NI-IMAQdx VI Reference Help

June 2008, 371969C-01

NI-IMAQdx driver software gives you the ability to acquire images with IEEE 1394 and GigE Vision cameras. This help file describes the VIs included in the NI-IMAQdx driver software.

For more information about this help file, refer to the following topics:

Using Help

Related Documentation

Glossary

Important Information

Technical Support and Professional Services

To comment on National Instruments documentation, refer to the <u>National</u> <u>Instruments Web site</u>.

© 2006–2008 National Instruments Corporation. All rights reserved.

Related Documentation

Some NI-IMAQdx manuals also are available as PDFs. You must have Adobe Reader with Search and Accessibility 5.0.5 or later installed to view the PDFs. Refer to the <u>Adobe Systems Incorporated Web site</u> at www.adobe.com to download Adobe Reader. Refer to the <u>National Instruments Product Manuals Library</u> at ni.com/manuals for updated documentation resources.

The following documents contain information that you may find helpful as you use this help file:

- Deployment Policy for NI-IMAQdx Note to Users—Contains information about the deployment policy for NI-IMAQdx driver software.
- Measurement & Automation Explorer Help for NI-IMAQdx—
 Describes how to configure NI-IMAQdx driver software, NI image acquisition devices, and cameras using Measurement & Automation Explorer.
- NI-IMAQdx Function Reference Help—Contains reference information about the LabWindows/CVI functions for NI-IMAQdx driver software.
- *NI-IMAQdx Help*—Contains fundamental programming concepts for NI-IMAQdx driver software.
- NI Vision Acquisition Software Release Notes—Contains information about new functionality, minimum system requirements, and installation instructions for NI-IMAQdx driver software.

Activating Your Software

How do I activate my software?

Use the NI Activation Wizard to obtain an activation code for your software. You can launch the NI Activation Wizard two ways:

- Launch the product and choose to activate your software from the list of options presented.
- Launch NI License Manager by selecting Start»All
 Programs»National Instruments»NI License Manager. Click the Activate button in the toolbar.



Note You do not need to activate your software if it is managed by NI Volume License Manager as a part of a Volume License Agreement.

What is activation?

Activation is the process of obtaining an activation code to enable your software to run on your computer. An *activation code* is an alphanumeric string that verifies the software, version, and computer ID to enable features on your computer. Activation codes are unique and are valid on only one computer.

What is the NI Activation Wizard?

The NI Activation Wizard is a part of NI License Manager that steps you through the process of enabling software to run on your machine.

What information do I need to activate?

You need your product serial number, user name, and organization. The NI Activation Wizard determines the rest of the information. Certain activation methods may require additional information for delivery. This information is used only to activate your product. Complete disclosure of National Instruments licensing privacy policy is available at ni.com/activate/privacy. If you optionally choose to register your software, your information is protected under the National Instruments privacy policy, available at ni.com/privacy.

How do I find my product serial number?

You can find your serial number on the proof-of-ownership and registration card that you received with your product, as shown in the

following example.



What is a Computer ID?

The computer ID contains unique information about your computer. National Instruments requires this information to enable your software. You can find your computer ID through the NI Activation Wizard or by using NI License Manager, as follows:

- Launch NI License Manager by selecting Start»Programs»National Instruments»NI License Manager.
- 2. Click the **Display Computer Information** button in the toolbar.

For more information about <u>product activation and licensing</u> refer to ni.com/activate.

Using Help

Conventions

Navigating Help

Searching Help

Printing Help File Topics

Conventions

This help file uses the following formatting and typographical conventions:

[] Square brackets enclose optional items—for example, [response].

The » symbol leads you through nested menu items and dialog box options to a final action. The sequence File»Page Setup»Options directs you to pull down the File menu, select the Page Setup item, and select Options from the last dialog box.

This icon denotes a tip, which alerts you to advisory information.

This icon denotes a note, which alerts you to important information.

bold Bold text denotes items that you must select or click in the software, such as menu items and dialog box options. Bold text also denotes parameter names.

green Underlined text in this color denotes a link to a help topic, help file, or Web address.

italic Italic text denotes variables, emphasis, cross-references, or an introduction to a key concept. Italic text also denotes text that is a placeholder for a word or value that you must supply.

monospace Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames, and extensions.

Navigating Help (Windows Only)

To navigate this help file, use the **Contents**, **Index**, and **Search** tabs to the left of this window or use the following toolbar buttons located above the tabs:

- **Hide**—Hides the navigation pane from view.
- Locate—Locates the currently displayed topic in the Contents tab, allowing you to view related topics.
- **Back**—Displays the previously viewed topic.
- **Forward**—Displays the topic you viewed before clicking the **Back** button.
- **Options**—Displays a list of commands and viewing options for the help file.

Searching Help (Windows Only)

Use the **Search** tab to the left of this window to locate content in this help file. If you want to search for words in a certain order, such as "related documentation," add quotation marks around the search words as shown in the example. Searching for terms on the **Search** tab allows you to quickly locate specific information and information in topics that are not included on the **Contents** tab.

Wildcards

You also can search using asterisk (*) or question mark (?) wildcards. Use the asterisk wildcard to return topics that contain a certain string. For example, a search for "prog*" lists topics that contain the words "program," "programmatically," "progress," and so on.

Use the question mark wildcard as a substitute for a single character in a search term. For example, "?ext" lists topics that contain the words "next," "text," and so on.



Note Wildcard searching will not work on Simplified Chinese, Traditional Chinese, Japanese, and Korean systems.

Nested Expressions

Use nested expressions to combine searches to further refine a search. You can use Boolean expressions and wildcards in a nested expression. For example, "example AND (program OR VI)" lists topics that contain "example program" or "example VI." You cannot nest expressions more than five levels.

Boolean Expressions

Click the **•** button to add Boolean expressions to a search. The following Boolean operators are available:

- AND (default)—Returns topics that contain both search terms. You
 do not need to specify this operator unless you are using nested
 expressions.
- **OR**—Returns topics that contain either the first or second term.
- **NOT**—Returns topics that contain the first term without the second term.
- **NEAR**—Returns topics that contain both terms within eight words of each other.

Search Options

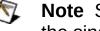
Use the following checkboxes on the **Search** tab to customize a search:

- **Search previous results**—Narrows the results from a search that returned too many topics. You must remove the checkmark from this checkbox to search all topics.
- Match similar words—Broadens a search to return topics that contain words similar to the search terms. For example, a search for "program" lists topics that include the words "programs," "programming," and so on.
- **Search titles only**—Searches only in the titles of topics.

Printing Help File Topics (Windows Only)

Complete the following steps to print an entire book from the **Contents** tab:

- 1. Right-click the book.
- 2. Select **Print** from the shortcut menu to display the **Print Topics** dialog box.
- 3. Select the **Print the selected heading and all subtopics** option.



Note Select Print the selected topic if you want to print the single topic you have selected in the **Contents** tab.

4. Click the **OK** button.

Printing PDF Documents

This help file may contain links to PDF documents. To print PDF documents, click the print button located on the Adobe Acrobat Viewer toolbar.

NI-IMAQdx VIs

Alphabetical
High-Level VIs
Low-Level VIs

NI-IMAQdx VIs

IMAQdx Close Camera

IMAQdx Configure Acquisition

IMAQdx Configure Grab

IMAQdx Discover Ethernet Cameras

IMAQdx Enumerate Attributes

IMAQdx Enumerate Cameras

IMAQdx Enumerate Video Modes

IMAQdx Get Image

IMAQdx Get Image Data

IMAQdx Grab

IMAQdx Open Camera

IMAQdx Property Node

IMAQdx Read Attributes

IMAQdx Read Memory

IMAQdx Read Register

IMAQdx Register Events

IMAQdx Reset Camera

IMAQdx Reset Ethernet Camera Address

IMAQdx Sequence

IMAQdx Snap

IMAQdx Start Acquisition

IMAQdx Stop Acquisition

IMAQdx Unconfigure Acquisition

IMAQdx Write Attributes

IMAQdx Write Memory

IMAQdx Write Register

Changing Palette Views

Complete the following steps to change to another palette view or format.

- Click the **Options** button on the **Controls** or **Functions** palette toolbar to display the Controls/Functions Palettes page of the Options dialog box.
- Select a palette view from the **Palette View** pull-down menu. Select from the following built-in palette views or any custom views:
 - Express—Includes subpalettes on the top level of the Controls and Functions palettes that contain Express VIs and other objects you need to build common measurement applications. The All Controls and All Functions subpalettes contain the complete set of built-in controls, indicators, VIs, and functions. The Express palette view is the default palette view.
 - Advanced—Includes subpalettes on the top level of the Controls and Functions palettes that contain the complete set of built-in controls, indicators, VIs, and functions. The Express subpalettes contain Express VIs and other objects you need to build common measurement applications.

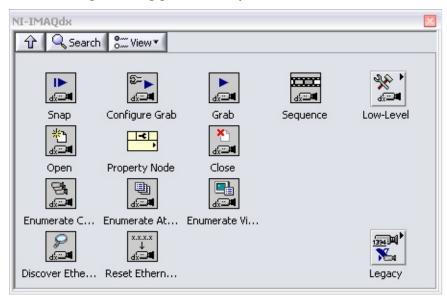


Note In the Express palette view, toolsets and modules do not install subpalettes on the top level of the **Controls** and **Functions** palettes. Instead, toolsets and modules install subpalettes on the **All Controls** and **All Functions** subpalettes. In the Advanced palette view, toolsets and modules install subpalettes on the top level.

- 3. Select a format from the **Format** pull-down menu, such as **Standard**, **All Icons**, **All Text**, or **Icons and Text**.
- 4. Click the **OK** button. The **Controls** and **Functions** palettes change to the palette view and format you selected.

High-Level Acquisition VIs

Use high-level VIs to set up your Vision system and acquire images. The high-level NI-IMAQdx VIs allow you to acquire images, open and close an interface, get/set attributes, get camera features and video modes, and configure triggered acquisitions.



