 9478 in the **Project Explorer** window and select **Properties** from the shortcut menu to display the **C Series Module Properties** dialog box.
3. Select the channel(s) for which you want to configure the active current limit from the **Channels** table. You can select more than one channel by holding the <Ctrl> or <Shift> key when selecting channels.
4. Select the active current limit for the channel(s) from the **Active Limit** pull-down menu.
5. Enter a value between 0 and 5.1 in the **Current Limit** text box that corresponds to the active current limit for the selected channel(s).
6. Place a checkmark in the **Enable Overcurrent Refresh** checkbox and set the time in the **Overcurrent Refresh Period** text box if you want to enable a channel to automatically recover if the channel exceeds its active current limit threshold.
7. Click the **OK** button.
8. Select **File»Save All** in the **Project Explorer** window.

Configuring Current Limits Using the FPGA I/O Property Node

Complete the following steps to configure the active current limit for each channel and the current limit thresholds using the FPGA I/O Property Node.

1. Place three FPGA I/O Property Nodes on the block diagram and **configure** them for the NI 9478.
2. Click the **Property** section on the first Property Node and select the **Active Current Limit** property from the shortcut menu.
3. Right-click the **Active Current Limit** input and select **Create»Control** from the shortcut menu.
4. On the front panel of the VI, select the active current limit for each channel from the **Active Current Limit** array.
5. Click the **Property** section on the second Property Node and select the **Current Limit A** property from the shortcut menu.
6. Right-click the **Current Limit A** input and select **Create»Control** from the shortcut menu.
7. Click the **Property** section on the third Property Node and select the **Current Limit B** property from the shortcut menu.
8. Right-click the **Current Limit B** input and select **Create»Control** from the shortcut menu.
9. On the front panel of the VI, enter binary values in the **Current Limit** controls to set the current limit thresholds. You can use the following equation in the host VI to calculate the binary value from amperes:

$$\text{Binary Value} = (\text{Current Value} / 5.12 \text{ A}) \times 256$$

NI 9481 (FPGA Interface)

CompactRIO 4-Channel, Form A Electromechanical Relay Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
CHx	Channel x, where x is the number of the channel. The NI 9481 has channels 0 to 3.
CH3:0	Digital port consisting of channels 0 through 3. Channel 3 is returned in bit 3, and channel 0 is returned in bit 0.

Arbitration

You can configure the arbitration settings for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Configuration](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital output before a loop containing digital output starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. The module might ignore or delay digital output operations if it is not ready.

While the module is performing digital output within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital output and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 MHz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9481 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

NI 9485 (FPGA Interface)

CompactRIO 8-Channel, Solid-State Relay (SSR) Module

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
CHx	Channel x, where x is the number of the channel. The NI 9485 has channels 0 to 7.
CH7:0	Digital port consisting of channels 0 through 7. Channel 7 is returned in the MSB, and channel 0 is returned in the LSB.

Arbitration

You can configure the arbitration settings for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

Module Methods

Use the [FPGA I/O Method Node](#) to access the following method for this device. This device does not support any I/O methods.

Method	Description
Check Status	<p>Returns a Boolean value that indicates whether the module is ready.</p> <p> Note During the first 2 seconds after you reset the FPGA VI, the error terminals on this method may not correctly report certain types of errors.</p>

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device. This device does not support any I/O properties.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Configuration](#) dialog box.

When the module is within a single-cycle Timed Loop, it must be ready to perform digital output before a loop containing digital output starts. Poll the **Ready** output of the Check Status method to determine whether the module is ready. The module might ignore or delay digital output operations if it is not ready.

While the module is performing digital output within a single-cycle Timed Loop, do not perform property reads or remove the module from the chassis. Doing either of these actions causes the module to be unable to perform digital output and the **Ready** output of the Check Status method to return FALSE.

FPGA Target Clock Support

This device supports only [top-level FPGA target clock rates](#) and [single-cycle Timed Loop clock rates](#) that are multiples of 40 MHz, such as 40 MHz, 80 Mhz, 120 MHz, and so on.

Hardware Documentation

Refer to the [NI 9485 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Memory Modules (FPGA Interface)

Use this book as a reference for information about which methods and properties each C Series memory module supports.



