

# IMAQ™

---

## NI-IMAQ™ VI Reference Manual

## **Worldwide Technical Support and Product Information**

[www.natinst.com](http://www.natinst.com)

## **National Instruments Corporate Headquarters**

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 794 0100

## **Worldwide Offices**

Australia 03 9879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20, Brazil 011 284 5011,  
Canada (Ontario) 905 785 0085, Canada (Québec) 514 694 8521, China 0755 3904939, Denmark 45 76 26 00,  
Finland 09 725 725 11, France 01 48 14 24 24, Germany 089 741 31 30, Hong Kong 2645 3186,  
India 91805275406, Israel 03 6120092, Italy 02 413091, Japan 03 5472 2970, Korea 02 596 7456,  
Mexico (D.F.) 5 280 7625, Mexico (Monterrey) 8 357 7695, Netherlands 0348 433466, Norway 32 27 73 00,  
Singapore 2265886, Spain (Madrid) 91 640 0085, Spain (Barcelona) 93 582 0251, Sweden 08 587 895 00,  
Switzerland 056 200 51 51, Taiwan 02 2377 1200, United Kingdom 01635 523545

For further support information, see the *Technical Support Resources* appendix. To comment on the documentation, send e-mail to [techpubs@natinst.com](mailto:techpubs@natinst.com).

© Copyright 1997, 1999 National Instruments Corporation. All rights reserved.

# Important Information

---

## 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 THEREOF 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.

## Trademarks

BridgeVIEW™, IMAQ™, LabVIEW™, NI-IMAQ™, RTSI™, and StillColor™ are trademarks of National Instruments Corporation.

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

## WARNING REGARDING MEDICAL AND CLINICAL USE OF NATIONAL INSTRUMENTS PRODUCTS

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. Applications of National Instruments products involving medical or clinical treatment can create a potential for death or bodily injury caused by product failure, or by errors on the part of the user or application designer. 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.

# Contents

---

## About This Manual

Conventions Used in This Manual.....	ix
--------------------------------------	----

## Chapter 1

### How to Use NI-IMAQ VIs

Introduction .....	1-1
Location of NI-IMAQ Examples .....	1-2
Location of the NI-IMAQ VIs .....	1-2
Common NI-IMAQ VI Parameters .....	1-4
Error Handling .....	1-4
Buffer Management .....	1-6
NI-IMAQ Acquisition Types.....	1-6
Snap .....	1-7
Grab .....	1-7
Sequence.....	1-8
Ring .....	1-9
Acquisition VIs .....	1-9
High-Level.....	1-9
Low-Level .....	1-9
StillColor Acquisition .....	1-10
Composite Snap.....	1-10
RGB Snap.....	1-12
Triggering .....	1-12
Image Display .....	1-13
Camera Attributes .....	1-15

## Chapter 2

### NI-IMAQ VIs

NI-IMAQ VIs .....	2-1
High-Level VIs .....	2-2
IMAQ Snap .....	2-3
IMAQ Grab Setup .....	2-4
IMAQ Grab Acquire.....	2-5
IMAQ Sequence .....	2-6
IMAQ Init.....	2-8
IMAQ Close .....	2-9
IMAQ Set Camera Attribute .....	2-10
IMAQ Get Camera Attribute.....	2-11

Signal I/O VIs.....	2-12
IMAQ Configure Trigger.....	2-13
IMAQ Trigger Drive.....	2-15
IMAQ Trigger Read.....	2-17
IMAQ Generate Pulse.....	2-18
IMAQ Wait Signal.....	2-21
IMAQ Occurrence Config.....	2-23
Low-Level VIs.....	2-25
IMAQ Configure List.....	2-26
IMAQ Configure Buffer.....	2-27
IMAQ Start.....	2-28
IMAQ Get Buffer.....	2-29
IMAQ Extract Buffer.....	2-30
IMAQ Copy.....	2-31
IMAQ Status.....	2-32
IMAQ Stop.....	2-33
IMAQ Set User LUT.....	2-34
IMAQ Vision VIs.....	2-35
IMAQ Create.....	2-36
IMAQ Dispose.....	2-37
IMAQ ImageToArray.....	2-38
IMAQ ColorImageToArray.....	2-39

## Chapter 3

### IMAQ Property Node

IMAQ Property Node.....	3-1
Analog Properties.....	3-2
Board Information Properties.....	3-2
Color Properties.....	3-4
Image Properties.....	3-8
Session Information Properties.....	3-11
Status Information Properties.....	3-11
StillColor Properties.....	3-12

## Appendix A

### Color Image Representation

## Appendix B

### Error Codes

## Appendix C

### Technical Support Resources

#### Glossary

#### Index

#### Figures

Figure 1-1.	LabVIEW Functions Palette with IMAQ Palette .....	1-2
Figure 1-2.	BridgeVIEW Functions Palette with IMAQ Palette .....	1-3
Figure 1-3.	Error Clusters .....	1-5
Figure 1-4.	Error Checking using the Simple Error Handler VI.....	1-5
Figure 1-5.	Simple Error Handler Dialog Box.....	1-5
Figure 1-6.	IMAQ Create.....	1-6
Figure 1-7.	IMAQ Dispose .....	1-6
Figure 1-8.	Acquiring an Image Using Snap .....	1-7
Figure 1-9.	Acquiring Images Using Grab.....	1-8
Figure 1-10.	Acquiring Images Using Sequence .....	1-8
Figure 1-11.	Acquiring a StillColor Composite Image.....	1-11
Figure 1-12.	Acquiring an RGB Image.....	1-12
Figure 1-13.	IMAQ Triggering .....	1-13
Figure 1-14.	Displaying an Image Using IMAQ WindDraw.....	1-13
Figure 1-15.	Intensity Graph for Image Display .....	1-14
Figure 1-16.	Displaying an Image Using an Intensity Graph .....	1-14
Figure 1-17.	Using a Picture Control to Display an RGB Image.....	1-15
Figure 1-18.	IMAQ Set Camera Attribute .....	1-16
Figure 1-19.	Using the IMAQ Get Camera Attribute .....	1-16
Figure 2-1.	NI-IMAQ High-Level and Attribute VI Palette .....	2-2
Figure 2-2.	NI-IMAQ Event VI Palette .....	2-12
Figure 2-3.	NI-IMAQ Low-Level VI Palette .....	2-25
Figure 2-4.	IMAQ Vision VI Palette.....	2-35
Figure 3-1.	IMAQ Property Node.....	3-1

#### Tables

Table A-1.	StillColor Image Representations.....	A-1
Table B-1.	NI-IMAQ VI Error Codes .....	B-2

# About This Manual


---

This manual describes the features, functions, and operation of the image acquisition (IMAQ) virtual instruments (VIs) for LabVIEW. To use this manual effectively, you must be familiar with image processing, LabVIEW or BridgeVIEW, and your IMAQ hardware device.

## Conventions Used in This Manual

---

The following conventions are used in this manual:

- [ ]** 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 note, which alerts you to important information.
- bold** Bold text denotes items that you must select or click on in the software, such as menu items and dialog box options. Bold text also denotes parameter names.
- italic* Italic text denotes variables, emphasis, a cross reference, or an introduction to a key concept. This font 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, and code excerpts.

---

# How to Use NI-IMAQ VIs

This chapter describes how to use National Instruments G programming and application software, such as LabVIEW, BridgeVIEW, and IMAQ Vision, with your IMAQ hardware and NI-IMAQ VIs.

## Introduction

---

LabVIEW and BridgeVIEW feature interactive graphics, a state-of-the-art user interface, and a powerful graphical programming language, G. The G NI-IMAQ VI Library, a series of virtual instruments (VIs) for using LabVIEW and BridgeVIEW with your IMAQ device, is included with your NI-IMAQ software.

IMAQ Vision for G is an image processing and analysis library that consists of more than 400 VIs for use with LabVIEW and BridgeVIEW. If you have not purchased the IMAQ Vision image processing and analysis libraries, you can use the four IMAQ Vision VIs included with your NI-IMAQ software. If you use these basic functions, you can later upgrade your programs to use IMAQ Vision without any changes to your image acquisition VIs.

Before you start building your IMAQ application, you should know the following basic G IMAQ concepts:

- Location of the NI-IMAQ examples
- Location of the NI-IMAQ VIs in LabVIEW or BridgeVIEW
- Common NI-IMAQ VI parameters
- Error handling
- Buffer management
- NI-IMAQ acquisition types
- Acquisition VIs
- StillColor acquisition
- Triggering
- Image display
- Camera attributes



## Location of NI-IMAQ Examples

The NI-IMAQ examples for G illustrate some common applications used with NI-IMAQ in LabVIEW and BridgeVIEW. You can find these examples in the `labview\examples\imaq` directory for LabVIEW and `bridgeview\examples\g_examples\imaq` for BridgeVIEW. For a brief description of any example, open the example VI and choose **Windows»Show VI Info** for a text description of the example.

## Location of the NI-IMAQ VIs

You can find the NI-IMAQ VIs in the **Functions** palette from your block diagram in LabVIEW and BridgeVIEW. Select the IMAQ icon near the bottom of the **Functions** palette, as shown in Figures 1-1 and 1-2, to pop up the **Image Acquisition** palette.

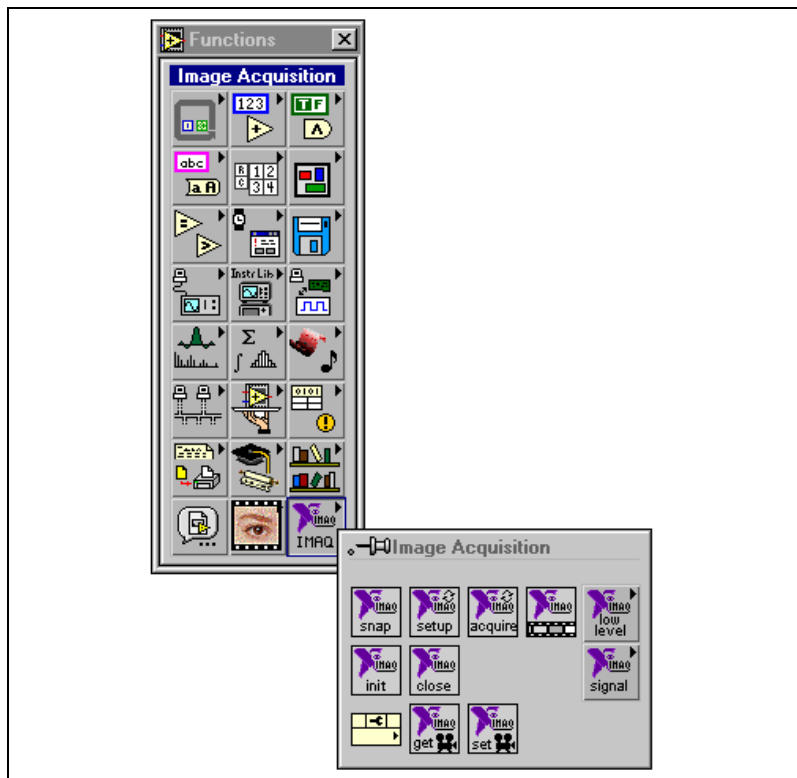
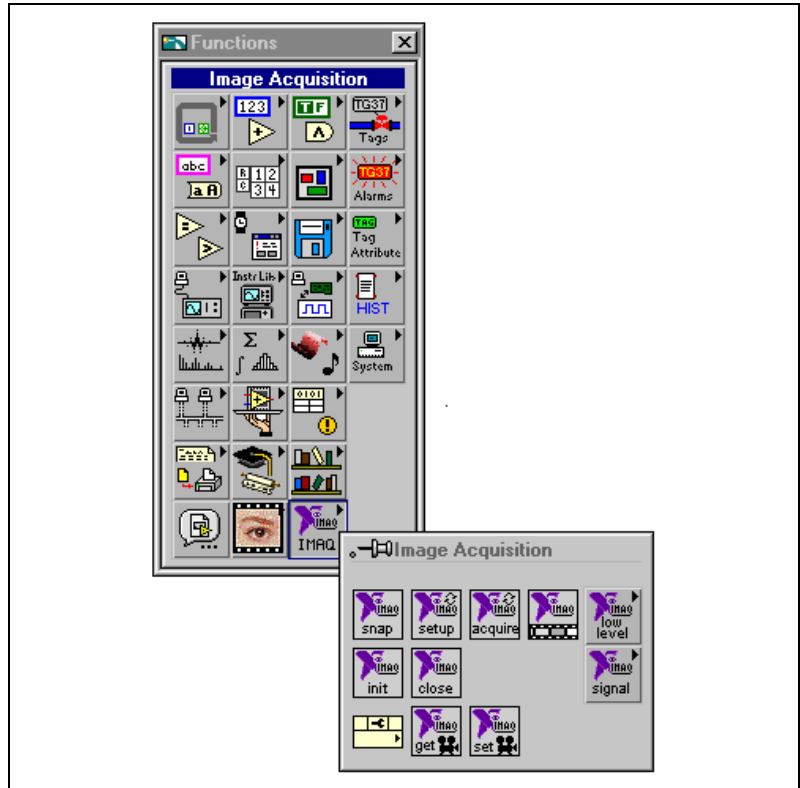


Figure 1-1. LabVIEW Functions Palette with IMAQ Palette



**Figure 1-2.** BridgeVIEW Functions Palette with IMAQ Palette

The most commonly used VIs are on the **Image Acquisition** palette. VIs are provided for basic acquisition and changing attributes. See the [High-Level VIs](#) section in Chapter 2, *NI-IMAQ VIs*, for more information on these VIs.