IMAQdx Snap

IMAQdx Configure Grab

IMAQdx Grab

IMAQdx Sequence

IMAQdx Open Camera

IMAQdx Property Node

IMAQdx Close Camera

IMAQdx Enumerate Cameras

IMAQdx Enumerate Attributes

IMAQdx Enumerate Video Modes

IMAQdx Discover Ethernet Cameras

IMAQdx Reset Ethernet Camera Address

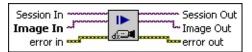
Subpalettes

NI-IMAQdx Low-Level

IMAQdx Snap VI

Installed With: NI Vision Acquisition Software

Configures, starts, acquires, and unconfigures a snap acquisition. Use a snap for low-speed or single-capture applications where ease of programming is essential. If you call this VI before calling the IMAQdx Open Camera VI, the IMAQdx Snap VI uses cam0 by default. If the image type does not match the video format of the camera, this VI changes the image type to a suitable format.



- Session In specifies the name of the camera you wish to reset. The default value is cam0.
- Image In is the reference to the image that receives the captured pixel data.
- error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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 negative error code. If **status** is FALSE, **code** is 0 or a warning code.
 - source identifies where an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

Session Out is a unique reference to the camera. **Session Out** is the same as **Session In**.

Image Out is the reference to the captured image.
 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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

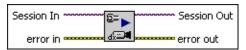
code input identifies the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

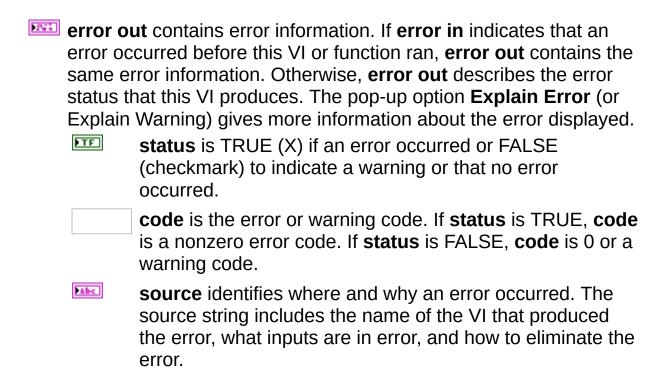
IMAQdx Configure Grab VI

Installed With: NI Vision Acquisition Software

Configures and starts a grab acquisition. A grab performs an acquisition that loops continually on a ring of buffers. Use the grab VI for high-speed image acquisition. Use the IMAQdx Grab VI to copy an image out of the buffer. If you call this VI before calling the IMAQdx Open Camera VI, the IMAQdx Configure Grab VI uses cam0 by default. Use the IMAQdx Unconfigure Acquisition VI to unconfigure the acquisition.



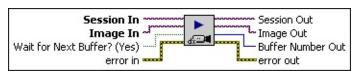
- Session In specifies the name of the camera you wish to reset. The default value is cam0.
- error in describes the error status before this VI or function runs. The default is no error. If an error occurred before this VI or function runs, the VI or function passes the error in value to error out. 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.
 - status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
 - code input identifies the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
 - source string describes the origin of the error or warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.
- Session Out is a unique reference to the camera. Session Out is the same as Session In.



IMAQdx Grab VI

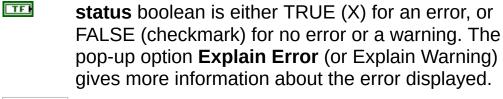
Installed With: NI Vision Acquisition Software

Acquires the most current frame into Image Out. Call this VI only after calling IMAQdx Configure Grab VI. If the image type does not match the video format of the camera, this VI changes the image type to a suitable format.





- Image In is the reference to the image that receives the captured pixel data.
- Wait for Next Buffer? (Yes) if the value is Yes, the driver will wait for the next available buffer. If the Wait for Next Buffer? value is No, the driver will not wait for the next available buffer, and will instead return the last acquired buffer.
- error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.



code input identifies the error or warning. The pop-up option Explain Error (or Explain Warning) gives more

warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed. **▶**1/0 **Session Out** is a unique reference to the camera. **Session** Out is the same as Session In. **D Image Out** is the reference to the captured image. **Buffer Number Out** is the actual acquired buffer number returned. 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, error out describes the error status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed. TF **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. labc **source** identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how

to eliminate the error.

information about the error displayed.

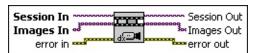
source string describes the origin of the error or

abc

IMAQdx Sequence VI

Installed With: NI Vision Acquisition Software

Configures, starts, acquires, stops, and unconfigures a sequence acquisition. Use this VI to capture multiple images. If you call this VI before calling the IMAQdx Open Camera VI, the IMAQdx Sequence VI uses cam0 by default.



- Session In specifies the name of the camera you wish to reset. The default value is cam0.
- Images In is the reference to the array of images that receives the captured pixel data.
- error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.
 - status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
 - code input identifies the error or warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.
 - source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
- Session Out is a unique reference to the camera. Session Out is

the same as **Session In**.

Images Out is the reference to the array of captured images.

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, error out describes the error

status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

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

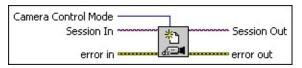
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Open Camera VI

Installed With: NI Vision Acquisition Software

Opens a camera, queries the camera for its capabilities, loads a camera configuration file, and creates a unique reference to the camera. Use the IMAQdx Close Camera VI when you are finished with the reference.



TF

1/0

Camera Control Mode is the control mode of the camera used during image broadcasting. Open a camera in controller mode to actively configure and acquire image data. Open a camera in listener mode to passively acquire image data from a session that was opened in controller mode on a different host or target computer. The default value is Controller.

Session In specifies the name of the camera you wish to open. The default value is cam0.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

code input identifies the error or warning. The popup option **Explain Error** (or Explain Warning) gives

more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

▶1/0

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

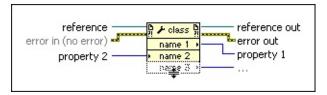
status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

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 identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

Property Node

Gets (reads) and/or sets (writes) properties of a reference. The Property Node automatically adapts to the class of the object that you **reference**. LabVIEW includes Property Nodes preconfigured to access VISA properties, .NET properties, and ActiveX properties. <u>Details</u>



- reference is the refnum associated with the object for which you want to set or get properties. If the Property Node class is Application or VI, you do not have to wire a refnum to this input. For the Application class, the default is the current application instance. For the VI class, the default is the VI containing the Property Node.
- 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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 specifies the origin of the error or warning and is, in most cases, the name of the VI or function that produced the error or warning. The default is an empty string.

property 2n are examples of properties you want to set (write).
reference out returns reference unchanged.
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 front panel indicator 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 describes the origin of the error or warning and is, in most cases, the name of the VI or function that produced the error or warning.
property 1n are examples of properties you want to get (read).

Property Node Details

To select the class on which to execute the property, wire the refnum to the **reference** input. For example, to select the VI, Generic, or Application class, wire the VI, VI object, or application refnum to the **reference** input. The node adapts to the class automatically. You also can right-click the node and select a class from the shortcut menu.

Move the cursor over terminals in the Property Node to display more information about the property in the Context Help window. You also can right-click a property terminal and select **Help For** *Property* from the shortcut menu, where *Property* is the name of the property.

To get property information, right-click the node and select **Change All to Read** from the shortcut menu. To set property information, right-click the node and select **Change All to Write** from the shortcut menu. If a property is read only, **Change to Write** is dimmed in the shortcut menu.

The node executes each terminal in order from top to bottom. If an error occurs on a terminal, the node stops at that terminal, returns an error, and does not execute any further terminals. You can right-click the node and select **Ignore Errors Inside Node** from the shortcut menu to ignore any errors and continue executing further terminals. If an error occurs when **Ignore Errors Inside Node** is set, the property node still returns that error. The **error out** cluster reports which property caused the error.

If the small direction arrow on the property is on the right, you are getting the property value. If the small direction arrow on a property is on the left, you are setting the property value. Properties have a short or long name that you can change by right-clicking and selecting **Name Format** from the shortcut menu. The **No Names** format displays only the data type for each property.

ActiveX Property Node (Windows)

If the property you want to write is variant, you can wire in LabVIEW data types and they automatically convert to variant data types indicated by a coercion dot. ActiveX does not support 64-bit integer data types. If you wire a 64-bit integer data type to a variant parameter of an ActiveX Property Node, LabVIEW converts the data type to a double-precision, floating-point number. If the property is variant, use the Variant To Data function to convert to a LabVIEW data type, if needed. If you right-click the Property Node and select **Select Class»ActiveX»Browse** from the

shortcut menu, LabVIEW displays the Select Object From Type Library dialog box.

On the front panel or block diagram, right-click an ActiveX object, select **Create»Property Node**, and select a property from the shortcut menu to set a property for the object. You also can select ActiveX-specific properties for an ActiveX object. On the block diagram, right-click an ActiveX object, select **Create»Property Node**, and select an ActiveX-specific property from the shortcut menu.

.NET Property Node (Windows)

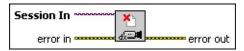
If you right-click the Property Node and select **Select Class».NET»Browse** from the shortcut menu, LabVIEW displays the Select Object From Assembly dialog box.

On the front panel or block diagram, right-click a .NET object, select **Create»Property Node**, and select a property from the shortcut menu to set a property for the object. You also can select .NET-specific properties for a .NET object. On the block diagram, right-click a .NET object, select **Create»Property Node**, and select a .NET-specific property from the shortcut menu.

IMAQdx Close Camera VI

Installed With: NI Vision Acquisition Software

Stops an acquisition in progress, releases resources associated with an acquisition, and closes the specified Camera Session.



Session In is a unique reference to the camera, which you can obtain with the IMAQdx Open Camera VI.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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 negative error code. If status is FALSE, code is 0 or a warning code.

source identifies where an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

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

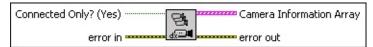
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Enumerate Cameras VI

Installed With: NI Vision Acquisition Software

Returns a list of all cameras on the host computer.