To view related topics, click the **Locate** button, shown at left, in the toolbar at the top of this window. The *LabVIEW Help* highlights this topic in the **Contents** tab so you can navigate the related topics.

NI 9802 (FPGA Interface)

CompactRIO 2 SD Card Memory Module

FPGA I/O Node

This device does not support the [FPGA I/O Node](#). You can use the [FPGA I/O Method Node](#) and [FPGA I/O Property Node](#) to access methods and properties for the device.



Note If you add the NI 9802 to the project, the Real-Time support for the Mount SD Card Interface method compiles onto the FPGA when you compile the FPGA VI, which can take up space on the FPGA. If you are not actively using the NI 9802, you might want to remove the module item from the project.

Arbitration

This device supports only the [Arbitrate if Multiple Requestors Only](#) option for arbitration. You cannot configure arbitration settings for this device.

SD Card Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. You also can use the [Mount SD Card](#) Interface method to mount the file system of an SD Card into a drive. This device does not support any module methods.

Method	Description
Close File	Closes the open file specified by FileRefIn . This method executes regardless of whether an error occurred in a preceding operation, which ensures that files are closed correctly. If the file reference is zero, the method returns an error and does not execute.
Delete File	Deletes the file specified by FileID . The file ID can be any unsigned integer value from 0 to 255 and appears as an 8-bit hexadecimal value in the filename. If the specified file is open, the method returns an error and does not execute.
Get File Size	<p>Returns the file size in bytes of the file specified by FileID. The file ID can be any unsigned integer value from 0 to 255 and appears as an 8-bit hexadecimal value in the filename. If the specified file is open, the method returns an error and does not execute.</p> <p>Refer to the NI 9802 Methods VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9802\NI 9802 Methods</code> directory for an example of using this method.</p> <p>■ Open example</p>
Get Space	<p>Returns the amount of unallocated storage in bytes on the SD Card file system. If a file is open on the specified card, the value returned may be less than the actual free space.</p> <p>Refer to the NI 9802 Methods VI in the <code>labview\examples\CompactRIO\Module Specific\NI 9802\NI 9802 Methods</code> directory for an example of using this method.</p>

	<p>■ Open example</p>
Open File	<p>Opens or creates a file specified by FileID.</p> <p> Note Only one file can be open on a given SD Card at one time.</p> <p>The file ID can be any unsigned integer value from 0 to 255 and appears as an 8-bit hexadecimal value in the filename. The name of the file is in the format CRIO-X##.BIN, where ## is the hexadecimal representation of the value you entered. The filename must be uppercase and must comply with the DOS 8.3 naming convention. The Mode input determines the access mode of the file: Read opens an existing file for reading only and returns an error if the file does not exist; Write New creates a new file for writing and returns an error if the file already exists; Write Over either creates a new file for writing or opens the existing file and overwrites existing data with new data; and Write Append either creates a new file for writing or opens the existing file and appends new data to the end of the file. FileRefOut returns a reference to the open file.</p> <p> Note You must use the Close File method to close the file you open. Not closing a file may result in incomplete data or errors the next time you use the file.</p>
Read U32	<p>Reads binary data in big-endian format from the open file specified by FileRefIn. If the file reference is zero, the method returns an error and does not execute. FileRefOut returns a reference to the open file. EOF returns TRUE when there is no more data to read. If the file does not contain the number of bytes requested, the method returns an error.</p>
Read U16	<p>Reads binary data in big-endian format from the open file specified by FileRefIn. If the file reference is zero, the method returns an error and does not execute. FileRefOut returns a reference to the open file. EOF returns TRUE when there is no more data to read. If the file does not contain the number of bytes requested, the method returns an error.</p>

Read U8	<p>Reads binary data in big-endian format from the open file specified by FileRefIn. If the file reference is zero, the method returns an error and does not execute. FileRefOut returns a reference to the open file. EOF returns TRUE when there is no more data to read. If the file does not contain the number of bytes requested, the method returns an error.</p> <p>Refer to the NI 9802 Read U8 Variable VI in the labview\examples\CompactRIO\Module Specific\NI 9802\NI 9802 Read U8 File directory for an example of using this method.</p> <p>■ Open example</p>
Write	<p>Writes binary data in big-endian format to the open file specified by FileRefIn. The data type you wire to the FileRefIn input determines whether the polymorphic method writes U32, U16, or U8 values. If the file reference is zero, the method returns an error and does not execute. FileRefOut returns a reference to the open file.</p> <p>Refer to the NI 9802 Write VI in the labview\examples\CompactRIO\Module Specific\NI 9802\NI 9802 Write File directory for an example of using this method.</p> <p>■ Open example</p>

SD Card Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Refer to the NI 9802 Methods VI in the `labview\examples\CompactRIO\Module Specific\NI 9802\NI 9802 Methods` directory for an example of using these properties.