The **Image Acquisition»IMAQ Low Level** palette contains VIs for more advanced applications. See the [Low-Level VIs](#) section in Chapter 2, *NI-IMAQ VIs*, for more information on the **Low-Level VI** palette.

The **Image Acquisition»IMAQ Signal I/O** palette contains VIs for using triggers and pulse generation with IMAQ devices. See the [Signal I/O VIs](#) section in Chapter 2, *NI-IMAQ VIs*, for more information on the **Signal I/O VI** palette.

## Common NI-IMAQ VI Parameters

---

**IMAQ Session** is a unique identifier that specifies the Interface file used for the acquisition. It is produced by the IMAQ Init VI and used as an input to all other NI-IMAQ VIs. **IMAQ Session Out**, which is identical to **IMAQ Session**, is used to simplify dataflow programming and is similar to the duplicate file sessions provided by the file I/O functions. The high-level acquisition VIs—IMAQ Snap, IMAQ Grab Setup, and IMAQ Sequence—require you to wire **IMAQ Session In** only if you are using an interface other than the default `img0`, you are using multiple boards, or you need to set IMAQ properties before the acquisition.

Many acquisition VIs require that you supply an image buffer to receive the captured image. You can create this image buffer with the IMAQ Create VI. Consult the *Buffer Management* section of this chapter for more information. The input that receives the image buffer is **Image in**. The **Image out** output returns the captured image.

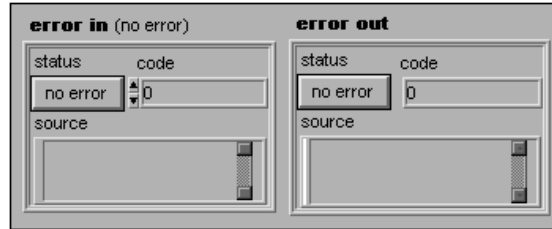
The acquisition VIs use the **Region of Interest** input to specify a rectangular portion of an image frame to be captured, which you can use to reduce the size of the image you want to capture. **Region of Interest** is an array of four elements with the elements defined as Left, Top, Right, Bottom. The width [Right-Left] must be a multiple of eight. If **Region of Interest** is not wired, the entire image acquisition window is captured. You configure the default acquisition window using the Measurement & Automation Explorer for IMAQ.

The acquisition VIs use the **Step x** and **Step y** inputs to specify a horizontal and vertical sampling step. The sampling step causes a reduction in spatial resolution; however, the entire acquisition window is transferred. Using a sampling step allows you to reduce the size of the image by trading spatial resolution for field of view.

## Error Handling

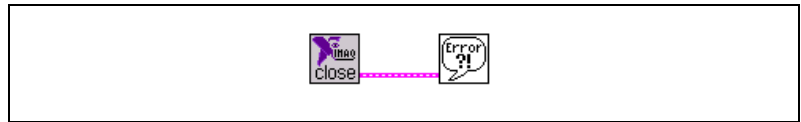
---

Every NI-IMAQ VI contains an **error in** input cluster and an **error out** output cluster, as shown in Figure 1-3. The clusters contain a Boolean that indicates whether an error occurred, the code for the error, and the source or the 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.



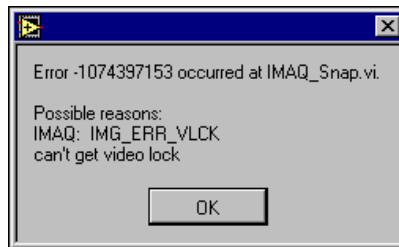
**Figure 1-3.** Error Clusters

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. Figure 1-4 shows how to wire an NI-IMAQ VI to the Simple Error Handler VI.



**Figure 1-4.** Error Checking using the Simple Error Handler VI

Figure 1-5 shows an example of the dialog box Simple Error Handler displays when an error occurs.



**Figure 1-5.** Simple Error Handler Dialog Box

For more information on error handling, see Appendix B, [Error Codes](#).

## Buffer Management

IMAQ Create and IMAQ Dispose manage image buffers in LabVIEW and BridgeVIEW. IMAQ Create, shown in Figure 1-6, allocates an image buffer. **Image Name** is a label for the buffer created. Each buffer must have a unique name. **ImageType** specifies the type of image being created. Use **8 bits** for 8-bit monochrome images, **16 bits** for 10-, 12-, and 14-bit monochrome images, **RGB** for RGB color images, and **HSL** for HSL color images.

**New Image** contains pointer information to the buffer, which is initially empty. When you wire **New Image** to the **Image in** input of an image acquisition VI, the image acquisition VI allocates the correct amount of memory for the acquisition. If you are going to process the image, you might need to wire to **Border Size**. **Border Size** is the width in pixels created around an image. Some image processing functions, such as labeling or morphology, require a border.

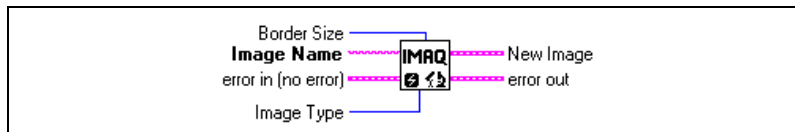


Figure 1-6. IMAQ Create

IMAQ Dispose, shown in Figure 1-7, frees the memory allocated for the image buffer. Call this VI only after the image is no longer required for processing.

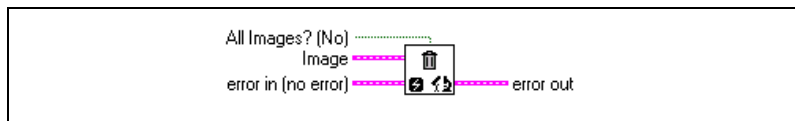


Figure 1-7. IMAQ Dispose

## NI-IMAQ Acquisition Types

Four NI-IMAQ image acquisition types are available in LabVIEW and BridgeVIEW—snap, grab, sequence, and ring. The following sections describe each acquisition type and give examples.

## Snap

A snap acquires a single image into a memory buffer. Use this acquisition mode to acquire a single frame or field to a buffer. When you invoke a snap, it initializes the board and acquires the next incoming video frame (or field) to a buffer. A snap is appropriate for low-speed or single-capture applications.

Use the IMAQ Snap VI for snap applications. Figure 1-8 shows a simplified block diagram for using IMAQ Snap.



Figure 1-8. Acquiring an Image Using Snap

## Grab

A grab is a continuous, high-speed acquisition of data to a single buffer in host memory. This function performs an acquisition that loops continually on one buffer. You can get a copy of the acquisition buffer by grabbing a copy to a G image buffer.

You must use two VIs, IMAQ Grab Setup and IMAQ Grab Acquire, for a grab acquisition in G. IMAQ Grab Setup, which you call only once, initializes the acquisition and starts capturing the image to an internal software buffer. IMAQ Grab Acquire, which you can call multiple times, copies the image currently stored in the internal buffer to a G image buffer. The **Immediate?** input to IMAQ Grab Acquire determines if the copy takes place immediately or if it waits for the next vertical blank. If **Immediate?** is FALSE, IMAQ Grab waits for the next vertical blank signal and then transfers the image from the internal buffer to the G image buffer. If **Immediate?** is TRUE, IMAQ Grab immediately transfers the image from the internal buffer to the G image buffer, which could result in portions of the image transferred being acquired at different times. A typical application for an immediate transfer is the acquisition of images of stationary objects. After the program is finished copying images, call IMAQ Close once to shut down the acquisition.

Figure 1-9 shows a simplified block diagram for using IMAQ Grab Setup and IMAQ Grab Acquire. In this example, you perform an immediate copy by wiring a TRUE to the **Immediate?** input.

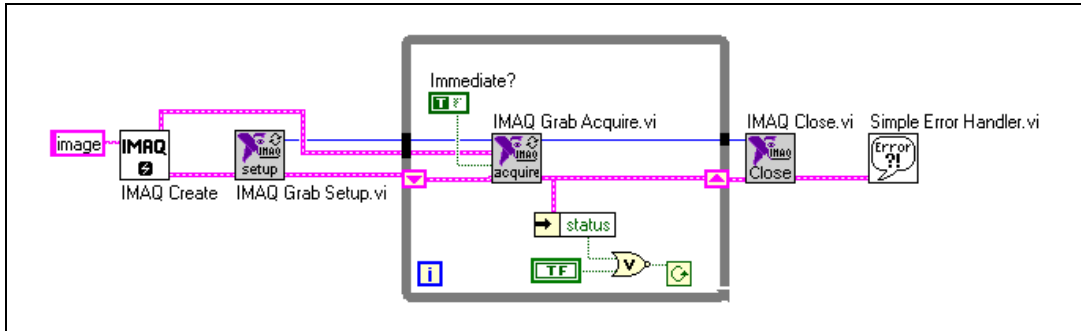


Figure 1-9. Acquiring Images Using Grab

## Sequence

A sequence initiates a variable-length and variable-delay transfer to multiple buffers. A sequence is appropriate for applications that process multiple images. You can configure a sequence to acquire every frame or skip a variable number of frames between each image.

The VI you use for sequence applications is IMAQ Sequence. IMAQ Sequence starts, acquires, and releases a sequence acquisition. The input **Skip Table** is an array containing the number of frames to skip between images. IMAQ Sequence does not return until the entire sequence is acquired.

Figure 1-10 shows a simplified block diagram for using IMAQ Sequence. IMAQ Create is used inside a For Loop to create an array of images for the **Images in** input to IMAQ Sequence. To Decimal and Concatenate Strings illustrate a useful method to create a unique name for each image in the array.

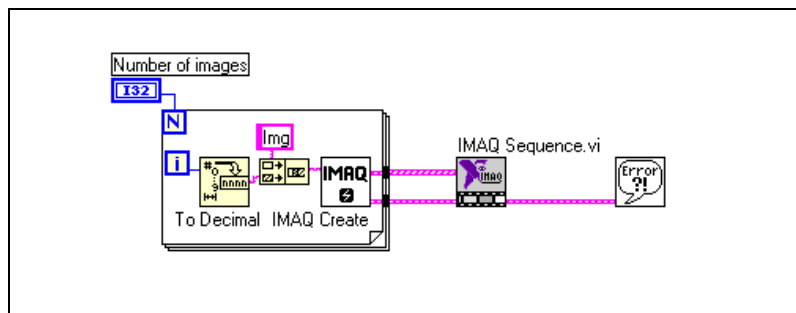


Figure 1-10. Acquiring Images Using Sequence

## Ring

A ring initiates a continuous high-speed acquisition to multiple buffers. A ring is appropriate for high-speed applications where you need to perform processing on every image. You must use multiple buffers because processing times may vary, depending on other applications and processing results.

You can configure a ring to acquire every frame or to skip a fixed number of frames between acquisitions. In G, you must use the NI-IMAQ low-level VIs to perform a ring.

## Acquisition VIs

---

Two acquisition VI types are available in LabVIEW and BridgeVIEW—high-level and low-level.

### High-Level

You can use the high-level acquisition VIs for basic image acquisition applications. VIs are included for snap, grab, and sequence as described in the *NI-IMAQ Acquisition Types* section. Examples of using the high-level acquisition VIs are included in the `examples\imaq\imaq_high_level.llb` file.

### Low-Level

You can use the low-level acquisition VIs for more advanced image acquisition applications, including ring acquisitions and acquisitions to onboard memory. The low-level VIs consist of VIs to configure an acquisition, start an acquisition, retrieve the acquired images, and stop an acquisition. You can use these VIs in conjunction with the event VIs to construct advanced IMAQ applications.