Connected Only? (Yes) If the Connected Only? value is Yes, then the Camera Information Array only contains cameras that are currently connected to the host computer. If the Connected Only? value is No, then the Camera Information Array contains cameras that are currently connected, and were previously connected, to the host computer.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

code input identifies the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Camera Information Array is an array of interface files that are on the host computer. This includes cameras that are currently connected or that were previously connected.

	Type has a value of 3. This designates an NI-IMAQdx interface file.
	Version is the version of the interface file. This number may increment with different versions of the driver as the format of the interface file changes.
	Flags is a bitwise mask of the current interface status. If bit 0 is on (value = 1), the interface represents a camera that is currently connected. If bit 0 is off (value = 0), the interface represents a disconnected camera.
	SerialNumberHigh is the upper 32-bits of the interface serial number. Every camera has a unique value for SerialNumberHigh.
	SerialNumberLow is the lower 32-bits of the interface serial number. Every camera has a unique value for SerialNumberLow.
	BusType specifies the bus type for the camera.
labc	InterfaceName is the name of the interface. Use this name when opening the interface.
Pabe	VendorName is the vendor name of the camera designated for this interface. VendorName varies from camera to camera.
Pabe	ModelName is the model name of the camera designated for this interface. ModelName varies from camera to camera.
Pabc	CameraFileName is the name of the camera file that this interface uses. The camera file contains all the settings for a given camera. You can configure and save these settings from Measurement & Automation Explorer (MAX).
Pabe	CameraAttributeURL is the URL of the XML file that describes the camera attributes.

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, error out describes the error

status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed. **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

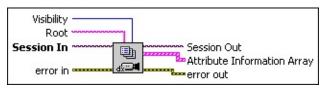
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Enumerate Attributes VI

Installed With: NI Vision Acquisition Software

Gets the attributes supported by the camera.



TF

abc

1/0

200

Visibility specifies the branch of the attributes to enumerate. Simple and Intermediate visibility enumerate a reduced list of attributes. Advanced visibility enumerates all available attributes.

Root specifies the branch of the attribute tree to enumerate. Specify an empty string to enumerate the entire attribute tree.

Session In specifies the name of the camera you wish to reset. The default value is cam0.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

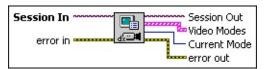
code input identifies the error or warning. The popup option **Explain Error** (or Explain Warning) gives more information about the error displayed.

	source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
▶ 1/0	Session Out is a unique reference to the camera. Session Out is the same as Session In.
[90]	Attribute Information Array is an array of possible features supported by the camera.
	Attribute Type specifies the type of attribute. Possible attribute type values include U32, I64, DBL, String, Enum, Bool, and Command.
	Readable returns True if readable.
	Writable returns True if writable.
	Attribute Name specifies the fully qualified name of the attribute located in the attribute tree.
	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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed. 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 identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Enumerate Video Modes VI

Installed With: NI Vision Acquisition Software

Returns a list of video modes supported by the camera.



Session In specifies the name of the camera you wish to reset. The default value is cam0.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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 negative error code. If status is FALSE, code is 0 or a warning code.

source identifies where an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

Video Modes is an array of video modes supported by the current camera.

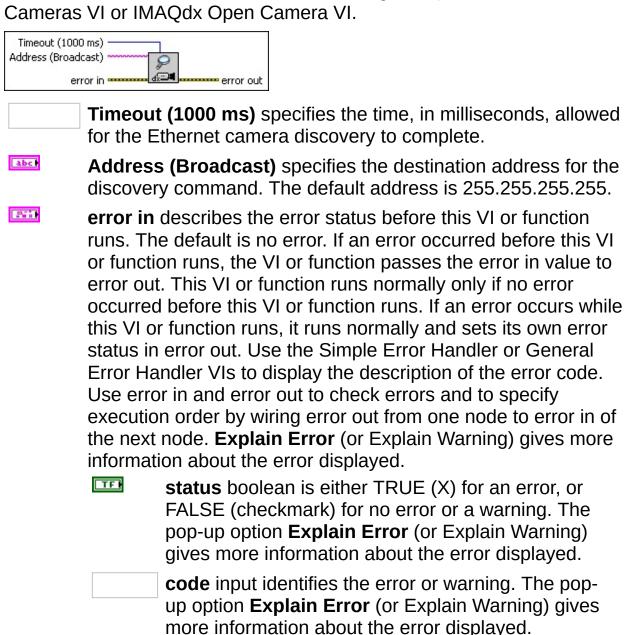
	Video Mode is the index for this video mode.
Pabe	Video Mode Name VideoMode Name is the name of the video mode, such as 640 x 480 Mono8.
	t Mode is the index into the Video Modes array of the mode used by the camera.
an error contains describe option E informa	ut contains error information. If error in indicates that r occurred before this VI or function ran, error out s the same error information. Otherwise, error out es the error status that this VI produces. The pop-up Explain Error (or Explain Warning) gives more ation about the error displayed.
FTF	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.
labc	source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Discover Ethernet Cameras VI

Installed With: NI Vision Acquisition Software

abc

Initiates a round of Ethernet camera discovery. Use this VI to find Ethernet cameras on the network with a remote subnet. This VI will be blocked and will return after the specified timeout. The address specifies the destination address for the discovery command. The default address is 255.255.255.255. Call this VI before calling IMAQdx Enumerate Cameras VI or IMAQdx Open Camera VI.



source string describes the origin of the error or

warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

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

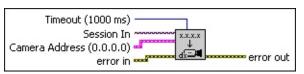
code is the error or warning code. If status is TRUE,code is a nonzero error code. If status is FALSE,code is 0 or a warning code.

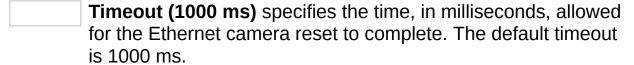
source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Reset Ethernet Camera Address VI

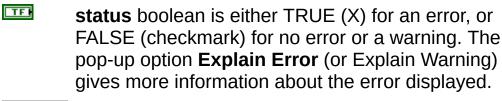
Installed With: NI Vision Acquisition Software

Reset Ethernet camera address. Use this function to reset Ethernet cameras on the network with a local subnet. This function will be blocked and will return when the reset is complete or after the specified timeout.





- Session In specifies the name of the camera you wish to reset. The default value is cam0.
- Camera Address (0.0.0.0) specifies the new address, subnet and gateway for the Ethernet camera to reset to. Specify a valid address to configure the camera's new address. Specify an empty address to merely reset the camera Ethernet address. The default address is 0.0.0.0.
- error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.



code input identifies the error or warning. The pop-

up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

abc

source string describes the origin of the error or warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

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

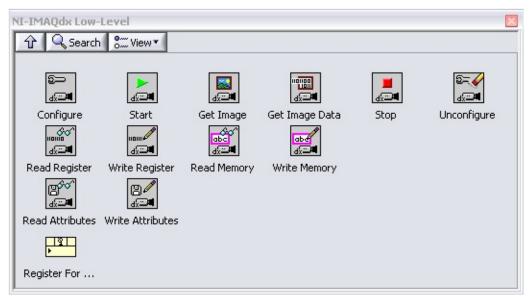
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred.

The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

Low-Level VIs

Use low-level NI-IMAQdx VIs for more direct control of your image acquisition hardware.



IMAQdx Configure Acquisition

IMAQdx Start Acquisition

IMAQdx Get Image

IMAQdx Get Image Data

IMAQdx Stop Acquisition

IMAQdx Unconfigure Acquisition

IMAQdx Read Register

IMAQdx Write Register

IMAQdx Read Memory

IMAQdx Write Memory

IMAQdx Reset Camera

IMAQdx Register for Events

IMAQdx Read Attributes

IMAQdx Write Attributes

IMAQdx Reset Camera

IMAQdx Configure Acquisition VI

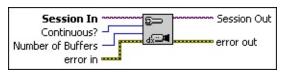
Installed With: NI Vision Acquisition Software

Configures a low-level acquisition previously opened with IMAQdx Open Camera VI. Specify the acquisition type with the **Continuous?** and **Number of Buffers** parameters.

Snap: Continuous = 0; Buffer Count = 1

Sequence: Continuous = 0; Buffer Count > 1

Grab: Continuous = 1; Buffer Count > 1



I/0

Session In is a unique reference to the camera, which you can obtain with the IMAQdx Open Camera VI.

Continuous? specifies whether the acquisition is continuous or one shot.

Number of Buffers For a one shot acquisition, this parameter specifies the number of images to acquire. For a continuous acquisition, this parameter specifies the number of buffers the driver uses internally.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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.

TF

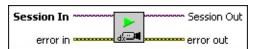
status is TRUE (X) if an error occurred before this VI or function ran or FALSE (checkmark) to indicate a warning or that no error occurred before this VI or function ran. The default is FALSE.

		code is the error or warning code. The default is 0. If status is TRUE, code is a negative error code. If status is FALSE, code is 0 or a warning code.
	abc	source identifies where an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.
) 1/0		Out is a unique reference to the camera. Session ne same as Session In.
F	an error contains describe option E informat	occurred before this VI or function ran, error out the same error information. Otherwise, error out the same error information. Otherwise, error out the error status that this VI produces. The pop-up explain Error (or Explain Warning) gives more ion about the error displayed.
	FTF	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.
	Pabc	source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Start Acquisition VI

Installed With: NI Vision Acquisition Software

Starts an acquisition that was previously configured with the IMAQdx Configure Acquisition VI. Use the IMAQdx Stop Acquisition VI to stop the acquisition.



Session In specifies the name of the camera you wish to reset. The default value is cam0.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

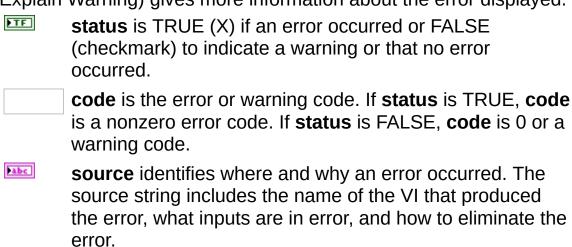
code input identifies the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

error out contains error information. If error in indicates that an error occurred before this VI or function ran, error out contains the

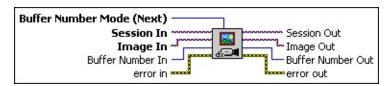
same error information. Otherwise, **error out** describes the error status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.