■ Open example

Property	Description
Card Present	Returns a Boolean value. A value of TRUE indicates that an SD Card is inserted in the slot.
Door Open	Returns a Boolean value. A value of TRUE indicates that the SD Card slot door is open and it is not safe to read from or write to the SD Card in the slot.

Module Properties

Use the [FPGA I/O Property Node](#) to access the following properties for this device.

Property	Description
Module ID	Returns the module ID .
Serial Number	Returns the unique serial number of the module.
Vendor ID	Returns the National Instruments vendor ID, 0x1093.

Removing SD Cards

Do not open an SD Card slot door or remove an SD Card while the yellow LED for that card is lit or flashing.



Caution Opening an SD Card slot door or removing an SD Card while the yellow LED for that card is lit or flashing might result in incomplete data.

If you open an SD Card slot door while the yellow LED for that card is lit or flashing, the software attempts to protect the SD Card file system by closing all open files, which can result in lost data. Refer to the [NI 9802 Operating Instructions and Specifications](#) for descriptions of each of the LEDs on the NI 9802 and when it is safe to remove SD Cards from the NI 9802.

Single-Cycle Timed Loop

This device does not support the single-cycle Timed Loop.

FPGA Target Clock Support

This device supports only the 40 MHz FPGA target clock setting.

Hardware Documentation

Refer to the [NI 9802 Operating Instructions and Specifications](#) to learn about module specifications and how to use the module. Refer to the [CompactRIO Related Documentation](#) for further information about CompactRIO documentation.

Single-Board RIO Devices (FPGA Interface)

An NI Single-Board RIO (sbRIO) device consists of a real-time processor connected to a reconfigurable [FPGA](#), with 110 digital I/O (DIO) lines and three expansion slots for C Series modules. The NI sbRIO-961x, NI sbRIO-963x, and NI sbRIO-964x devices have additional I/O lines provided by onboard C Series modules.

The user-reconfigurable FPGA controls the digital and analog I/O lines on the sbRIO device. You can [configure the sbRIO device](#) and use the LabVIEW FPGA Module and NI-RIO to [create](#) and [download](#) a custom VI to the FPGA to define the timing and functionality of the sbRIO device. If you have only LabVIEW and NI-RIO but not the FPGA Module, you cannot create new FPGA VIs, but you can create VIs that run on Windows or a [LabVIEW Real-Time \(RT\)](#) target to control existing FPGA VIs.

Use this book as a reference for information about which [FPGA I/O functions](#), I/O resources, arbitration options, methods, and properties each sbRIO device supports.



To view related topics, click the **Locate** button, shown at left, in the toolbar at the top of this window. The *LabVIEW Help* highlights this topic in the **Contents** tab so you can navigate the related topics.

NI sbRIO-9601 (FPGA Interface)

Single-Board Reconfigurable I/O (DIO)

110 3.3 V DIO channels, 1 million gate FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9601 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9602 (FPGA Interface)

Single-Board Reconfigurable I/O (DIO)

110 3.3 V DIO channels, 2 million gate FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9602 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#).

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#). Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9611 (FPGA Interface)

Single-Board Reconfigurable I/O (AI, DIO)

32 AI channels, 110 3.3 V DIO channels, 1 million gate FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9611 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). For information about arbitration on channels of an onboard or installed C Series module, refer to the help topic for the module. You can find the help topic for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.

Onboard C Series Module I/O

This device contains an onboard [NI 9205](#) module.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#) for digital I/O only. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9612 (FPGA Interface)

Single-Board Reconfigurable I/O (AI, DIO)

32 AI channels, 110 3.3 V DIO channels, 2 million gate FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9612 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). For information about arbitration on channels of an onboard or installed C Series module, refer to the help topic for the module. You can find the help topic for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.