Follow these general steps to perform a low-level acquisition:

1. Call IMAQ Init to initialize the board and create an **IMAQ Session**.
2. Configure the acquisition with IMAQ Configure List and IMAQ Configure Buffer. IMAQ Configure List configures a buffer list to be used in an acquisition. The buffer list contains a specific number of buffers that will contain the acquired images. The buffers can be stored either in system memory or in onboard memory (for boards with onboard memory such as the IMAQ PCI-1424).



3. Call IMAQ Configure Buffer once for each buffer in the buffer list. The buffer contains the channel from which to acquire and how many frames to skip before acquiring into the buffer.
4. After configuring the buffer list and individual buffers, call IMAQ Start to start the acquisition asynchronously. IMAQ Start returns immediately after the acquisition has started.
5. Access the acquired images using either IMAQ Get Buffer or IMAQ Extract Buffer. IMAQ Get Buffer returns acquired images from the buffer list and is normally used for snap and sequence acquisitions. IMAQ Get Buffer waits until the requested buffer has been acquired to return the image. You can also use this VI to return all images in the buffer list. IMAQ Get Buffer can retrieve images from a continuous acquisition only if the acquisition has been stopped.

IMAQ Extract Buffer extracts a buffer from a continuous acquisition and allows for the examination of a buffer during acquisition. The buffer is removed from the acquisition and new data will not be written into the buffer until this VI is called again. Use IMAQ Extract Buffer in ring acquisitions when you must process images during the acquisition. IMAQ Copy returns a copy of an acquired image. IMAQ Copy allows you to create a copy of any buffer at any time during the acquisition.

6. After an acquisition, release the resources associated with the acquisition using IMAQ Close. IMAQ Close will also stop the acquisition if one is in progress. If you would like to stop the acquisition without releasing the resources (such as the image buffers), use IMAQ Stop.

Examples of the low-level acquisition VIs are included in `examples/imaq/IMAQ Low Level.llb`.

## StillColor Acquisition

---

You can use an IMAQ PCI/PXI-1408 to acquire color images from either a color composite or RGB camera.

### Composite Snap

A StillColor composite snap acquires a single color image into a memory buffer from a color composite camera. You can use a StillColor composite snap for high-quality color images of still or very slowly moving objects. For more information on StillColor, refer to Appendix A, *Color Basics and StillColor*, of the *NI-IMAQ User Manual*.

To perform a StillColor composite snap, use the IMAQ Snap VI for acquisition. Figure 1-11 shows a simplified block diagram for performing a StillColor snap. Perform the following steps to acquire an image using a StillColor composite snap:

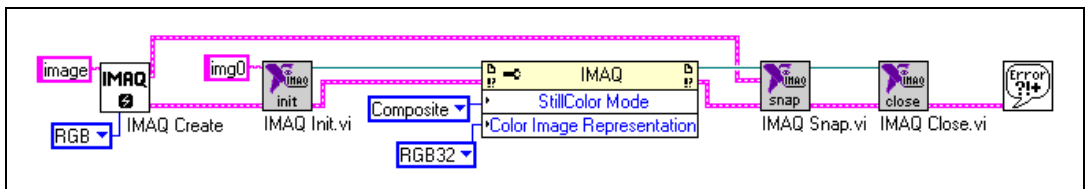
1. Use the IMAQ Init VI to generate an **IMAQ Session** for StillColor acquisition.
2. Use the IMAQ Property Node to set up the IMAQ device for StillColor.
3. Set the **StillColor»Color Mode** and **Color»Color Image Representation** properties. The input to **Color Mode** is a ring control.
  - a. To create the ring control, pop up on the **Set StillColor Mode** input and choose **Create Constant**.
  - b. Click on the created constant with the Operating Tool and select **Composite**. The input to **Color Image Representation** is also a ring control. This input specifies the type of image data to be returned by the IMAQ Snap VI, which is RGB 32-bit in this example.

The IMAQ Create VI must create an image buffer that corresponds to the image type specified by **Color Image Representation**. Refer to Appendix A, [Color Image Representation](#) for a list of image representations and the corresponding image types.

4. Finally, use IMAQ Snap VI to acquire the image from the IMAQ device.



**Note** For more information about using the IMAQ Property Node, see Chapter 3, [IMAQ Property Node](#).



**Figure 1-11.** Acquiring a StillColor Composite Image

## RGB Snap

An RGB snap acquires a single color image into a memory buffer from an RGB camera. You can use an RGB snap for acquiring high-quality color images of still or very slowly moving objects. For more information on StillColor RGB refer to Appendix A, *Color Basics and StillColor*, in the *NI-IMAQ User Manual*.

To perform an RGB snap, use the IMAQ Snap VI for acquisition. Figure 1-12 shows a simplified block diagram for performing an RGB snap. An RGB snap is identical to a composite snap except that the RGB snap's **Color Mode** property should be set to RGB.

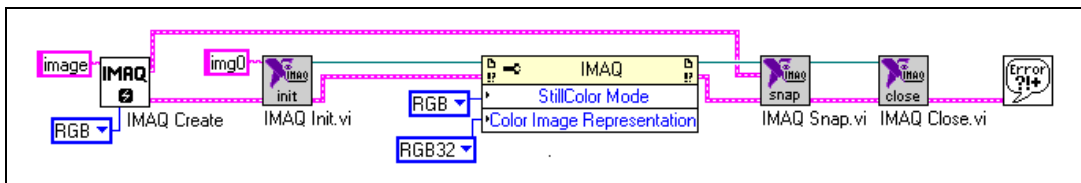


Figure 1-12. Acquiring an RGB Image

## Triggering

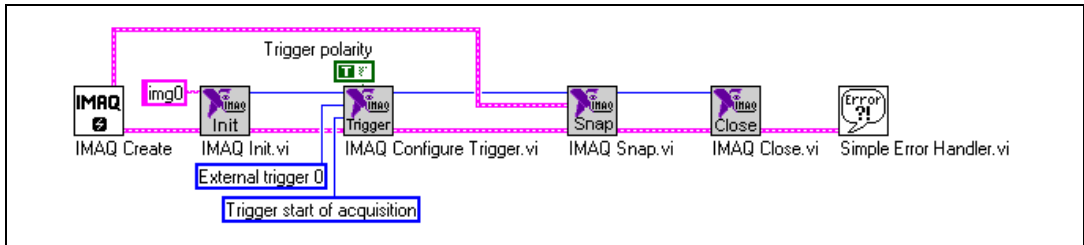
Often you may need to link or coordinate a vision action or function with events external to the computer, such as the receipt of a strobe pulse for lighting or a pulse from an infrared detector that indicates the position of an item on an assembly line. A trigger on an IMAQ device can be any TTL-level signal. All of the trigger lines are fully bidirectional so that the device can generate or receive the triggers on any line. The IMAQ PCI/PXI-1407 and IMAQ PCI/PXI-1411 have one external trigger line. The IMAQ PCI/PXI-1408, PCI/PXI-1422, and PCI-1424 have four external trigger lines and seven Real-Time System Integration (RTSI) bus lines for general purpose use. Use the RTSI triggers to coordinate your IMAQ device with other National Instruments boards, such as data acquisition (DAQ) boards.



**Note** You can use only four of the seven RTSI triggers at once.

Use IMAQ Configure Trigger to configure the trigger conditions for an acquisition. You must call IMAQ Configure Trigger before the acquisition VI. The **Trigger line** input specifies which external or RTSI trigger receives the incoming trigger signal. Each trigger line has a programmable polarity that is specified with **Trigger polarity**.

**Frame timeout** specifies the amount of time to wait for the trigger. Figure 1-13 shows how to use IMAQ Configure Trigger to perform a snap acquisition based on a trigger.

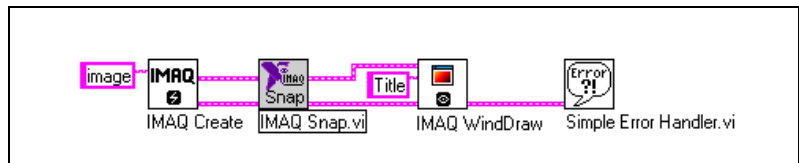


**Figure 1-13.** IMAQ Triggering

## Image Display

Many image acquisition applications require that one or more images be displayed. Three options are available for displaying images in LabVIEW and BridgeVIEW.

If you have IMAQ Vision for G, the image processing and analysis software for LabVIEW and BridgeVIEW, you can use IMAQ WindDraw. IMAQ WindDraw (**IMAQ Vision»Display(basics)**) displays an image in an image window. Figure 1-14 illustrates using IMAQ WindDraw to display an image acquired using IMAQ Snap. Images can be displayed in the same way using any acquisition type. For more information on the display capabilities of IMAQ Vision, consult the *IMAQ Vision for G Reference Manual*.



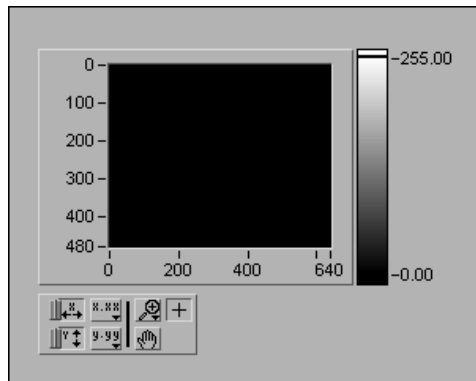
**Figure 1-14.** Displaying an Image Using IMAQ WindDraw

If you do not have IMAQ Vision, you can display an image on a G Intensity Graph for 8-bit and 16-bit monochrome images or on a picture control for RGB images.

Before you can properly display an image on an Intensity Graph, you need to make some minor changes to the default properties of the Intensity Graph. Perform the following steps to modify the properties:

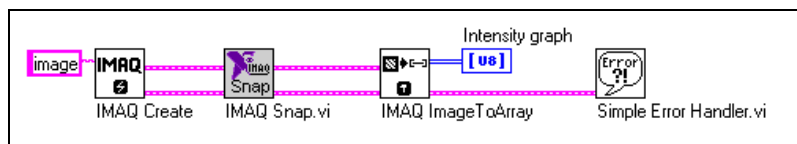
1. Place the Intensity Graph on the front panel, pop up on the graph, and choose **Transpose Array**.
2. Create the correct grayscale color palette by popping up on the marker labeled 50 on the color ramp and choosing **Delete Marker**. Also, change the maximum value on the color palette from 100 to the maximum pixel value in your image—255 for 8-bit images, 1,023 for 10-bit images, and 4,095 for 12-bit images.
3. Change the y-axis so that it is inverted. You might also need to change the ranges of the x- and y-axes to match the width and height of the image.

Your intensity graph now should appear similar to the image shown in Figure 1-15. For more information on the Intensity Graph, consult the *LabVIEW User Manual* or the *G Programming Reference Manual*.



**Figure 1-15.** Intensity Graph for Image Display

Use the IMAQ ImageToArray VI to copy an image from an image buffer into a G array. Then you can wire this array directly to an Intensity Graph for display. Figure 1-16 illustrates using an Intensity Graph to display an image acquired using IMAQ Snap.



**Figure 1-16.** Displaying an Image Using an Intensity Graph

To display an RGB image on a picture control, place the picture control on the front panel of your VI. Use the IMAQ ColorImageToArray VI to copy an image from an image buffer into a G array. Then you can wire this array to the Draw True-Color Pixmap VI. Wire the new image output from Draw True-Color Pixmap to the picture control indicator. For more information on the picture control, consult the LabVIEW online reference. Figure 1-17 illustrates using a picture control to display an RGB image acquired with IMAQ Snap.

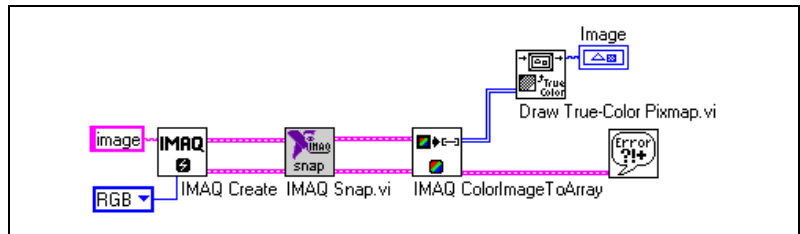


Figure 1-17. Using a Picture Control to Display an RGB Image

## Camera Attributes

The camera attribute VIs allow you to control camera functions, such as integration time and pixel binning, directly from LabVIEW or BridgeVIEW. These camera attributes are camera-specific and can also be set in the Measurement & Automation Explorer for IMAQ on the **Advanced** tab in the Properties dialog box for the IMAQ PCI-1424 and IMAQ PCI/PXI-1422. You can find information about specific attributes for your camera in the <my camera>.txt file, which is in the camera info directory in your ni-imaq directory. For more information about your camera's attributes and their uses, please consult your camera documentation.