IMAQdx Get Image VI

Installed With: NI Vision Acquisition Software

Acquires the specified frame into **Image Out**. Call this VI only after calling the IMAQdx Configure Acquisition VI. If the image type does not match the video format of the camera, this VI changes the image type to a suitable format.



Buffer Number Mode (Next) specifies the buffer number mode. The default value is Next, which specifies that the driver will wait for the next acquired buffer. Other possible values are Last, which specifies that the driver will not wait for the next buffer but will instead return the last acquired buffer, and Buffer Number, which specifies an exact buffer number to wait for. Use Buffer Number Mode in conjunction with Buffer Number In.

Session In specifies the name of the camera you wish to reset. The default value is cam0.

Image In is the reference to the image that receives the captured pixel data.

Buffer Number In is the cumulative image number to get.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

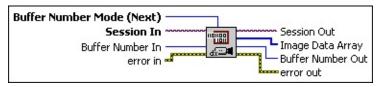
1/0

	TE	status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.		
		code input identifies the error or warning. The popup option Explain Error (or Explain Warning) gives more information about the error displayed.		
	abc	source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.		
170		n Out is a unique reference to the camera. Session the same as Session In.		
	Image	Out is the reference to the captured image.		
	Buffer returne	Number Out is the actual acquired buffer number d.		
Part	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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.			
	FTF	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.		
	Pabc	source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.		

IMAQdx Get Image Data VI

Installed With: NI Vision Acquisition Software

Copies the raw data of the specified frame into **Image Data Array**. Call this VI only after calling the IMAQdx Configure Acquisition VI.



Buffer Number Mode (Next) specifies the buffer number mode. The default value is Next, which specifies that the driver will wait for the next acquired buffer. Other possible values are Last, which specifies that the driver will not wait for the next buffer but will instead return the last acquired buffer, and Buffer Number, which specifies an exact buffer number to wait for. Use Buffer Number Mode in conjunction with Buffer Number In.

Session In specifies the name of the camera you wish to reset. The default value is cam0.

Buffer Number In is the cumulative image number to get.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

I/0

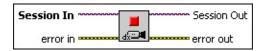
TF

	code input identifies the error or warning. The popup option Explain Error (or Explain Warning) gives more information about the error displayed. source string describes the origin of the error or
	warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
170	Session Out is a unique reference to the camera. Session Out is the same as Session In.
	Image Data Array is an array of unsigned bytes that represent the raw image data as transferred from the camera.
	Buffer Number Out is the actual acquired buffer number returned.
Pail	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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
	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 identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Stop Acquisition VI

Installed With: NI Vision Acquisition Software

Stops an acquisition previously started with the IMAQdx Start Acquisition VI.



Session In specifies the name of the camera you wish to reset. The default value is cam0.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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 boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

option **Explain Error** (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

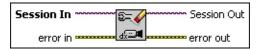
error out contains error information. If error in indicates that an error occurred before this VI or function ran, error out contains the same error information. Otherwise, error out describes the error status that this VI produces. The pop-up option Explain Error (or

Explain Warning) gives more information about the error displayed.
 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 identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Unconfigure Acquisition VI

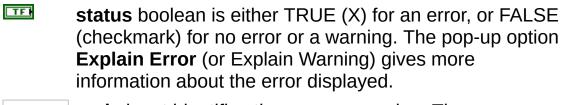
Installed With: NI Vision Acquisition Software

Unconfigures an acquisition previously configured with the IMAQdx Configure Acquisition VI.



Session In specifies the name of the camera you wish to reset. The default value is cam0.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.



code input identifies the error or warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

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, error out describes the error

status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed. **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

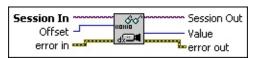
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Read Register VI

Installed With: NI Vision Acquisition Software

Accesses registers on the camera and reads a 32-bit value from the camera. Data is byte-swapped for little endian alignment after transfer.



1/0

Session In is a unique reference to the camera, which you can obtain with the IMAQdx Open Camera VI.

Offset is the register location to access. Refer to the camera documentation for more information about camera-specific register ranges. Note: Use the **Property Node Properties»Camera Information»Base Address** to get the base address for your camera.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

TF

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

code input identifies the error or warning. The popup option **Explain Error** (or Explain Warning) gives more information about the error displayed.

abc

source string describes the origin of the error or warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the

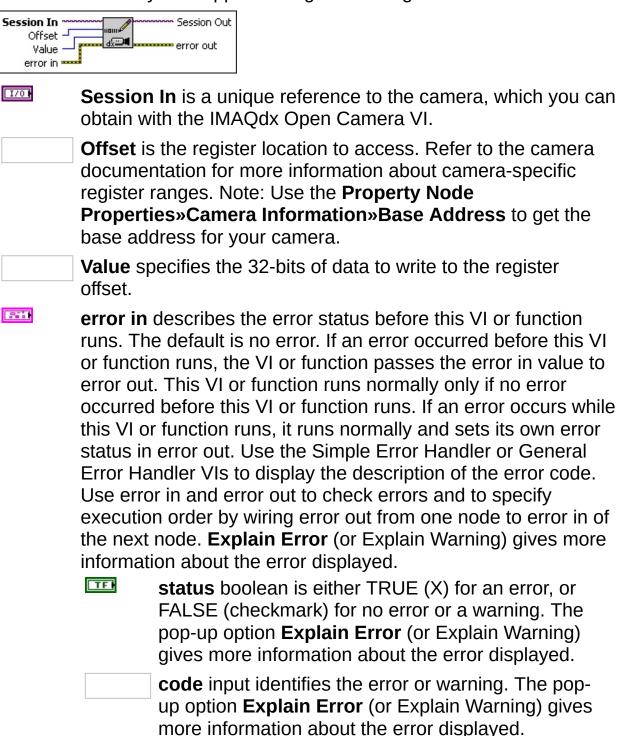
error displayed.

▶ 1/0	Out is t	n Out is a unique reference to the camera. Session he same as Session In. specifies the 32-bits of data read from the register offset.	
Pers	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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.		
	▶ TF	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.	
	Pabc	source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.	

IMAQdx Write Register VI

Installed With: NI Vision Acquisition Software

Accesses registers on the camera and writes a 32-bit value to the camera. Data is byte-swapped for big endian alignment before transfer.



source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

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

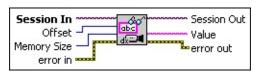
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Read Memory VI

Installed With: NI Vision Acquisition Software

Accesses registers on the camera and reads a string from the camera.



abc

1/0

Session In specifies the name of the camera you wish to reset. The default value is cam0.

Offset is the register location to access. Refer to the camera documentation for more information about camera-specific register ranges. Note: Use the **Property Node**

Properties»Camera Information»Base Address to get the base address for your camera.

Memory Size specifies the number of bytes to read.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

code input identifies the error or warning. The popup option **Explain Error** (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option **Explain Error** (or

Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

Value specifies the string read from the memory offset.

P-11

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

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 identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Write Memory VI

Installed With: NI Vision Acquisition Software

Accesses registers on the camera and writes a string to the camera.



TF

abc

1/0

200

Session In specifies the name of the camera you wish to reset. The default value is cam0.

Offset is the register location to access. Refer to the camera documentation for more information about camera-specific register ranges. Note: Use the Property Node Properties»Camera Information»Base Address to get the base address for your camera.

Value specifies the string to write to the memory offset.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

code input identifies the error or warning. The popup option Explain Error (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option **Explain Error** (or

Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

1944

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, error out describes the error status that this VI produces. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

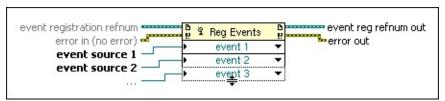
status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

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 identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

Register For Events

Dynamically registers events. The events for which you can register depend on the type of the reference you wire to each **event source** input. Wire the **event reg refnum out** output to an Event structure or to another Register For Events function. <u>Details</u>



- **event registration refnum** is a reference to an existing event registration a Register For Events function created.
- 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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 specifies the origin of the error or warning and is, in most cases, the name of the VI or function that produced the error or warning. The default is an empty string.
- event source 1..n is a reference to an application, VI, control, or user event. References must be to local objects. You cannot wire a reference to a remote object. If you wire an array or cluster of refnums to this input, LabVIEW registers all elements of the array

or cluster for the event.

event reg refnum out returns the reference to a new or existing
event registration.

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 front panel indicator and select Explain Error from the shortcut menu for more information about the error.

TF	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 describes the origin of the error or warning and is, in most cases, the name of the VI or function that produced the error or warning.

Register For Events Details

Each **event source** input is a reference to an application, VI, control, or user event. You can right-click each data item and select which event to register for that event source. You can wire the **event reg refnum out** terminal of the Register For Events function to the dynamic event terminals on the Event structure border, to the Unregister For Events function, or to the top left input of another Register For Events function. If you wire the top left input of the Register For Events function, the function modifies the existing registration information associated with that refnum instead of registering the event again.

Registered events stay registered until you explicitly unregister them or until the VI that registered the events finishes running or you abort the VI. If the VI which registered for events was a subVI, events are unregistered when that VI's top-level VI finishes execution or is aborted. If you use the Run VI method to execute a subVI that registered for events, events are unregistered when the subVI finishes execution or is aborted.

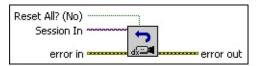


Note LabVIEW does not include an event registration refnum on the Controls palette because the event registration refnum is strictly typed and a generic version does not exist. You can create an event registration refnum by configuring a Register For Events function or right-clicking the function and creating a control or indicator from the shortcut menu. You also can select the part of the application that includes the event registration refnum and selecting Edit»Create SubVI to create a subVI from the selection. If you later modify the Register For Events function and change the type of the event registration refnum, you must recreate the control or indicator to match.