Onboard C Series Module I/O

This device contains an onboard [NI 9205](#) module.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#) for digital I/O only. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9631 (FPGA Interface)

Single-Board Reconfigurable I/O (AI, AO, DIO)

32 AI channels, 4 AO channels, 110 3.3 V DIO channels, 1 million gate
FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9631 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). For information about arbitration on channels of an onboard or installed C Series module, refer to the help topic for the module. You can find the help topic for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.

Onboard C Series Module I/O

This device contains an onboard [NI 9205](#) module and an onboard [NI 9263](#) module.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#) for digital I/O only. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9632 (FPGA Interface)

Single-Board Reconfigurable I/O (AI, AO, DIO)

32 AI channels, 4 AO channels, 110 3.3 V DIO channels, 2 million gate
FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9632 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). For information about arbitration on channels of an onboard or installed C Series module, refer to the help topic for the module. You can find the help topic for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.

Onboard C Series Module I/O

This device contains an onboard [NI 9205](#) module and an onboard [NI 9263](#) module.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#) for digital I/O only. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9641 (FPGA Interface)

Single-Board Reconfigurable I/O (AI, AO, DI, DO, DIO)

32 AI channels, 4 AO channels, 110 3.3 V DIO channels, 32 24 V DI channels, 32 24 V DO channels, 1 million gate FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9641 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). For information about arbitration on channels of an onboard or installed C Series module, refer to the help topic for the module. You can find the help topic for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.

Onboard C Series Module I/O

This device contains onboard [NI 9205](#), [NI 9263](#), [NI 9425](#), and [NI 9476](#) modules.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#) for digital I/O only. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

NI sbRIO-9642 (FPGA Interface)

Single-Board Reconfigurable I/O (AI, AO, DI, DO, DIO)

32 AI channels, 4 AO channels, 110 3.3 V DIO channels, 32 24 V DI channels, 32 24 V DO channels, 2 million gate FPGA

FPGA I/O Node

You can use an [FPGA I/O Node](#), configured for [reading](#) and [writing](#), with this device.

Terminals in Software

Use the FPGA I/O Node to access the following terminals for this device.

Terminal	Description
Portx/DIOy	Digital input/output channel <i>y</i> on port <i>x</i> , where <i>y</i> is the channel number and <i>x</i> is the port number. The NI sbRIO-9642 has ports 0 to 9. Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access this channel.
Portx/DIOCTL	Digital control channel on port <i>x</i> . Do not use DIOCTL for high-speed digital communication.
Portx/DIO9:0	Digital input/output channels 0 to 9 on port <i>x</i> . Use the FPGA I/O Node or the Set Data Output or Set Data Enable method to access all channels on the port.

Arbitration

You can configure the arbitration settings for the DIO channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. The default arbitration setting is [Never Arbitrate](#). For information about arbitration on channels of an onboard or installed C Series module, refer to the help topic for the module. You can find the help topic for the module you are using by navigating on the **Contents** tab to **FPGA Module»CompactRIO Reference and Procedures»Module Type»NI 9xxx**.

Onboard C Series Module I/O

This device contains onboard [NI 9205](#), [NI 9263](#), [NI 9425](#), and [NI 9476](#) modules.

I/O Methods

Use the [FPGA I/O Method Node](#) to access the following methods for this device. This device does not support any module methods.

Method	Description
Set Output Data	Refer to the FPGA I/O Method Node topic for a description of this method.
Set Output Enable	Refer to the FPGA I/O Method Node topic for a description of this method.

Properties

This device does not support any properties.

Single-Cycle Timed Loop

This device supports the [single-cycle Timed Loop](#) for digital I/O only. Configure the number of output synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Properties](#) dialog box. Configure the number of input synchronizing registers for the channels of this device in the [Advanced Code Generation](#) page of the [FPGA I/O Node Properties](#) dialog box.

Configuring a Project for a CompactRIO or Single-Board RIO System (FPGA Interface)

You can configure a project for a [CompactRIO Reconfigurable or Integrated system](#), [CompactRIO R Series Expansion system](#), or [Single-Board RIO Reconfigurable Embedded system](#).