**Note** Currently only the IMAQ PCI-1424 and IMAQ PCI/PXI-1422 support camera attributes.

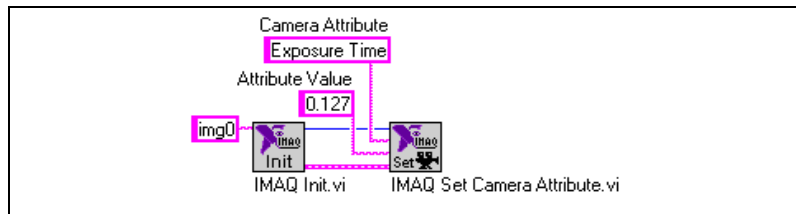
Use the Set Camera Attribute VI to set the value of a camera attribute. The camera attribute file mentioned above lists all attributes for the camera where each attribute description contains four fields: **Attribute Name**, **Description**, **Data Type**, and **Possible Values**. The **Attribute Name** field contains the name of the attribute in quotes and should be wired to the **Camera Attribute** input on Set Camera Attribute VI. The **Data Type** field contains the data type of the attribute which can either be **String**, **Integer**, or **Float**. **String** indicates that there is a list of possible values which are

listed in **Possible Values** in quotes. To set the value of a string attribute, wire the desired string value to **Attribute Value** on Set Camera Attribute.



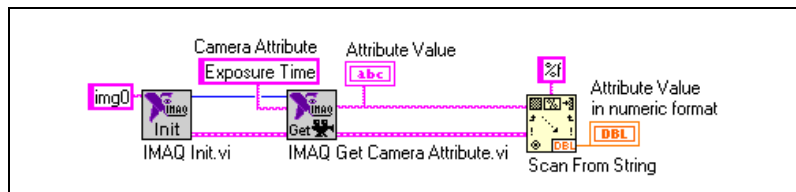
**Note** The spelling and syntax of the **Attribute Name** and string values must match the camera attribute file exactly. (You should not enter the quotes into the string control or constant in G.)

A data type of **Integer** indicates that the string wired to **Attribute Value** will be converted to an integer. **Float** indicates that the string wired to **Attribute Value** will be converted to a floating point number. The valid numeric values for integer and float data types are listed in **Possible Values**. Use **Format into String (String subpalette)** to convert numerics into strings for use with the IMAQ Set Camera Attribute VI. Figure 1-18 shows how to use IMAQ Set Camera Attribute to set the value of a float camera attribute.



**Figure 1-18.** IMAQ Set Camera Attribute

Use the IMAQ Get Camera Attribute VI to get the value of a camera attribute. Use the camera attribute file described above to find information about the attributes for your camera. All camera attributes are returned in string format. If the data type of the attribute is integer or float, use the **Scan from String (String subpalette)** function to convert the string into a numeric. Figure 1-19 shows how to use IMAQ Get Camera Attribute with **Scan from String** to get the value of a float camera attribute.



**Figure 1-19.** Using the IMAQ Get Camera Attribute

---

# NI-IMAQ VIs

This chapter describes the NI-IMAQ and IMAQ Vision VIs included with your NI-IMAQ software.

## NI-IMAQ VIs

---

The NI-IMAQ VIs gives you the basic functions to:

- Load information about boards and cameras from a configuration file
- Select a video channel
- Adjust the analog parameters
- Start or stop an acquisition
- Perform a StillColor or RGB acquisition
- Transfer an image from your IMAQ device memory to an IMAQ Vision image buffer
- Monitor and control the IMAQ device trigger lines

The VIs described in this document are arranged according to type—High-Level and Attribute, Signal I/O, Low-Level, and IMAQ Vision—and then in palette order.

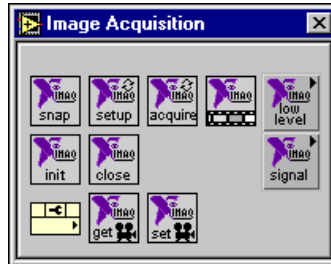


## High-Level VIs

---

You can use high-level VIs to set up your IMAQ system and acquire images.

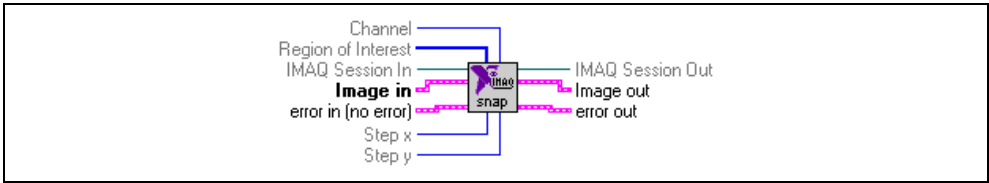
When you choose the **Function»Image Acquisition** palette, you will see the high-level VI palette shown in Figure 2-1.



**Figure 2-1.** NI-IMAQ High-Level and Attribute VI Palette

The high-level NI-IMAQ VIs are sufficient for many applications. These VIs allow you to acquire images, open and close an interface, and get and set camera attributes.

# IMAQ Snap



Acquires a single image into **Image out**. If necessary, this VI performs a system initialization using IMAQ Init before the acquisition.

When you invoke a snap, it initializes the board and acquires the next incoming video frame (or field) to a buffer. A snap is appropriate for low-speed or single-capture applications where ease of programming is essential.



**Channel** specifies from which channel on the IMAQ device to acquire.



**Region of Interest** specifies a rectangular portion of the image to be captured. **Region of Interest** is defined by an array of four elements [Left, Top, Right, Bottom]. You must set the width [Right-Left] to a multiple of eight. If **Region of Interest** is not connected or empty, the entire acquisition window is captured.



**IMAQ Session In** is a unique identifier that identifies the Interface file. If you are using the default interface, `img0`, no connection is needed.



**Image In** is the reference to the image that will receive the captured pixel data.



**Step x** is a horizontal sampling step or horizontal reduction factor. If it is set to its default value of 1, each column of the image is transferred. If **Step x** is set to another value  $n$ , only one column every  $n$  columns is transferred. **Step x** only accepts values of 1, 2, 4, or 8.



**Step y** is a vertical sampling step or vertical reduction factor. If it is set to its default value of 1, each line of the image is transferred. If **Step y** is set to another value  $n$ , only one line every  $n$  lines is transferred. **Step y** only accepts values of 1, 2, 4, or 8.



**IMAQ Session Out** has the same value as **IMAQ Session In**.

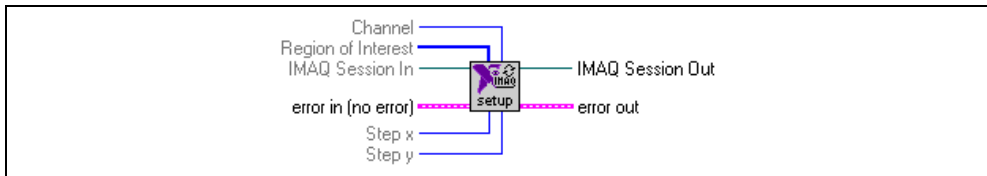


**Image Out** is the reference to the captured image.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Grab Setup



Starts a grab acquisition. This function performs an acquisition that loops continually on one buffer. Use the grab function for high-speed image acquisition. Use IMAQ Grab Acquire to copy an image out of the buffer.

If necessary, this VI performs a system initialization using IMAQ Init.



**Channel** specifies from which channel on the IMAQ device to acquire.



**Region of Interest** specifies a rectangular portion of the image to be captured. **Region of Interest** is defined by an array of four elements [Left, Top, Right, Bottom]. You must set the width [Right-Left] to a multiple of eight. If **Region of Interest** is not connected or empty, the entire acquisition window is captured.



**IMAQ Session In** is a unique identifier that identifies the Interface file. If you are using the default interface, `img0`, no connection is needed.



**Step x** is a horizontal sampling step or horizontal reduction factor. If it is set to its default value of 1, each column of the image is transferred. If **Step x** is set to another value  $n$ , only one column every  $n$  columns is transferred. **Step x** only accepts values of 1, 2, 4, or 8.



**Step y** is a vertical sampling step or vertical reduction factor. If it is set to its default value of 1, each line of the image is transferred. If **Step y** is set to another value  $n$ , only one line every  $n$  lines is transferred. **Step y** only accepts values of 1, 2, 4, or 8.

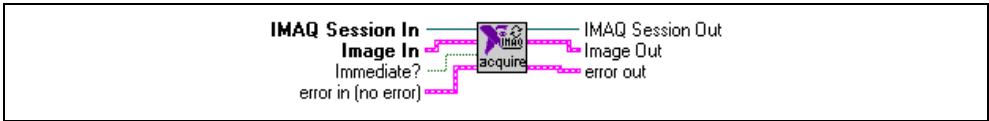


**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Grab Acquire



Acquires an image from a grab acquisition. This function performs an acquisition that loops continually on one buffer. Use the grab function for high-speed image acquisition.

IMAQ Grab Acquire returns a copy of the current image. Use IMAQ Grab Setup to start the acquisition and IMAQ Stop to stop the acquisition.



**IMAQ Session In** is a unique identifier that identifies the Interface file. If you are using the default interface, `img0`, no connection is needed.



**Image In** is the reference to the image that will receive the captured pixel data.



**Immediate?** determines if the grab operation synchronizes on vertical blank or does an immediate transfer. The default value is FALSE—synchronize on vertical blank.



**IMAQ Session Out** has the same value as **IMAQ Session In**.

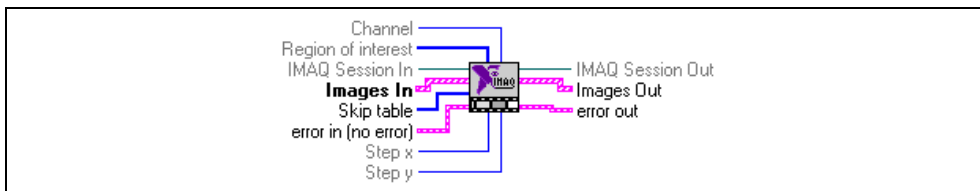


**Image Out** is the reference to the captured image.



**Note** See Appendix B, *Error Codes* for error input and output information and error codes.

## IMAQ Sequence



Starts, acquires, and releases a sequence acquisition. Use this VI to capture multiple images with fixed or variable delays between images. If necessary, this VI initializes the system using the IMAQ Init VI.



**Channel** specifies from which channel on the IMAQ device to acquire.



**Region of Interest** specifies a rectangular portion of the image to be captured. **Region of Interest** is defined by an array of four elements [Left, Top, Right, Bottom]. You must set the width [Right-Left] to a multiple of eight. If **Region of Interest** is not connected or empty, the entire acquisition window is captured.



**IMAQ Session In** is a unique identifier that identifies the Interface file. If you are using the default interface, `img0`, no connection is needed.



**Images In** is an array of image references that will receive the captured pixel data.



**Skip Table** is an array containing the number of frames to skip before acquiring each buffer. Each element in the array specifies the number of frames to skip before acquiring the corresponding buffer in the buffer list. **Skip Table** should contain the same number of elements as the **Images In** array.



**Step x** is a horizontal sampling step or horizontal reduction factor. If it is set to its default value of 1, each column of the image is transferred. If **Step x** is set to another value  $n$ , only one column every  $n$  columns is transferred. **Step x** only accepts values of 1, 2, 4, or 8.



**Step y** is a vertical sampling step or vertical reduction factor. If it is set to its default value of 1, each line of the image is transferred. If **Step y** is set to another value  $n$ , only one line every  $n$  lines is transferred. **Step y** only accepts values of 1, 2, 4, or 8.

**IMAQ**

**IMAQ Session Out** has the same value as **IMAQ Session In**.

**[Images Out]**

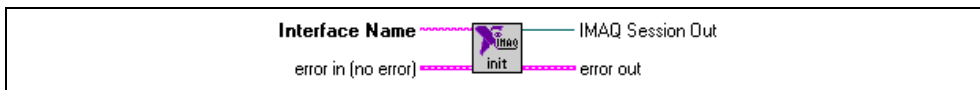
**Images Out** is the array of references to the captured images.



**Note** See Appendix B, *Error Codes* for error input and output information and error codes.

## IMAQ Init

---



Loads an NI-IMAQ configuration file and configures the IMAQ device.