IMAQdx Reset Camera VI

Installed With: NI Vision Acquisition Software

Performs a manual reset on a camera. Stops any ongoing acquisitions.



- Reset All? (No) If the value is No, then only the specified camera will be reset. If the Reset All? value is Yes, then all of the connected cameras will be reset.
- Session In specifies the name of the camera you wish to reset. The default value is cam0.
- 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. **Explain Error** (or Explain Warning) gives more information about the error displayed.
 - status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
 - code input identifies the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
 - source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.
- 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, error out describes the error

status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed. **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

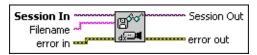
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

IMAQdx Read Attributes VI

Installed With: NI Vision Acquisition Software

Loads a configuration file for a camera.



Session In specifies the name of the camera you wish to reset. The default value is cam0.

Filename specifies the name of the source file to read from.

error in describes the error status 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 the Simple Error Handler or General Error Handler VIs to display the description of the error code. 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. Explain Error (or Explain Warning) gives more information about the error displayed.

status boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

code input identifies the error or warning. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

source string describes the origin of the error or warning. The pop-up option Explain Error (or Explain Warning) gives more information about the error displayed.

Session Out is a unique reference to the camera. Session Out is the same as Session In.

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, error out describes the error

status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed. **status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

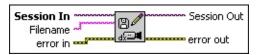
code is the error or warning code. If status is TRUE, code is a nonzero error code. If status is FALSE, code is 0 or a warning code.

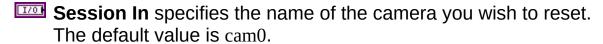
source identifies where and why an error occurred. The source string includes the name of the VI that produced the error, what inputs are in error, and how to eliminate the error.

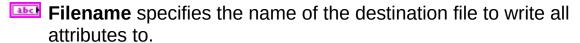
IMAQdx Write Attributes VI

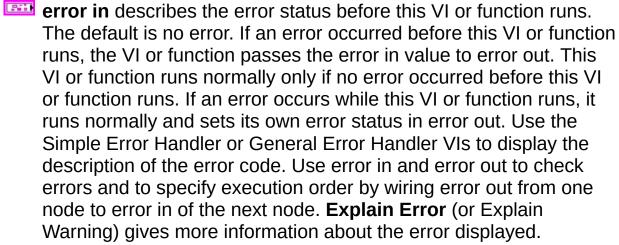
Installed With: NI Vision Acquisition Software

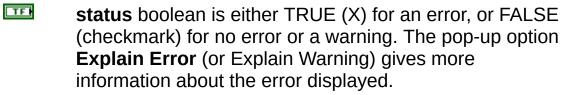
Saves a configuration file for a camera.

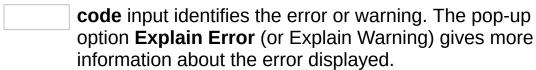


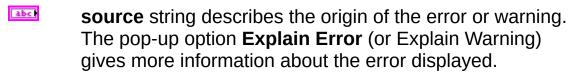






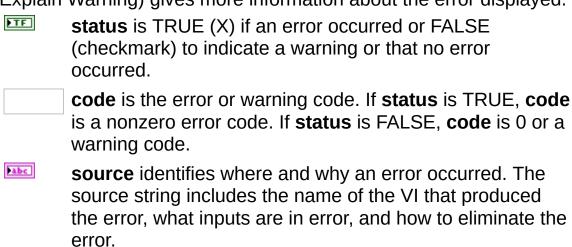






- Session Out is a unique reference to the camera. Session Out is the same as Session In.
- error out contains error information. If error in indicates that an error occurred before this VI or function ran, error out contains the

same error information. Otherwise, **error out** describes the error status that this VI produces. The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.



Vision Express

Use the Vision Express VIs to quickly develop common image acquisition and processing applications.

Palette Object	Description
	Use NI Vision Assistant from within the LabVIEW environment to perform common image processing tasks. You must install the NI Vision Development Module to use the Vision Assistant Express VI.
•	Acquires images from cameras using NI-IMAQ or NI-IMAQdx. Refer to the <i>NI Vision Acquisition Express VI Help</i> for more information.

Vision Acquisition Express VI

Acquires images from cameras using NI-IMAQ or NI-IMAQdx. Refer to the *NI Vision Acquisition Express VI Help* for more information. The *NI Vision Acquisition Express VI Help* installs to the <LabVIEW>\help directory, where <LabVIEW> is the location to which you installed LabVIEW.

NI-IMAQdx Properties

Refer to the class name below for the properties associated with each NI-IMAQdx class. Refer to the *NI-IMAQdx Function Reference Help* for more information about NI-IMAQdx properties and attributes.

Attribute	Long Name	Туре	Description
BaseAddress	Camera Information : Base Address	U321	Read only. Gets the base address of the camera registers.
BusType	Camera Information : Bus Type		Read only. Gets the bus type of the camera.
ModelName	Camera Information : Model Name	abc	Read only. Returns the model name.
SerialNumberHigh	Camera Information : Serial Number High	U321	Read only. Gets the upper 32-bits of the camera 64- bit serial number.
SerialNumberLow	Camera Information : Serial Number Low	U321	Read only. Gets the lower 32-bits of the camera 64- bit serial number.
VendorName	Camera Information : Vendor Name	abci	Read only. Returns the vendor name.
HostIPAddress	Camera Information : Host IP Address	åbc	Read only. Returns the host adapter IP address.
IPAddress	Camera Information : IP Address	abci	Read only. Returns the IP address.
PrimaryURLString	Camera	abc	Read only. Gets

	Information : Primary URL String		the camera's primary URL string.
SecondaryURLString	Camera Information : Secondary URL String	àbc	Read only. Gets the camera's secondary URL string.
AcquisitionInProgress	Status Information : Acquisition In Progress	TF	Read only. Gets the current state of the acquisition. TRUE if acquiring; otherwise FALSE.
BufferCount	Status Information : Last Buffer Count	U32 l	Read only. Gets the number of transferred buffers.
LastBufferNumber	Status Information : Last Buffer Number	U32 I	Read only. Gets the last cumulative buffer number transferred.
LostBufferCount	Status Information : Lost Buffer Count	U32 I	Read only. Gets the number of lost buffers during an acquisition session.
LostPacketCount	Status Information : Lost Packet Count	U321	Read only. Gets the number of lost packets during an acquisition session.
RequestedResendPackets	Status Information: Requested Resend Packet Count	U32	Read only. Gets the number of packets requested to be resent during an

			acquisition session.
ReceivedResendPackets	Status Information : Received Resend Packets	U321	Read only. Gets the number of packets that were requested to be resent during an acquisition session and were completed.
BayerGainB	Acquisition Attributes : Bayer : Gain B	DBL	Sets/gets the white balance gain for the blue component of the Bayer conversion.
BayerGainG	Acquisition Attributes : Bayer : Gain G	DBL	Sets/gets the white balance gain for the green component of the Bayer conversion.
BayerGainR	Acquisition Attributes : Bayer : Gain R	DBL	Sets/gets the white balance gain for the red component of the Bayer conversion.
BayerPattern	Acquisition Attributes : Bayer : Pattern		Sets/gets the Bayer pattern to use.
StreamChannelMode	Acquisition Attributes: Controller: Stream Channel Mode		Gets/sets the mode for allocating a FireWire stream channel.
DesiredStreamChannel	Acquisition Attributes : Controller :	U32 I	Gets/sets the stream channel to manually allocate.

	Desired Stream Channel		
FrameInterval	Acquisition Attributes : Frame Interval	U32 I	Read only. Gets the duration in milliseconds between successive frames.
IgnoreFirstFrame	Acquisition Attributes : Ignore First Frame	TE	Gets/sets the video delay of one frame between starting the camera and receiving the video feed.
OffsetX	Acquisition Attributes : Offset X	U321	Gets/sets the left offset of the image.
OffsetY	Acquisition Attributes : Offset Y	U321	Gets/sets the top offset of the image.
Width	Acquisition Attributes : Width	U321	Gets/sets the width of the image.
Height	Acquisition Attributes : Height	U321	Gets/sets the height of the image.
PixelFormat	Acquisition Attributes : Pixel Format	U321	Gets/sets the pixel format of the source sensor.
PacketSize	Acquisition Attributes : Packet Size	U321	Gets/sets the packet size in bytes.
PayloadSize	Acquisition Attributes : Payload Size	U321	Gets/sets the frame size in bytes.

Speed	Acquisition Attributes : Speed		Gets/sets the transfer speed in Mbps for a FireWire packet.
ShiftPixelBits	Acquisition Attributes : Shift Pixel Bits	TF	Gets/sets the alignment of 16-bit cameras. Downshift the pixel bits if the camera returns most significant bit-aligned data.
SwapPixelBytes	Acquisition Attributes : Swap Pixel Bytes	TE	Gets/sets the endianness of 16-bit cameras. Swap the pixel bytes if the camera returns little endian data.
OverwriteMode	Acquisition Attributes : Overwrite Mode		Gets/sets the overwrite mode, used to determine acquisition when an image transfer cannot be completed due to an overwritten internal buffer.
Timeout	Acquisition Attributes : Timeout	U321	Gets/sets the timeout value in milliseconds, used to abort an acquisition when the image transfer cannot be completed within the delay.

VideoMode	Acquisition Attributes : Video Mode	U321	Gets/sets the video mode for a camera.
BitsPerPixel	Acquisition Attributes : Bits Per Pixel		Gets/sets the actual bits per pixel. For 16-bit components, this represents the actual bit depth (10-, 12-, 14-, or 16-bit).
ReserveDualPackets	Acquisition Attributes : Reserve Dual Packets	TF	Gets/sets if dual packets will be reserved for a very large FireWire packet.
ReceiveTimestampMode	Acquisition Attributes: Receive Timestamp Mode		Gets/sets the mode for timestamping images received by the driver.
ActualPeakBandwidth	Acquisition Attributes: Advanced Ethernet: Bandwidth Control: Actual Peak Bandwidth	DBL	Read only. Returns the actual maximum peak bandwidth the camera will be configured to use.
DesiredPeakBandwidth	Acquisition Attributes: Advanced Ethernet: Bandwidth Control: Desired Peak Bandwidth	DBL	Gets/sets the desired maximum peak bandwidth the camera should use.