**Interface Name** is the name of the interface to be loaded. The name must match the configuration file name used in the Measurement & Automation Explorer for IMAQ. The default value is `img0`.



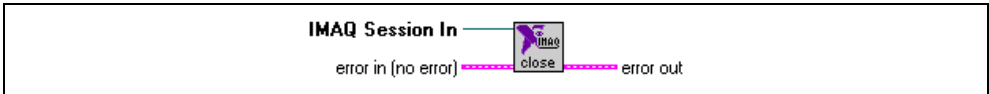
**IMAQ Session** is a unique identifier to an IMAQ session that identifies the Interface file.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Close

---



Stops the acquisition if one is in progress, releases resources associated with the acquisition, and closes the specified **IMAQ Session**.



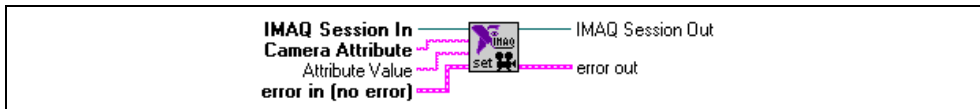
**IMAQ Session In** is a unique identifier that identifies the Interface file.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.



## IMAQ Set Camera Attribute



Sets the value of camera attributes. Consult the `<my camera>.txt` file in the `ni-imaq\camera` directory for more information on valid attributes for your camera.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**abc**

**Camera Attribute** is the attribute name as documented in the `<my camera>.txt` file.

**abc**

**Attribute Value** is the new value of the attribute.

**I32**

**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Get Camera Attribute



Gets the value of camera attributes. Consult the `<my camera>.txt` file in the `ni-imaq\camera` directory for more information on valid attributes for your camera.

**IMAQ Session In**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**Camera Attribute**

**Camera Attribute** is the attribute name as documented in the `<my camera>.txt` file.

**IMAQ Session Out**

**IMAQ Session Out** has the same value as **IMAQ Session In**.

**Attribute Value**

**Attribute Value** is the current value of the attribute.

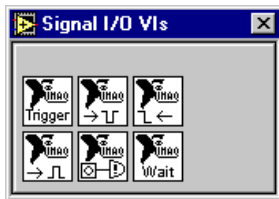


**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## Signal I/O VIs

---

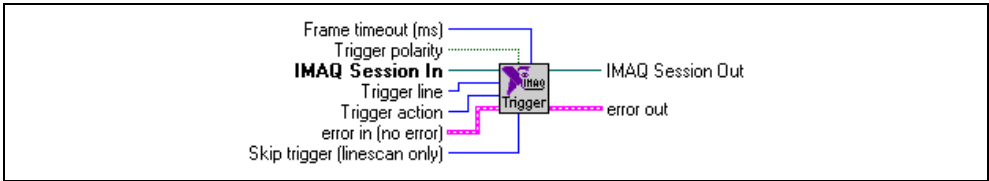
You will see the Signal I/O VI palette shown in Figure 2-2 when you choose the **Function»Image Acquisition»Image Acquisition»Signal I/O VIs** palette.



**Figure 2-2.** NI-IMAQ Event VI Palette

The Signal I/O VIs allow you to control the trigger lines, generate pulses, and wait for a hardware event.

## IMAQ Configure Trigger



Configures the trigger conditions for an acquisition. Use this VI before any acquisition VI to setup a triggered image acquisition.

**I32**

**Frame timeout (ms)** specifies the amount of time in milliseconds that NI-IMAQ waits for the trigger to occur and the image to be captured.

**TF**

**Trigger polarity** specifies the polarity of the trigger signal. FALSE indicates LOW-TRUE; TRUE indicates HIGH-TRUE. The default is LOW-TRUE.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**U16**

**Trigger line** specifies the source of the trigger signal. Values are:

External trigger 0 (0)	RTSI line 2 (6)
External trigger 1 (1)	RTSI line 3 (7)
External trigger 2 (2)	RTSI line 4 (8)
External trigger 3 (3)	RTSI line 5 (9)
RTSI line 0 (4)	RTSI line 6 (10)
RTSI line 1 (5)	

**U16**

**Trigger action** specifies if an assertion edge of this trigger line should start an acquisition. Values are:

Disabled (0)	Triggering is disabled
Trigger start of acquisition (1)	When the assertion edge of the trigger is received, the acquisition is started
Trigger start of each buffer list (2)	When the assertion edge of a trigger is received, the buffer list is acquired. If the acquisition is continuous, buffer index 0 will always wait on a trigger before acquiring

Trigger each buffer (3)

Each buffer waits for a trigger before acquiring an image into the buffer

Trigger each line (4)

Each line is triggered. This is useful when using an encoder to acquire line scan images



**Note** The **Trigger each line** value is valid only for line scan cameras used with the IMAQ PCI-1424 or IMAQ PCI/PXI-1422.



**Skip trigger** is the number of triggers to skip between lines. This is useful when using an encoder to trigger lines. For example, you may have an encoder that outputs 1,000 ticks per revolution. If you only need 100 lines per revolution, set **Skip trigger** to 9.



**Note** This input is valid only when using a line scan camera with the IMAQ PCI-1424 or IMAQ PCI/PXI-1422. Set this input to 0 when using an area scan camera.

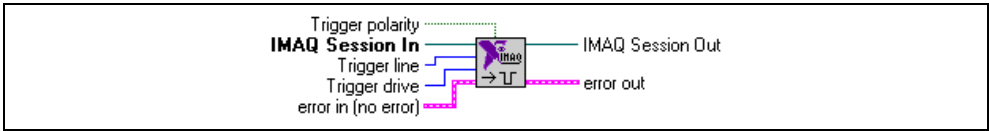


**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Trigger Drive



Drives a trigger line with a specified signal.



**Trigger polarity** specifies the polarity of the trigger signal. FALSE indicates LOW-TRUE; TRUE indicates HIGH-TRUE. The default is LOW-TRUE.



**IMAQ Session In** is a unique identifier that identifies the Interface file.



**Trigger line** specifies the source of the trigger signal. Values are:

External trigger 0 (0)	RTSI line 2 (6)
External trigger 1 (1)	RTSI line 3 (7)
External trigger 2 (2)	RTSI line 4 (8)
External trigger 3 (3)	RTSI line 5 (9)
RTSI line 0 (4)	RTSI line 6 (10)
RTSI line 1 (5)	



**Trigger drive** specifies the signal which will drive the trigger line. Values are:

Disabled (0)	The trigger line is disabled
Acquisition in Progress (1)	High when acquisition is in progress
Acquisition done (2)	Asserted when the entire acquisition is finished
Pixel Clock (3)	Pixel clock times the sampling of pixels
Unasserted (4)	
Asserted (5)	
Horizontal Synchronization Signal (6)	Horizontal synchronization signal produced at the beginning of each line by the camera
Vertical Synchronization Signal (7)	Vertical synchronization signal produced at the beginning of each field by the camera

Frame start (8)	High when a frame is being captured
Frame done (9)	Asserted at the end of each frame that is captured

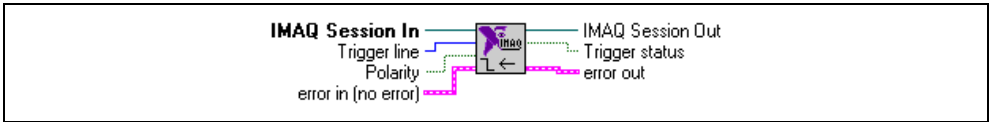


**IMAQ Session out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Trigger Read



Reads the current value of a trigger line.



**Trigger polarity** specifies the polarity of the trigger signal. FALSE indicates LOW-TRUE; TRUE indicates HIGH-TRUE. The default is LOW-TRUE.



**IMAQ Session in** is a unique identifier that identifies the Interface file.



**Trigger line** specifies the source of the trigger signal to read. Values are:

External trigger 0 (0)	RTSI line 2 (6)
External trigger 1 (1)	RTSI line 3 (7)
External trigger 2 (2)	RTSI line 4 (8)
External trigger 3 (3)	RTSI line 5 (9)
RTSI line 0 (4)	RTSI line 6 (10)
RTSI line 1 (5)	



**IMAQ Session out** has the same value as **IMAQ Session In**.



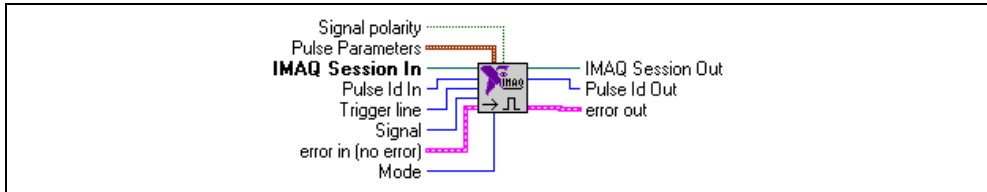
**Trigger status** specifies the current value on the trigger line.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.



## IMAQ Generate Pulse



Generates a pulse on a trigger line. IMAQ Generate Pulse can generate a pulse immediately or on the assertion edge of a status or trigger signal.



**Note** Currently, pulses can be generated only by the IMAQ PCI-1424 and IMAQ PCI/PXI-1422.



**Pulse Parameters** specifies parameters used to describe a pulse.



**Pulse Delay ( $\mu\text{s}$ )** is the desired duration of the first phase of the signal in microseconds. If **Pulse Delay** = 0.0, the VI selects a minimum delay of one cycle of the timebase used.



**Pulse Width ( $\mu\text{s}$ )** is the desired duration of the second phase of the pulse, in microseconds. If **Pulse Width** = 0.0, the VI selects a minimum width of one cycle of the timebase used.



**Pulse Polarity (high:0)** is the polarity of second phase (period two) of each delayed pulse.

- |                |  |
|----------------|--|
| High pulse (0) | Pulse starts at a low TTL level and ends at a high level (default) |
| Low pulse (1)  | Pulse starts at a high TTL level and ends at a low level           |



**IMAQ Session In** is a unique identifier that identifies the Interface file.



**Trigger line** specifies the line on which the pulse will be generated. Values are:

- |                        |                  |
|------------------------|------------------|
| External trigger 0 (0) | RTSI line 2 (6)  |
| External trigger 1 (1) | RTSI line 3 (7)  |
| External trigger 2 (2) | RTSI line 4 (8)  |
| External trigger 3 (3) | RTSI line 5 (9)  |
| RTSI line 0 (4)        | RTSI line 6 (10) |
| RTSI line 1 (5)        |                  |



**Mode** indicates if the pulse is repeated. Values are:

Pulse train (0)	Generate a continuous pulse train on the first assertion edge of <b>Signal</b>
Single pulse (1)	Generate a pulse on the first assertion edge of <b>Signal</b>
Rearmed Single pulse (2)	Generate a pulse on all assertion edges of <b>Signal</b>
Stop (3)	Stop the generation of pulses on <b>Trigger line</b>



**Signal** specifies the signal which will cause the pulse to be generated. The assertion edge of the following signals can initiate pulse generation. Values are:

Acquisition in progress (0)	Asserted when the acquisition begins
Acquisition done (1)	Asserted when the entire acquisition is finished
Frame start (2)	Asserted at the beginning of each frame that is captured
Frame done (3)	Asserted at the end of each frame that is captured
Assertion of external trigger 0 (4)	
Assertion of external trigger 1 (5)	
Assertion of external trigger 2 (6)	
Assertion of external trigger 3 (7)	
Assertion of RTSI line 0 (8)	
Assertion of RTSI line 1 (9)	
Assertion of RTSI line 2 (10)	
Assertion of RTSI line 3 (11)	
Assertion of RTSI line 4 (12)	
Assertion of RTSI line 5 (13)	
Assertion of RTSI line 6 (14)	
Immediate (15)	Asserted immediately when the VI is called



**Signal Polarity** indicates the polarity of the **Signal** input. FALSE indicates LOW-TRUE; TRUE indicates HIGH-TRUE. The default is LOW-TRUE.

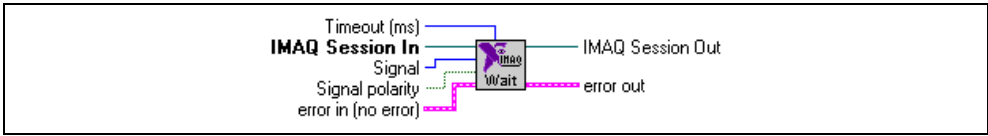


**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Wait Signal



Waits for either a status or trigger signal to be asserted. IMAQ Wait Signal will not return until the specified signal is asserted or a timeout occurs.

**DBL**

**Timeout (ms)** specifies the amount of time to wait for the assertion edge of **Signal** in milliseconds. The VI will return with a timeout error if the assertion edge does not occur within the specified time.

**I32**

**IMAQ Session In** is a unique identifier to an IMAQ session that identifies the Interface file. If you are using the default interface, `img0`, no connection is needed.

**U16**

**Signal** specifies the assertion edge of the signal which will cause the VI to return. Values are:

Acquisition in progress (0)	Asserted when the acquisition begins
Acquisition done (1)	Asserted when the entire acquisition is finished
Frame start (2)	Asserted at the beginning of each frame that is captured
Frame done (3)	Asserted at the end of each frame that is captured
Buffer complete (4)	Asserted when a buffer has been transferred to memory
Assertion of external trigger 0 (5)	
Assertion of external trigger 1 (6)	
Assertion of external trigger 2 (7)	
Assertion of external trigger 3 (8)	
Assertion of RTSI line 0 (9)	
Assertion of RTSI line 1 (10)	
Assertion of RTSI line 2 (11)	
Assertion of RTSI line 3 (12)	
Assertion of RTSI line 4 (13)	

Assertion of RTSI line 5 (14)

Assertion of RTSI line 6 (15)



**Signal Polarity** indicates the polarity of **Signal** input. FALSE indicates LOW-TRUE; TRUE indicates HIGH-TRUE. The default is LOW-TRUE.

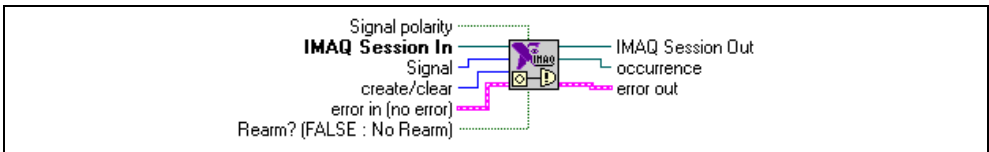


**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Occurrence Config



Creates occurrences that are set by image acquisition signals such as the assertion of a status or trigger signal. Occurrences produced by this VI are used as inputs to the Wait on Occurrence primitive. Anything dependent on the execution of this primitive will sleep until the occurrence is set.



**Signal Polarity** indicates the polarity of **Signal** input. FALSE indicates LOW-TRUE; TRUE indicates HIGH-TRUE. The default is LOW-TRUE.



**IMAQ Session in** is a unique identifier that identifies the Interface file.



**Signal** specifies the assertion edge of the signal which will cause the occurrence to be set. Values are:

Acquisition in progress (0)	Asserted when the acquisition begins
Acquisition done (1)	Asserted when the entire acquisition is finished
Frame start (2)	Asserted at the beginning of each frame that is captured
Frame done (3)	Asserted at the end of each frame that is captured
Buffer complete (4)	Asserted when a buffer has been transferred to memory
Assertion of external trigger 0 (5)	
Assertion of external trigger 1 (6)	
Assertion of external trigger 2 (7)	
Assertion of external trigger 3 (8)	
Assertion of RTSI line 0 (9)	
Assertion of RTSI line 1 (10)	
Assertion of RTSI line 2 (11)	
Assertion of RTSI line 3 (12)	
Assertion of RTSI line 4 (13)	

Assertion of RTSI line 5 (14)

Assertion of RTSI line 6 (15)



**create/clear** instructs this VI to create an occurrence or to clear all occurrences that have been created for the IMAQ device identified **IMAQ Session**. Values are:  
create an occurrence (0)  
clear all occurrences (1)



**Re Arm? (FALSE : No Re Arm)** indicates whether the occurrence needs to be generated once or multiple times.



**IMAQ Session out** has the same value as **IMAQ Session In**.



**occurrence** is the occurrence value created. Wire this output to a wait on occurrence primitive. Wire the output of the primitive to the part of your diagram you wish to execute when the IMAQ event happens and the occurrence is set. If LabVIEW is unable to create an occurrence the occurrence value is set to the **Not a Session** file I/O constant.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

# Low-Level VIs

---

You will see the low-level VI palette shown in Figure 2-3 when you choose the **Function»IMAQ Library»Image Acquisition»IMAQ Low-Level Acquisition** palette.

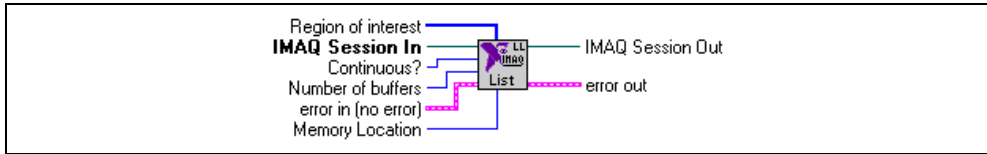


**Figure 2-3.** NI-IMAQ Low-Level VI Palette

The low-level NI-IMAQ VIs give you more direct hardware control.



## IMAQ Configure List



Configures a buffer list to be used in an acquisition. The buffers must be configured individually with IMAQ Configure Buffer.

**I32**

**Region of Interest** specifies a rectangular portion of the image to be captured. **Region of Interest** is defined by an array of four elements [Left, Top, Right, Bottom]. You must set the width [Right-Left] to a multiple of eight. If **Region of Interest** is not connected or empty, the entire acquisition window is captured.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**I32**

**Continuous?** specifies if the acquisition is continuous or one-shot. Values are:

- One-Shot (0) Images are acquired into the buffer list once.
- Continuous (1) Images are acquired continuously into the buffer list until the acquisition is stopped.

**I32**

**Number of Buffers** is the number of buffers to allocate in the buffer list.

**I32**

**Memory Location** specifies if the acquired images will be stored in system memory or in onboard memory on the IMAQ device. Values are:

- System (0) Images stored in system memory.
- Onboard (1) Images stored in RAM on the IMAQ device.

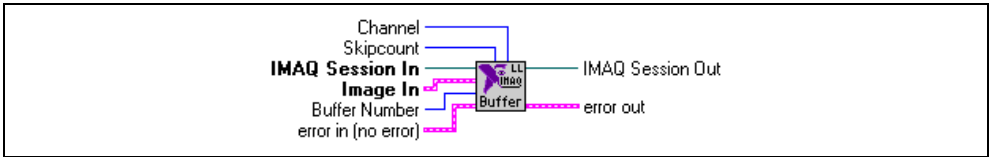
**I32**

**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Configure Buffer



Configures individual buffers in the buffer list. You must call IMAQ Configure Buffer for each buffer in the buffer list.

**U32**

**Channel** is the video source to acquire from the IMAQ device. If **Channel** is not connected or empty, the default channel from the Interface file will be used.

**U32**

**Skipcount** is the number of frames or fields to skip before the acquisition into this buffer.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**I32**

**Image In** is the reference to the image that will receive the captured pixel data.

**U32**

**Buffer Number** is the number of the buffer in the buffer list to configure.

**I32**

**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Start

---



Starts an asynchronous image acquisition. IMAQ Start will return immediately after the acquisition has started. Before calling this VI, you must configure the acquisition with IMAQ Configure List and IMAQ Configure Buffer.



**IMAQ Session In** is a unique identifier that identifies the Interface file.

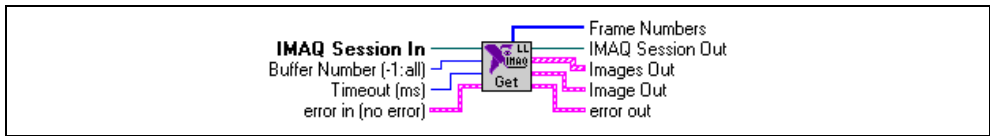


**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Get Buffer



Returns an acquired image (or all acquired images). IMAQ Get Buffer will wait until the requested buffer has been acquired before returning an image. Furthermore, this VI will return an error if the acquisition is continuous and has not been stopped before calling this VI.

If the image was acquired into onboard memory, the image will be copied into a buffer in system memory and then returned.



**IMAQ Session In** is a unique identifier that identifies the Interface file.



**Buffer Number (-1: all)** is the number of the acquired buffer to return. If Buffer Number is -1, all image buffers will be returned in the **Images Out** output.



**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Images Out** is the array of references to the captured images. Contains all images in the buffer list if **Buffer Number** is equal to -1.



**Image Out** is the reference to the captured image.

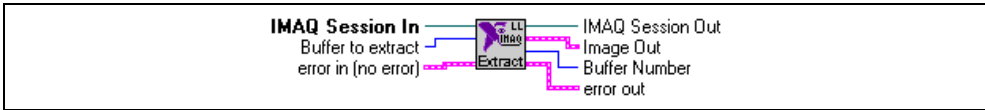


**Frame Numbers** returns the actual cumulative frame number of each buffer in the **Images Out** array. Useful for determining the order of images returned after a ring acquisition.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Extract Buffer



Extracts a buffer from a continuous acquisition. IMAQ Extract Buffer allows for the examination of buffer during acquisition. The buffer is extracted from the acquisition and new data will not be written into the buffer until IMAQ Extract Buffer is called again.

When IMAQ Extract Buffer is called, any currently extracted buffers are reinserted into the buffer list.



**IMAQ Session In** is a unique identifier that identifies the Interface file.



**Buffer to Extract** is the buffer to extract from the buffer list, as specified by the cumulative buffer index. Input a -1 to release the currently extracted buffer without extracting a new buffer.



**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Image Out** is the reference to the captured image.

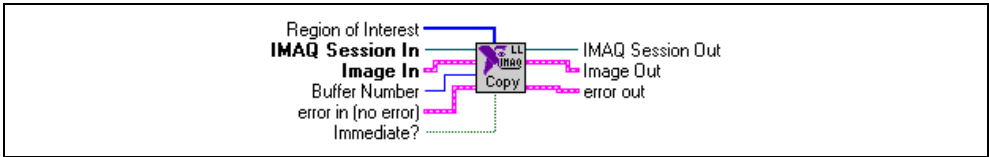


**Buffer Number** is the cumulative buffer number of the buffer in the list that was returned in **Image Out**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Copy



Returns a copy of an acquired image. IMAQ Copy allows for a copy from onboard memory to system memory or from system to system memory.

**[I32]**

**Region of Interest** specifies a rectangular portion of the image to be captured. **Region of Interest** is defined by an array of four elements [Left, Top, Right, Bottom]. You must set the width [Right-Left] to a multiple of eight. If **Region of Interest** is not connected or empty, the entire acquisition window is captured.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**Image In**

**Image In** is the reference to the image that will receive the copied pixel data.

**I32**

**Buffer Number** is the number of the acquired buffer to copy into **Image In**.

**TF**

**Immediate?** determines whether the copy is done immediately or if the copy occurs after the completion of the next acquisition into the buffer. The default value is FALSE, wait for next acquisition.

**I32**

**IMAQ Session Out** has the same value as **IMAQ Session In**.

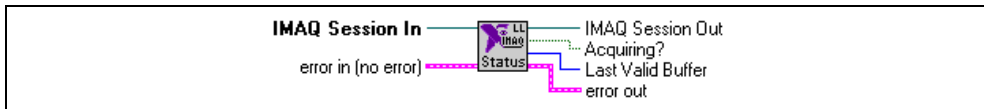
**Image Out**

**Image Out** is the reference to the captured image.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Status



Returns status information about the acquisition such as the state of the acquisition and the last valid buffer acquired.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**I32**

**IMAQ Session Out** has the same value as **IMAQ Session In**.

**TF**

**Acquiring?** specifies if the IMAQ device is acquiring images. **Acquiring?** is TRUE if the device is acquiring.

**I32**

**Last Valid Buffer** is the last buffer that has received an acquired image.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Stop



Stops the currently executing acquisition on the IMAQ device specified by **IMAQ Session**; however, it does not free the resources associated with the acquisition. Call IMAQ Close to free resources.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file.

**I32**

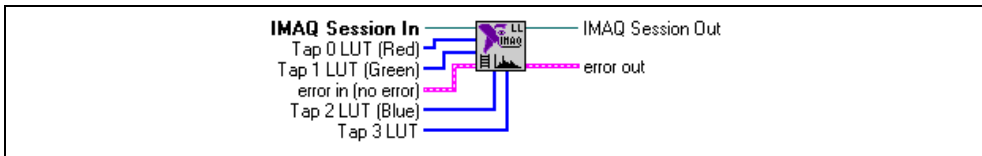
**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.



## IMAQ Set User LUT



Downloads a custom LUT to an IMAQ device. This VI can download 8-bit or 16-bit LUTs for analog or digital devices. If your IMAQ device has multiple LUTs, you must call this VI once for each LUT.