DestinationMode	Acquisition Attributes: Advanced Ethernet: Controller: Destination Mode		Gets/Sets where the camera is instructed to send the image stream.
DestinationMulticastAddress	Acquisition Attributes: Advanced Ethernet: Controller: Destination Multicast Address	abc	Gets/Sets the multicast address the camera should send data in multicast mode.
LostPacketMode	Acquisition Attributes: Advanced Ethernet: Lost Packet Mode		Gets/sets the behavior when the user extracts a buffer that has missing packets.
MemoryWindowSize	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Memory Window Size	U32 I	Gets/sets the size of the memory window of the camera in kilobytes. Should match the camera's internal buffer size.
ResendsEnabled	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Resends Enabled	TF	Gets/sets if resends will be issued for missing packets.
ResendThresholdPercentage	Acquisition	U32	Gets/sets the

	Attributes: Advanced Ethernet: Resend Parameters: Resend Threshold Percentage		threshold of the packet processing window that will trigger packets to be resent.
ResendBatchingPercentage	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Resend Batching Percentage	U321	Gets/sets the percent of the packet resend threshold that will be issued as one group past the initial threshold sent in a single request.
MaxResendsPerPacket	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Max Resends Per Packet	U321	Gets/sets the maximum number of resend requests that will be issued for a missing packet.
ResendResponseTimeout	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Resend Response Timeout	U32	Gets/sets the time to wait for a resend request to be satisfied before sending another.
NewPacketTimeout	Acquisition Attributes : Advanced	U32	Gets/sets the time to wait for new packets to

	Ethernet : Resend Parameters : New Packet Timeout		arrive in a partially completed image before assuming the rest of the image was lost.
MissingPacketTimeout	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Missing Packet Timeout	U32 I	Gets/sets the time to wait for a missing packet before issuing a resend.
ResendTimerResolution	Acquisition Attributes: Advanced Ethernet: Resend Parameters: Resend Timer Resolution	U321	Gets/sets the resolution of the packet processing system that is used for all packet-related timeouts.
ActiveAttribute	Camera Attributes : Active Attribute	abc	Gets/sets the active attribute
ValueU32	Camera Attributes : Value : U32	U321	Gets/sets the current value as an integer for the active attribute.
Valuel64	Camera Attributes : Value : I64	164)	Gets/sets the current value as a 64-bit integer for the active attribute.
ValueDBL	Camera Attributes :	DBL	Gets/sets the current value as a

	Value : DBL		double precision floating point number for the active attribute.
ValueString	Camera Attributes : Value : String	abc	Gets/sets the current value as a string for the active attribute.
ValueBool	Camera Attributes : Value : Bool	TF	Gets/sets the current value as a boolean for the active attribute.
MinU32	Camera Attributes : Minimum : U32	U32 I	Read only. Gets the current minimum as an integer for the active attribute.
MinI64	Camera Attributes : Minimum : 164	1641	Read only. Gets the current minimum as a 64- bit integer for the active attribute.
MinDBL	Camera Attributes : Minimum : DBL	DBL	Read only. Gets the current minimum as a double precision floating point number for the active attribute.
MinString	Camera Attributes : Minimum : String	abc	Read only. Gets the current minimum as a string for the active attribute.
MaxU32	Camera Attributes :	U32 l	Read only. Gets the current

	Maximum : U32		maximum as an integer for the active attribute.
MaxI64	Camera Attributes : Maximum : 164	1641	Read only. Gets the current maximum as a 64-bit integer for the active attribute.
MaxDBL	Camera Attributes : Maximum : DBL	DBL	Read only. Gets the current maximum as a double precision floating point number for the active attribute.
MaxString	Camera Attributes : Maximum : String	abc	Read only. Gets the current maximum as a string for the active attribute
Туре	Camera Attributes : Type		Read only. Gets the active type for the active attribute.
EnumValues[]	Camera Attributes : Enum : Values[]	[032]	Read only. Returns an array of enum attribute values for the active attribute.
EnumStrings[]	Camera Attributes : Enum : Strings[]	[abc)	Read only. Returns an array of enum attribute strings for the active attribute.
IsReadable	Camera	TF	Read only. Gets

	Attributes : Is Readable	the read access for the active attribute. TRUE if readable; otherwise FALSE.
IsWritable	Camera Attributes : Is Writable	Read only. Gets the write access for the active attribute. TRUE if writable; otherwise FALSE.
IncU32	Camera Attributes: Increment: U32	Read only. Gets the increment as an integer for the active attribute.
Incl64	Camera Attributes: Increment: 164	Read only. Gets the increment as a 64-bit integer for the active attribute.
IncDBL	Camera Attributes: Increment: DBL	Read only. Gets the increment as a double precision floating point number for the active attribute.
IncString	Camera Attributes: Increment: String	Read only. Gets the increment as a string for the active attribute.
Tooltip	Camera Attributes: Tooltip	Read only. Gets the tooltip for the active attribute.
Units	Camera Attributes :	Read only. Gets the units for the

	Units		active attribute.
Visibility	Camera Attributes : Visibility		Read only. Gets the visibility for the active attribute.
Description	Camera Attributes : Description	àbc	Read only. Gets the description for the active attribute.
DisplayName	Camera Attributes : Display Name	àbc	Read only. Gets the display name for the active attribute.

LabVIEW VI Error Handling

Every NI-IMAQdx VI contains an **error in** input cluster and an **error out** output cluster. The clusters contain a Boolean that indicates whether an error occurred, the code for the error, and the source or name of the VI that returned the error. If **error in** indicates an error, the VI passes the error information to **error out** and does not execute any NI-IMAQ function.

You can use the Simple Error Handler VI (**Functions»Time&Dialog**) to check for errors that occur while executing a VI. If you wire an error cluster to the Simple Error Handler VI, the VI deciphers the error information and displays a dialog box that describes the error. If no error occurred, the Simple Error Handler VI does nothing.

Error Codes

Error Codes

The following table describes the error codes used in NI-IMAQdx.

Error Code	Status Name	Descriptio
-1074360320	IMAQdxErrorSystemMemoryFull	Not enough memory
-1074360319	IMAQdxErrorInternal	Internal err
-1074360318	IMAQdxErrorInvalidParameter	Invalid parameter
-1074360317	IMAQdxErrorInvalidPointer	Invalid pointer
-1074360316	IMAQdxErrorInvalidInterface	Invalid camera session
-1074360315	IMAQdxErrorInvalidRegistryKey	Invalid registry key
-1074360314	IMAQdxErrorInvalidAddress	Invalid address
-1074360313	IMAQdxErrorInvalidDeviceType	Invalid device type
-1074360312	IMAQdxErrorNotImplemented	Not implemente yet
-1074360311	IMAQdxErrorCameraNotFound	Camera no found
-1074360310	IMAQdxErrorCameraInUse	Camera is already in use.
-1074360309	IMAQdxErrorCameraNotInitialized	Camera is not initialized.
-1074360308	IMAQdxErrorCameraRemoved	Camera ha been removed.

-1074360307	IMAQdxErrorCameraRunning	Acquisition in progress
-1074360306	IMAQdxErrorCameraNotRunning	No acquisition in progress
-1074360305	IMAQdxErrorAttributeNotSupported	Attribute n supported by the camera.
-1074360304	IMAQdxErrorAttributeNotSettable	Unable to set attribut
-1074360303	IMAQdxErrorAttributeNotReadable	Unable to get attribut
-1074360302	IMAQdxErrorAttributeOutOfRange	Attribute value is ou of range.
-1074360301	IMAQdxErrorBufferNotAvailable	Requested buffer is unavailabl
-1074360300	IMAQdxErrorBufferListEmpty	Buffer list in empty. Add one or mobuffers.
-1074360299	IMAQdxErrorBufferListLocked	Buffer list already locked. Reconfigu acquisitior and try again.
-1074360298	IMAQdxErrorBufferListNotLocked	No buffer list. Reconfigu acquisitior and try again.

-1074360297	IMAQdxErrorResourcesAllocated	Transfer engine resources already allocated. Reconfiguacquisition and try again.
-1074360296	IMAQdxErrorResourcesUnavailable	Insufficien transfer engine resources
-1074360295	IMAQdxErrorAsyncWrite	Unable to perform asychrono register write.
-1074360294	IMAQdxErrorAsyncRead	Unable to perform asychrono register read.
-1074360293	IMAQdxErrorTimeout	Timeout
-1074360292	IMAQdxErrorBusReset	Bus reset occurred during a transactio
-1074360291	IMAQdxErrorInvalidXML	Unable to load camera's XML file.
-1074360290	IMAQdxErrorFileAccess	Unable to read/write file.
-107/360280	IMAQdxErrorInvalidCameraURLString	Camera h

		malformed URL string
-1074360288	IMAQdxErrorInvalidCameraFile	Invalid camera file
-1074360287	IMAQdxErrorGenICamError	Unknown Genicam error.
-1074360286	IMAQdxErrorFormat7Parameters	For format The combinatio of speed, image position, image size and color coding is incorrect.
-1074360285	IMAQdxErrorInvalidAttributeType	The attributype is not compatible with the passed variable type.
-1074360284	IMAQdxErrorDLLNotFound	The DLL could not b found.
-1074360283	IMAQdxErrorFunctionNotFound	The functic could not b found.
-1074360282	IMAQdxErrorLicenseNotActivated	License no activated.
-1074360281	IMAQdxErrorCameraNotConfiguredForListener	The camer is not configured properly to

		support a listener.
-1074360280	IMAQdxErrorCameraMulticastNotAvailable	Unable to configure the system for multicas support.
-1074360279	IMAQdxErrorBufferHasLostPackets	The requested buffer has lost packet and the use requested an error to be generated.
-1074360278	IMAQdxErrorGiGEVisionError	Unknown GiGE Visio error.
-1074360277	IMAQdxErrorNetworkError	Unknown network error.
-1074360276	IMAQdxErrorCameraUnreachable	Unable to connect to the camera
-1074360275	IMAQdxErrorHighPerformanceNotSupported	High performant acquisition not supported on the specified network interface. Connect th camera to a network

	interface running the high performand driver.
--	-----------------------------------------------------------

Glossary

A B C D E F G H I L M N O P Q R S T

U V W Y

Α

A/D Analog-to-digital.