**I32**

**IMAQ Session In** is a unique identifier that identifies the Interface file. If you are using the default interface, `img0`, no connection is needed.

**U8**

**Tap 0 LUT (Red)** is used for analog monochrome devices, the red channel for RGB digital cameras, or tap 0 on digital devices. If your device is an 8-bit device, wire an array of 8-bit unsigned integers containing 256 elements. If your device is greater than 8-bits, wire an array of 16-bit unsigned integers containing  $2^n$  elements where  $n$  is the bit depth of the camera (1,024 for 10-bit, 2,048 for 12-bit, and so on).

**U8**

**Tap 1 LUT (Green)** is used for the green channel for RGB digital cameras or tap 1 on digital devices. If your device is an 8-bit device, wire an array of 8-bit unsigned integers containing 256 elements. If your device is greater than 8-bits, wire an array of 16-bit unsigned integers containing  $2^n$  elements where  $n$  is the bit depth of the camera (1,024 for 10-bit, 2,048 for 12-bit, and so on).

**U8**

**Tap 2 LUT (Blue)** is used for the blue channel for RGB digital cameras or tap 2 on digital devices. If your device is an 8-bit device, wire an array of 8-bit unsigned integers containing 256 elements.

**U8**

**Tap 3 LUT** is used for tap 3 on digital devices. If your device is an 8-bit device, wire an array of 8-bit unsigned integers containing 256 elements.

**I32**

**IMAQ Session Out** has the same value as **IMAQ Session In**.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Vision VIs

---

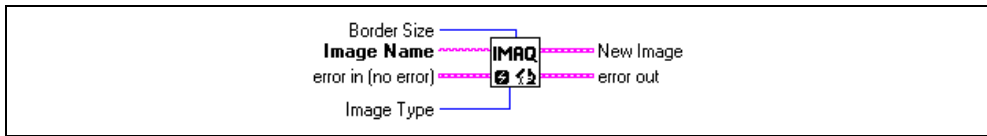
This section describes some basic IMAQ Vision for G functions that you can use with NI-IMAQ VIs. The VIs support creating and disposing of images and the conversion of images to arrays. After you convert an image to an array, you can use standard LabVIEW or BridgeVIEW techniques to process and display the images.

You can find the IMAQ Vision VIs in the **Functions** palette from your block diagram. If you have not purchased IMAQ Vision, the palette shown in Figure 2-4 will appear. If you have IMAQ Vision installed, your palette will contain more functions.



**Figure 2-4.** IMAQ Vision VI Palette

## IMAQ Create



Creates an image buffer that can be input into any of the acquisition functions of your IMAQ device.



**Border Size** determines the width in pixels of the border created around an image. These pixels are required for certain image processing functions.



**Image Name** is the name that will be associated with the created image.



**Image Type** specifies the type of image that is being created. This input is an enumerated type with the following values:

8 bits	8-bit unsigned image used for monochrome images
16 bits	16-bit signed image used for StillColor images and 10-, 12-, and 14-bit acquisitions on the IMAQ PCI-1424 and PCI/PCX-1422
Float	32-bit floating point image, not used with IMAQ devices
Complex	2 by 32-bit floating point image, not used with IMAQ devices
RGB	32-bit standard color image used for StillColor and IMAQ PCI/PCX-1411 images
HSL	32-bit HSL color image used for the IMAQ PCI/PXI-1411 and StillColor images



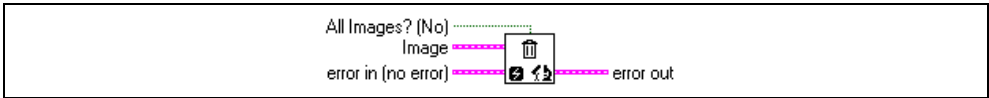
**New Image** is the image structure that will be supplied as an input to all subsequent functions.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ Dispose

---



Disposes an image and frees the memory allocated for the image. Call IMAQ Dispose only when the image is no longer required for the remainder of the processing.



**All images? (No)** determines whether you want to dispose only the image input into this function or dispose all images. The default value is FALSE.

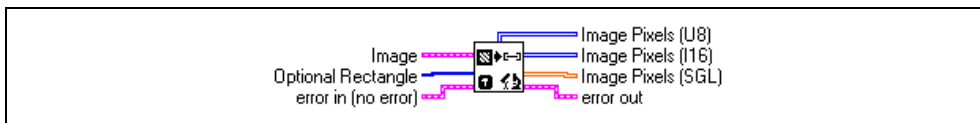


**Image** is the image to be disposed.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ ImageToArray



Copies the incoming image to a G array. This array can be used for pixel processing or displaying in a G intensity graph.



**Image** is the image to be copied. This image must be of an 8-bit or 16-bit type.



**Optional Rectangle** defines an array of four elements containing the coordinates [Left, Top, Right, Bottom] of the region to extract. The operation will be applied to the entire image if the input is empty or not connected.



**Image Pixels (U8)** returns the extracted pixel values into a 2D array [line, column]. This output is used with an 8-bit image type.



**Image Pixels (I16)** returns the extracted pixel values into a 2D array [line, column]. This output is used with a 16-bit image type, which is used only for StillColor acquisitions and with the IMAQ PCI-1424 or IMAQ PCI/PXI-1422.

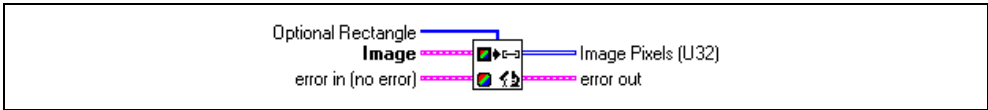


**Image Pixels (SGL)** is not used with IMAQ devices.



**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

## IMAQ ColorImageToArray



Copies the incoming color image to a G array. This VI returns the values as a 2D array of unsigned 32-bit integers.

**[132]**

**Optional Rectangle** defines an array of four elements containing the coordinates [Left, Top, Right, Bottom] of the region to extract. The operation will be applied to the entire image if the input is empty or not connected.

**[56]**

**Image** is the image to be copied. The image must be of type RGB or HSL.

**[U32]**

**Image Pixels (U32)** returns the pixel values as a 2D Array [line, column].



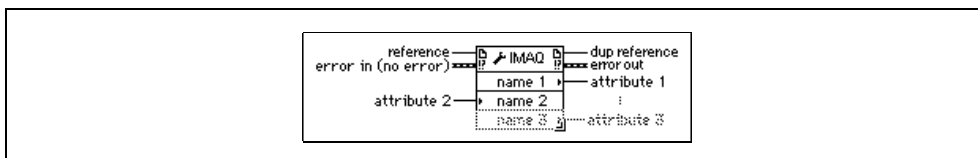
**Note** See Appendix B, *Error Codes*, for error input and output information and error codes.

# IMAQ Property Node

This chapter describes the IMAQ VI property node and how it works with LabVIEW 5.1 or later. If you are using LabVIEW 5.0 or earlier, see the *NI-IMAQ Release Notes*.

## IMAQ Property Node

The IMAQ Property Node gets and/or sets IMAQ Properties. The node is growable; evaluation starts from the top and proceeds downward until an error or the final evaluation occurs. To access the property node, select **Functions»Image Acquisition** and select the **Property Node** icon located on the bottom row of the Image Acquisition palette, as shown in Figure 3-1.



**Figure 3-1.** IMAQ Property Node

The properties are divided into seven major groups—**analog properties**, **board information**, **color**, **image properties**, **session information**, **status information**, and **StillColor**—and are listed alphabetically within each group. Each property discussion contains the description, possible values or range, and datatype.

To select a specific property, pop up on one of the name terminals and select **Properties**. To set property information, pop up and select **Change to Write**. To get property information pop up and select **Change to Read**. Some properties are read only, so you cannot see **Change to Write** in the popup menu.

The Property Node works the same way as Attribute Nodes. If you want to add items to the node, pop up and select **Add Element** or click and drag the node to expand the number of items in the node.

The properties are changed in the order from top to bottom. If an error occurs on one of the properties, the node stops at that property and returns an error. No further properties are handled. The error string reports which property caused the error.

If the small direction arrow on a property is on the left, you are setting the property value. If the small direction arrow on the property is on the right, you are getting the property value. Each property name has a short or long name which you can select by popping up and changing **Name Format**.

## Analog Properties

Analog properties allow you to set analog device parameters such as antichrominance filter and input range.

### Antichrominance Filter



Gets/sets the antichrominance filter used by the IMAQ device. Valid only on devices with an antichrominance filter. Possible values are Disabled, NTSC, and PAL.

Access Privilege: Read/Write

### Black Reference (Volts)



Sets the black reference value of the channel associated with this session. Values are 0–0.5 V.

Access Privilege: Read/Write

### White Reference (Volts)



Sets the white reference value of the channel associated with this session. Possible values are 0–1.26 V.

Access Privilege: Read/Write

## Board Information Properties

Board information properties return information concerning your IMAQ device.

### Color Interface



Returns TRUE if the IMAQ device is color capable.

Access Privilege: Read only

### Interface Type



Returns the IMAQ device associated with this session.

Access Privilege: Read only



## Line Scan Camera



Returns TRUE if the camera associated with this session is a line scan camera.

Access Privilege: Read only

## Maximum Horizontal Resolution



Returns the maximum horizontal resolution of the interface.

Access Privilege: Read only

## Maximum Vertical Resolution



Returns the maximum vertical resolution of the interface.

Access Privilege: Read only

## Onboard RAM



Returns TRUE if the IMAQ device has onboard memory.

Access Privilege: Read only

## Onboard RAM Size



Returns the size of onboard RAM on the IMAQ device in bytes.

Access Privilege: Read only

## Pixel Clock Detect



Determines if the existence of a pixel clock is checked before starting an acquisition.

Access Privilege: Read only

## Pixel Depth



Returns the maximum pixel depth of the IMAQ device in bits.

Access Privilege: Read only

## Serial Number



Returns the serial number of the IMAQ device associated with this session.

Access Privilege: Read only

## Color Properties

Color properties set parameters associated with a color acquisition. Some properties are valid for StillColor acquisition, as noted.

### Color Brightness



Adjusts the brightness of the image or the amount of white light added or subtracted to each pixel. The unit is IRE (percentage of the white level), which ranges from -50 to 50 in increments of 1. Default value is zero. Also valid for StillColor.

Access Privilege: Read/Write

### Color Chroma Bandwidth



Gets/sets the resulting bandwidth of the chroma information of the image.

High Highest bandwidth (default)

Low Lowest bandwidth

Access Privilege: Read/Write

### Color Chroma Comb



Gets/sets the type of comb filter used in the chroma path.

Comb Off Comb filter disabled [default in S-Video (Y/C mode)]

Comb 1 Line Comb filtering using 1 delayed line

Comb 2 Lines Comb filtering using 2 delayed lines

Access Privilege: Read/Write

### Color Chroma Phase



Specifies the value of a correction angle that can be applied to the chroma vector (adjustment of tint). This attribute is only active when an NTSC camera is used. The range is -180 to 180 in degrees. The default value is 0°.

Access Privilege: Read/Write

## Color Chroma Process



Gets/sets the processing applied to the chroma signal. Possible values are:

Chroma Process Always Off

Use with a monochrome camera (default for CCIR or RS-170)

Always On

Use with a color camera (default for NTSC or PAL)

Autodetect

Use if the camera type (monochrome or color) is unknown.

Access Privilege: Read/Write

## Color Chroma Trap



Enables the chroma trap filter in the luma signal path. Should always be disabled in S-Video (Y/C mode). Values are:

FALSE

Chroma trap filter disabled [default in S-Video (Y/C mode)]

TRUE

Chroma trap filter enabled, if the chroma trap filter is needed in composite mode

Access Privilege: Read/Write

## Color Contrast



Adjusts the contrast of the image. The value is a scaling factor applied to every pixel. The contrast adjustment is centered around the median pixel value. (For example, an 8-bit image would be centered around 128.) The range is from 0.5 to 1.5. Default value is 1. Also valid for StillColor.

Access Privilege: Read/Write

## Color HSL Coring Level



Gets/sets the HSL coring level when the Image Representation is chosen to be HSL (Hue, Saturation, Luminance). In HSL mode, on any image pixel, if the saturation value (S) of the pixel is lower than the specified value, then the Hue value (H) of the pixel will be set to the HSL coring replace value. The range is from 0 to 255 in LSBs with a default value of 0.