AC Alternating current.

The image size specific to a video standard or acquisition

window camera resolution.

active line The region of lines actively being stored; defined by a region

line start (relative to vertical sync signal) and a line

count.

active pixel

The region of pixels actively being stored; defined by region a pixel start (relative to the horizontal sync signal)

and a pixel count.

Analog-to-digital converter. An electronic device, **ADC**

often an integrated circuit, that converts an analog

voltage to a digital number.

Character code that identifies a specific location (or address

series of locations) in memory.

American National Standards Institute. ANSI

antichrominance Removes the color information from the video signal.

filter

API Application programming interface.

A rectangular portion of an acquisition window or area

frame that is controlled and defined by software.

Ordered, indexed set of data elements of the same array

type.

ASIC Application-specific integrated circuit. A proprietary

semiconductor component designed and

manufactured to perform a set of specific functions

for a specific customer.

The ratio of a picture or image's width to its height. aspect ratio

asynchronous (1) Independent in time from any other event. (2)

> Communication mechanism on the IEEE 1394 bus, which guarantees delivery of the message but does

not guarantee timing.

В

back The area of the video signal between the rising edge of the

porch horizontal sync signal and the active video information.

Bayer Method to produce color images with a single imaging

encoding sensor, as opposed to three individual sensors for the red,

green, and blue components of light.

Bayer Color filter array pattern that can appear in four variations,

pattern depending on the current left and top offsets of the

acquisition window:

GBGB GRGR BGBG RGRG RGRG BGBG GRGR GBGB

big Describes computers that store bytes of memory by placing endian the most significant byte at the memory location with the

the most significant byte at the memory location with the lowest address, the next significant byte at the next memory

location, and so on.

black The level that represents the darkest an image can get. **See**

reference also white reference level.

level

BMP Bitmap. Image file format commonly used for 8-bit and color

images (extension .bmp).

buffer Temporary storage for acquired data.

bus The group of conductors that interconnect individual circuitry

in a computer, such as the PCI bus; typically the expansion

vehicle to which I/O or other devices are connected.

C

cache High-speed processor memory that buffers commonly

used instructions or data to increase processing

throughput.

camera

A process-safe handle to a camera.

session

CCIR Comite Consultatif International des

Radiocommunications. A committee that developed

standards for color video signals.

chrominance The color information in a video signal.

CMOS Complementary metal-oxide semiconductor.

CompactPCI Refers to the core specification defined by the PCI

Industrial Computer Manufacturer's Group (PICMG).

compiler A software utility that converts a source program in a

high-level programming language, such as Basic, C, or Pascal, into an object or compiled program in machine language. Compiled programs run 10 to 1,000 times faster than interpreted programs. **See also** interpreter.

conversion device

Device that transforms a signal from one form to another;

for example, analog-to-digital converters (ADCs) for

analog input and digital-to-analog converters (DACs) for

analog output.

CPU Central processing unit.

CSYNC Composite sync signal. A combination of the horizontal

and vertical sync pulses.

D

D/A Digital-to-analog.

DAC Digital-to-analog converter; an electronic device, often an integrated circuit, that converts a digital number into a corresponding analog voltage or current.

DAQ Data acquisition. (1) Collecting and measuring electrical signals from sensors, transducers, and test probes or fixtures and inputting them to a computer for processing. (2) Collecting and measuring the same kinds of electrical signals with A/D or DIO devices plugged into a computer, and possibly generating control signals with D/A and/or DIO devices in the same computer.

DC Direct current.

default A default parameter value recorded in the driver; in many setting cases, the default input of a control is a certain value (often 0) that means use the current default setting.

DLL Dynamic link library. A software module in Microsoft Windows containing executable code and data that can be called or used by Windows applications or other DLLs; functions and data in a DLL are loaded and linked at run time when they are referenced by a Windows application or other DLLs.

DMA Direct memory access. A method by which data can be transferred to and from computer memory from and to a device or memory on the bus while the processor does something else; DMA is the fastest method of transferring data to/from computer memory.

DRAM Dynamic RAM.

driver Software that controls a specific hardware device such as an image acquisition device.

dynamic The ratio of the largest signal level a circuit can handle to the range smallest signal level it can handle (usually taken to be the noise level), normally expressed in decibels.

Ε

EEPROM Electrically erasable programmable read-only memory.

ROM that can be erased with an electrical signal and

reprogrammed.

endianness The convention describing the ordering of bytes in

memory or the sequence in which bytes are transmitted.

external A voltage pulse from an external source that triggers an

trigger event such as A/D conversion.

_
_
_

field For an interlaced video signal, a field is half the number of horizontal lines needed to represent a frame of video; the first field of a frame contains all of the odd-numbered lines, and the second field contains all of the even-numbered lines.

FIFO First-in first-out memory buffer. The first data stored is the first data sent to the acceptor; FIFO buffers are used on image acquisition devices to temporarily store incoming data until that data can be retrieved.

flash An ADC whose output code is determined in a single step by ADC a bank of comparators and encoding logic.

frame A complete image; in interlaced formats, a frame is composed of two fields.

front The area of a video signal between the start of the horizontal porch blank and the start of the horizontal sync.

function A set of software instructions executed by a single line of code that may have input and/or output parameters and returns a value when executed.

G

gain Applied value to compensate for discrepancies in the filter for a particular color.

gamma The nonlinear change in the difference between the video signal's brightness level and the voltage level needed to produce that brightness.

genlock Circuitry that aligns the video timing signals by locking together the horizontal, vertical, and color subcarrier frequencies and phases and generates a pixel clock to clock pixel data into memory for display or into another circuit for processing.

Gigabit Describes technologies which transmit Ethernet packets at a Ethernet rate of a gigabit per second.

GigE A camera interface standard developed using the Gigabit Vision Ethernet communication protocol.

grab Performs an acquisition that loops continually on one buffer. You obtain a copy of the acquisition buffer by grabbing a copy to a separate buffer that can be used for analysis.

GUI Graphical user interface. An intuitive, easy-to-use means of communicating information to and from a computer program by means of graphical screen displays; GUIs can resemble the front panels of instruments or other objects associated with a computer program.

Н

hardware The physical components of a computer system, such as

the circuit boards, plug-in boards, chassis, enclosures,

peripherals, cables, and so on.

hardware Separates software API capabilities, such as general

abstraction acquisition and control functions, from hardware-specific

layer information.

HSYNC Horizontal sync signal. The synchronization pulse signal

produced at the beginning of each video scan line that keeps a video monitor's horizontal scan rate in step with

the transmission of each new line.

hue Represents the dominant color of a pixel. The hue function

is a continuous function that covers all the possible colors generated using the R, G, and B primaries. **See also RGB**.

I/O Input/output. The transfer of data to/from a computer

system involving communications channels, operator

interface devices, or data acquisition and control interfaces.

IEEE Institute of Electrical and Electronics Engineers.

INL Integral nonlinearity. A measure, in LSB, of the worst-case

deviation from the ideal A/D or D/A transfer characteristic of

the analog I/O circuitry.

instrument A set of high-level software functions, such as NI-IMAQ, driver

that controls specific plug-in computer boards; instrument

drivers are available in several forms, ranging from a function callable from a programming language to a virtual

instrument (VI) in LabVIEW.

interlaced A video frame composed of two interleaved fields; the

number of lines in a field are half the number of lines in an

interlaced frame.

A page-locked buffer. **See also** page-locked buffer. internal

buffer

interpreter A software utility that executes source code from a high-

level language, such as Java or Basic, by reading one line at a time and executing the specified operation. In contrast, a compiler converts all source code to executable machine

code before execution. Compiled languages give significantly higher performance than interpreted

languages. Examples of compiled languages are C, C++,

and LabVIEW, while Java and Basic are generally

interpreted languages. See also compiler.

interrupt A computer signal indicating that the CPU should suspend

its current task to service a designated activity.

interrupt The relative priority at which a device can interrupt.

level

IRE A relative unit of measure (named for the Institute of Radio

Engineers). 0 IRE corresponds to the blanking level of a video signal, 100 IRE to the white level. Note that for

CIR/PAL video the black level is equal to the blanking level or 0 IRE, while for RS-170/NTSC video, the black level is at

7.5 IRE.

Interrupt request. **See also interrupt**. **IRO**

library A file containing compiled object modules, each comprised

of one or more functions, that can be linked to other object

modules that make use of these functions.

line count The total number of horizontal lines in the picture.

little Describes computers that store bytes of memory by placing endian the least significant byte at the memory location with the

the least significant byte at the memory location with the lowest address, the second least significant byte at the next

memory location, and so on.

LSB Least significant bit.

luminance The brightness information in the video picture. The

luminance signal amplitude varies in proportion to the brightness of the video signal and corresponds exactly to

the monochrome picture.

LUT Lookup table. A selection in Measurement & Automation

Explorer (MAX) for Vision that contains formulas that let you

implement simple imaging operations such as contrast enhancement, data inversion, gamma manipulation, or

other nonlinear transfer functions.

M

MAX

Measurement & Automation Explorer. The National Instruments Windows-based graphical configuration utility you can use to configure NI software and hardware, execute system diagnostics, add new channels and interfaces, and view the devices and instruments you have connected to your computer. MAX is installed on the desktop during the National Instruments driver software installation.

memory **See** buffer.

buffer

memory Continuous blocks of memory that can be accessed quickly window by changing addresses on the local processor.

MSB Most significant bit.

MTBF Mean time between failure.

mux Multiplexer. A switching device with multiple inputs that

selectively connects one of its inputs to its output.

Ν

NI-IMAQ Driver software for National Instruments image

acquisition hardware.

NI-IMAQdx National Instruments driver software for IEEE 1394 and

GigE Vision cameras.

noninterlaced A video frame where all the lines are scanned

sequentially, rather than being divided into two frames

as in an interlaced video frame.

NTSC National Television Standards Committee. The

committee that developed the color video standard used primarily in North America, which uses 525 lines per

frame. See also PAL.

NVRAM Nonvolatile RAM. RAM that is not erased when a device

loses power or is turned off.

0

one- Applies to pulse generation and acquisitions. A one-shot pulse shot or acquisition happens only once.

Ρ Memory page that is marked as non-pagable by the virtual pagefile system. Page-locked buffers remain in physical memory locked and do not cause page faults. buffer PAL Phase Alternation Line. One of the European video color standards; uses 625 lines per frame. See also NTSC. PCI Peripheral Component Interconnect. A high-performance expansion bus architecture originally developed by Intel to replace ISA and EISA; it is achieving widespread acceptance as a standard for PCs and workstations and offers a theoretical maximum transfer rate of 133 Mbytes/s. **PCIe** PCI Express. A high-performance expansion bus architecture originally developed by Intel to replace PCI. PCIe offers a theoretical maximum transfer rate that is dependent upon lane width. A x1 link theoretically provides 250 MB/s in each direction—to and from the device. Once overhead is accounted for, a x1 link can provide approximately 200 MB/s of input capability and 200 MB/s of output capability. Increasing the number of lanes in a link increases maximum throughput by approximately the same factor. PCLK Pixel clock signal. Times the sampling of pixels on a video line. PGIA Programmable gain instrumentation amplifier. picture The ratio of the active pixel region to the active line region; for standard video signals such as RS-170 or CCIR, the fullaspect ratio size picture aspect ratio typically is 4/3 (1.33). Picture element. The smallest division that makes up the pixel video scan line; for display on a computer monitor, a pixel's optimum dimension is square (aspect ratio of 1:1, or the width equal to the height). pixel The ratio between the physical horizontal size and the vertical size of the region covered by the pixel. An acquired aspect pixel should optimally be square, thus the optimal value is ratio 1.0; however, typically it falls between 0.95 and 1.05,

pixel The total number of pixels between two horizontal sync

Divides the incoming horizontal video line into pixels.

depending on camera quality.

pixel

clock

Q

quadlet A 32-bit (four-byte) word.

quadrature An encoding technique for a rotating device where two encoder tracks of information are placed on the device, with the

signals on the tracks offset by 90 degrees from each other. The phase difference indicates the position and direction of

rotation.

R

RAM Random-access memory.

real time A property of an event or system in which data is processed

as it is acquired instead of being accumulated and

processed at a later time.

relative A measure in LSB of the accuracy of an ADC; it includes all

accuracy nonlinearity and quantization errors but does not include offset and gain errors of the circuitry feeding the ADC.

resolution The smallest signal increment that can be detected by a

measurement system; resolution can be expressed in bits, in proportions, or in percent of full scale. For example, a

system has 12-bit resolution, one part in 4,096 resolution,

and 0.0244 percent of full scale.

RGB Red, green, and blue. The three primary colors used to

represent a color picture. An RGB camera is a camera that

delivers three signals, one for each primary.

ribbon A flat cable in which the conductors are side by side.

cable

ring Performs an acquisition that loops continually on a specified

number of buffers.

ROI Region of interest. (1) An area of the image that is

graphically selected from a window displaying the image. This area can be used focus further processing; (2) A

hardware-programmable rectangular portion of the

acquisition window.

ROM Read-only memory.

RS-170 The U.S. standard used for black-and-white television.

RTSI bus Real-Time System Integration Bus. The National

Instruments timing bus that connects image acquisition and DAQ devices directly, by means of connectors on top of the

devices, for precise synchronization of functions.

saturation The richness of a color. A saturation of zero corresponds to no color, that is, a gray pixel. Pink is a red with low

saturation.

scaling Circuitry that scales down the resolution of a video signal.

down circuitry

A type of DMA that allows the DMA controller to reconfigure

gather on-the-fly.

DMA

scatter-

sequence Performs an acquisition that acquires a specified number of

buffers, then stops.

snap Acquires a single frame or field to a buffer.

SRAM Static RAM.

StillColor A post-processing algorithm that allows the acquisition of

high-quality color images generated either by an RGB or composite (NTSC or PAL) camera using a monochrome

video acquisition device.

sync Tells the display where to put a video picture; the horizontal

sync indicates the picture's left-to-right placement and the

vertical sync indicates top-to-bottom placement.

syntax Set of rules to which statements must conform in a

particular programming language.

system RAM installed on a personal computer and used by the

RAM operating system, as contrasted with onboard RAM.

Т

rate

circuitry

timeout Length of time, in milliseconds, that the driver waits for an

image from the camera before returning an error

transfer The rate, measured in bytes/s, at which data is moved

from source to destination after software initialization and setup operations; the maximum rate at which the hardware

can operate.

trigger Any event that causes or starts some form of data capture.

trigger Circuitry that routes, monitors, and drives the external and RTSI bus trigger lines; you can configure each of these

and lines to start or stop acquisition on a rising or falling edge. mapping

TTL Transistor-transistor logic. A digital circuit composed of bipolar transistors wired in a certain manner. A typical medium-speed digital technology. Nominal TTL logic levels

are 0 and 5 V.

U

user Memory buffer created by the user as a destination for the buffer image. In LabVIEW, this is created with the IMAQ Create VI.

UV See YUV.
plane

V

VCO Voltage-controlled oscillator. An oscillator that changes frequency depending on a control signal; used in a PLL to generate a stable pixel clock.

VI Virtual Instrument.

- A combination of hardware and/or software elements, typically used with a PC, that has the functionality of a classic stand-alone instrument
- 2. A LabVIEW software module (VI), which consists of a front panel user interface and a block diagram program.

video A video line consists of a horizontal sync signal, back porch, line active pixel region, and a front porch.

VSYNC Vertical sync signal. The synchronization pulse generated at the beginning of each video field that tells the video monitor when to start a new field.

W

white reference level

The level that defines what is white for a particular video system. **See also** black reference level.

Υ

YUV A representation of a color image used for the coding of NTSC or PAL video signals. The luminance information is called Y, while the chrominance information is represented by two components, U and V, that represent the coordinates in a color plane.

Important Information

Warranty

Copyright

Trademarks

<u>Patents</u>

Warning Regarding Use of NI Products

Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action

accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

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

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

Trademarks

National Instruments, NI, ni.com, and LabVIEW are trademarks of National Instruments Corporation. Refer to the *Terms of Use* section on ni.com/legal for more information about National Instruments trademarks.

FireWire® is the registered trademark of Apple Computer, Inc.

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

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

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

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

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

Patents

For patents covering National Instruments products, refer to the appropriate location: **Help»Patents** in your software, the patents.txt file on your media, or <u>ni.com/patents</u>.

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

- (1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.
- (2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS. SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE. INJURY. OR DEATH. THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES. INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR

APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Technical Support and Professional Services

Visit the following sections of the award-winning National Instruments Web site at ni.com for technical support and professional services:

- <u>Support</u>—Technical support resources at ni.com/support include the following:
 - Self-Help Resources—For answers and solutions, visit
 ni.com/support for software drivers and updates, a searchable
 KnowledgeBase, product manuals, step-by-step
 troubleshooting wizards, thousands of example programs,
 tutorials, application notes, instrument drivers, and so on.
 Registered users also receive access to the NI Discussion
 Forums at ni.com/forums. NI Applications Engineers make
 sure every question submitted online receives an answer.
 - Standard Service Program Membership—This program entitles members to direct access to NI Applications
 Engineers via phone and email for one-to-one technical support, as well as exclusive access to on demand training modules via the <u>Services Resource Center</u>. NI offers complementary membership for a full year after purchase, after which you may renew to continue your benefits.

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

- <u>Training and Certification</u>—Visit ni.com/training for self-paced training, eLearning virtual classrooms, interactive CDs, and Certification program information. You also can register for instructor-led, hands-on courses at locations around the world.
- System Integration—If you have time constraints, limited in-house technical resources, or other project challenges, National Instruments Alliance Partner members can help. To learn more, call your local NI office or visit ni.com/alliance.

If you searched ni.com and could not find the answers you need, contact your <u>local office</u> or NI corporate headquarters. You also can visit the <u>Worldwide Offices</u> section of ni.com/niglobal to access the branch office Web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

Branch Offices

Office	Telephone Number
Australia	1800 300 800
Austria	43 662 457990-0
Belgium	32 (0) 2 757 0020
Brazil	55 11 3262 3599
Canada	800 433 3488
China	86 21 5050 9800
Czech Republic	420 224 235 774
Denmark	45 45 76 26 00
Finland	358 (0) 9 725 72511
France	33 (0) 1 57 66 24 24
Germany	49 89 7413130
India	91 80 41190000
Israel	972 0 3 6393737
Italy	39 02 41309277
Japan	0120-527196 / 81 3 5472 2970
Korea	82 02 3451 3400
Lebanon	961 (0) 1 33 28 28
Malaysia	1800 887710
Mexico	01 800 010 0793
Netherlands	31 (0) 348 433 466
New Zealand	0800 553 322
Norway	47 (0) 66 90 76 60
Poland	48 22 3390150
Portugal	351 210 311 210
Russia	7 495 783 6851
Singapore	1800 226 5886
Slovenia	386 3 425 42 00

South Africa	27 0 11 805 8197
Spain	34 91 640 0085
Sweden	46 (0) 8 587 895 00
Switzerland	41 56 2005151
Taiwan	886 02 2377 2222
Thailand	662 278 6777
Turkey	90 212 279 3031
United Kingdom	44 (0) 1635 523545
United States (Corporate)	512 683 0100