Access Privilege: Read/Write

## Color HSL Coring Replace Value



Gets/sets the HSL coring replace value when the Image Representation is chosen to be HSL (Hue, Saturation, Luminance). In HSL mode, on any image pixel, if the saturation value (S) of the pixel is lower than the HSL coring level, then the Hue value (H) of the pixel will be set to the HSL coring replace value. The range is from 0 to 255 in LSBs with a default value of 0.

Access Privilege: Read/Write

## Color Hue Offset Angle



Gets/sets the offset angle for the hue calculation. A value of 0 (default) results in a red color to toggle between 0 and max (255 or 32767). Changing this value will move the toggling point to other colors. The unit is degrees, which range from  $-180^\circ$  to  $180^\circ$ . The degree units corresponds to the rotation angle in the chromaticity space. Also valid for StillColor.

Access Privilege: Read/Write

## Color Image Representation



Specifies the type of image data that will be returned when a color image is acquired. Values are RGB32, Red8, Green8, Blue8, Lum8, Hue8, Sat8, Int8, Lum16, Hue16, Sat16, Int16, RGB48, RGB24, RGB16, HSL32, and HSI32. See Appendix A, *Color Image Representation*, for descriptions of these image representations. Also valid for StillColor.

Access Privilege: Read/Write

## Color Luma Bandwidth



Gets/sets the different bandwidths for the luminance signal. Possible values are:

Full	All filters including decimation filter disabled. Default value in CCIR or RS-170 mode
High	Highest available bandwidth with decimation filter enabled. Default value for PAL or NTSC mode
Medium	Decimation filter enabled, medium bandwidth
Low	Decimation filter enabled, lowest bandwidth

Access Privilege: Read/Write

## Color Luma Comb



Gets/sets the type of comb filter used in the luma path. Possible values are:

Comb Off	Comb filter disabled [default in S-Video (Y/C mode)]
Comb 1 Line	Comb filtering using 1 delayed line
Comb 2 Lines	Comb filtering using 2 delayed lines

Access Privilege: Read/Write

## Color NTSC Setup Enable



Enables the setup correction of 7.5 IRE in NTSC mode. A standard NTSC signal has a setup level that moves up the black level 7.5% of the white level (or 7.5 IRE). Correction for this setup can be performed during acquisition by enabling this attribute. Values are:

FALSE	Disables the setup correction (default in PAL or CCIR mode)
TRUE	Enables the setup correction (default in NTSC or RS-170 mode)

Access Privilege: Read/Write

## Color Peaking Enable



Enables the peaking filter in the luma path. Values are:

FALSE	Peaking filter disabled (default)
TRUE	Peaking filter enabled

Access Privilege: Read/Write

## Color RGB Coring Level



Sets/gets the different coring levels. On any image pixel, if the color saturation of the pixel is lower than the specified value, then the saturation will be set to zero resulting in a monochrome pixel. Values are:

No Coring	The coring function is disabled
C1	Coring activated for saturation equal or below 1 LSB
C3	Coring activated for saturation equal or below 3 LSB
C7	Coring activated for saturation equal or below 7 LSB

Access Privilege: Read/Write

## Color Saturation



Gets/sets the color saturation of the image. Saturation of 0 corresponds to a monochrome image. The range is from 0.5 to 1.5. Default value is 1. Also valid for StillColor.

Access Privilege: Read/Write

## Image Properties

Image properties define parameters that affect an image acquisition, such as region of interest.

### Acquire Field



Sets the field acquired when the Frame/Field property is set to Field. Possible values are:

Even	Acquire even fields
Odd	Acquire odd fields
All	Acquire all fields

Even and Odd are valid only for interlaced cameras. All is the only option supported on IMAQ 140x devices.

Access Privilege: Read/Write

### Bits per Pixel



Returns the bits per pixel of the camera associated with this session.

Access Privilege: Read only

### Bytes per Pixel



Returns the bytes per pixel of the camera associated with this session.

Access Privilege: Read only

### Channel



Programs the current channel selected on the interface (0–3), valid on the IMAQ PCI/PXI-1408 only.

Access Privilege: Read/Write

## Frame Timeout (ms)



Gets/sets the frame timeout value in milliseconds.  
Access Privilege: Read/Write

## Horizontal Scaling



Gets/sets the horizontal hardware scaling factor for the channel associated with this session. Possible values are:

None	no scaling
1/2	scale by 2
1/4	scale by 4
1/8	scale by 8

Access Privilege: Read/Write

## Interface Mode



Gets/sets the current mode of the interface (Frame or Field).  
Access Privilege: Read/Write

## Invert



Gets/sets the invert image mode. If this property is set to Invert, the image will be upside down in memory.

Access Privilege: Read/Write

## Look-Up Table



Programs the look-up table (LUT) for the given session. Possible values are Normal, Inverse, Log, Inverse Log, Binary, and Inverse Binary.

Access Privilege: Read/Write

## Region of Interest Height



Gets/sets the region of interest height of the camera/channel associated with this session.

Access Privilege: Read/Write

## Region of Interest Left



Gets/sets the region of interest left of the camera/channel associated with this session.

Access Privilege: Read/Write

## Region of Interest Top



Gets/sets the region of interest top of the camera/channel associated with this session.

Access Privilege: Read/Write

## Region of Interest Width



Gets/sets the region of interest width of the camera/channel associated with this session.

Access Privilege: Read/Write

## Rowpixels



Gets/sets the true width (in pixels) of a horizontal line in memory. Used to calculate the next lines memory offset.

Access Privilege: Read/Write

## Start Field



Gets/sets the start field setting of the camera associated with this session. Valid when the Frame/Field property is set to Frame. Possible values are Even or Odd.

Access Privilege: Read/Write

## Vertical Scaling



Gets/sets the vertical hardware scaling factor for the channel associated with this session. Possible values are:

- |      |            |
|------|------------|
| None | no scaling |
| 1/2  | scale by 2 |
| 1/4  | scale by 4 |
| 1/8  | scale by 8 |

Access Privilege: Read/Write



## Session Information Properties

Session information properties set information about the maximum possible image size for an acquisition.

### Acquisition Window Height



Gets/sets the acquisition window height of the camera associated with this session.

Access Privilege: Read/Write

### Acquisition Window Left



Gets/sets the acquisition window left of the camera associated with this session.

Access Privilege: Read/Write

### Acquisition Window Top



Gets/sets the acquisition window top of the camera associated with this session.

Access Privilege: Read/Write

### Acquisition Window Width



Gets/sets the acquisition window width of the camera associated with this session.

Access Privilege: Read/Write

## Status Information Properties

Status information properties return status information about an acquisition.

### Acquisition in Progress



Returns TRUE if an acquisition is in progress on the camera associated with this session.

Access Privilege: Read only

## Frame Count



Returns the number of frames acquired since the start of the acquisition.  
Access Privilege: Read only

## Free Buffers



Returns the number of reserved driver buffers currently available.  
Access Privilege: Read only

## Last Valid Buffer



Returns the buffer element number of the last received frame buffer.  
Access Privilege: Read only

## Last Valid Frame



Returns the cumulative buffer index of the last image acquired.  
Access Privilege: Read only

## Line Count



Returns the current line count of the frame being acquired.  
Access Privilege: Read only

## Lost Frames



Returns the number of retries if an acquisition was not initially successful.  
Access Privilege: Read only

## StillColor Properties

StillColor properties configure parameters for StillColor acquisitions.

### Color Averaging Count



Gets/sets the number of color images to be acquired and averaged for one output image.  
Access Privilege: Read/Write

## Color High Reference Level in Volts



Gets/sets the hardware white reference level of your IMAQ device when StillColor is selected. Possible values are 0–1.26 V.

Access Privilege: Read/Write

## Color Low Reference Level in Volts



Gets/sets the hardware white reference of your IMAQ device when StillColor is selected. Possible values are 0–0.5 V.

Access Privilege: Read/Write

## Color NTSC Setup Value



Gets/sets the NTSC setup compensation value. The unit is IRE (percentage of white level). Default value is 7.5% (StillColor NTSC only).

Access Privilege: Read/Write

## Color Software Chroma Filter



Gets/sets the software filter to clean the chroma signal (StillColor NTSC only). Possible values are Disabled or Enabled.

Access Privilege: Read/Write

## Color Software Post Gain



Gets/sets the color software post gain value.

Access Privilege: Read/Write

## Color Tint



Gets/sets the tint of your image. Tint is specified in degrees and corresponds to the rotation of the UV color plane. Default value is 0.

Access Privilege: Read/Write

## StillColor Mode



Gets/sets the color acquisition mode. Possible values are:

None                      StillColor mode disabled

RGB                        StillColor RGB mode

Composite                StillColor Composite mode

Access Privilege: Read/Write

# Color Image Representation

This appendix lists image representations used with Color Image Representation property. The *image representation* describes the type of image data that will be returned after color acquisition from the IMAQ PCI/PXI-1411 or a StillColor acquisition. Table A-1 lists a summary of the image representations.

*Image representation* is the constant name of the image representation. *Image type* specifies the image type of the image buffer returned by the image acquisition VI after the color acquisition. *Description* describes the image representation.

**Table A-1.** StillColor Image Representations

Image Representation	Image Type	Description
Blue 8-bit (Blue8)	8-bit	The blue plan encoded in 8 bits extracted from the RGB image.
Green 8-bit (Green8)	8-bit	The green plan encoded in 8 bits extracted from the RGB image.
HSI 32-bit (HSI32)	RGB	A color image encoded in 32 bits, 8 bits unused and 8 bits for the Hue, Saturation, and Intensity planes. (StillColor only)
HSL 32-bit (HSL32)	HSL	A color image encoded in 32 bits, 8 bits unused and 8 bits for the Hue, Saturation, and Luminance planes.
Hue 8-bit (Hue8)	8-bit	The hue information encoded in 8 bits extracted from the RGB image. The hue is calculated as: $H = \text{ATN2}(Y, X)$ where $Y = (\text{Green} - \text{Blue}) / \sqrt{2}$ and $X = (2 \times \text{Red} - \text{Green} - \text{Blue}) / \sqrt{6}$

**Table A-1.** StillColor Image Representations (Continued)

<b>Image Representation</b>	<b>Image Type</b>	<b>Description</b>
Hue 16-bit (Hue16)	16-bit	The hue information encoded in 16 bits extracted from the RGB image. The hue is calculated as: $H = \text{ATN2}(Y, X)$ where $Y = (\text{Green} - \text{Blue}) / \sqrt{2}$ and $X = (2 \times \text{Red} - \text{Green} - \text{Blue}) / \sqrt{6}$ (StillColor only)
Intensity 8-bit (Int8)	8-bit	The intensity information encoded in 8 bits extracted from the RGB image. The intensity is calculated as: $I = (\text{Red} + \text{Green} + \text{Blue}) / 3$
Intensity 16-bit (Int16)	16-bit	The intensity information encoded in 16 bits extracted from the RGB image. The intensity is calculated as: $I = (\text{Red} + \text{Green} + \text{Blue}) / 3$ (StillColor only)
Luminance 8-bit (Lum8)	8-bit	The luminance information encoded in 8 bits extracted from the RGB image. The luminance is calculated as: $Y = 0.299 \times \text{Red} + 0.587 \times \text{Green} + 0.114 \times \text{Blue}$
Luminance 16-bit (Lum16)	16-bit	The luminance information encoded in 16 bits extracted from the RGB image. The luminance is calculated as $Y = 0.299 \times \text{Red} + 0.587 \times \text{Green} + 0.114 \times \text{Blue}$ (StillColor only)
Red 8-bit (Red8)	8-bit	The red plan encoded in 8 bits extracted from the RGB image.
RGB 16-bit (Red16)	16-bit	A color image encoded in 16 bits—5 bits each for the Red, Green, and Blue planes: (0RRR RRGG GGGB BBBB) (StillColor only)
RGB 32-bit (RGB32)	RGB	A color image encoded in 32 bits—8 bits for the alpha channel (which is not used) and 8 bits each for the Red, Green, and Blue planes.

**Table A-1.** StillColor Image Representations (Continued)

Image Representation	Image Type	Description
Saturation 8-bit (Sat8)	8-bit	<p>The saturation information encoded in 8 bits extracted from the RGB image. The saturation is calculated for StillColor as:</p> $S = \sqrt{X^2 + Y^2}$ <p>where <math>Y = (\text{Green} - \text{Blue}) / \sqrt{2}</math> and <math>X = (2 \times \text{Red} - \text{Green} - \text{Blue}) / \sqrt{6}</math></p> <p>The saturation is calculated on the IMAQ PCI/PXI-1411 as:</p> $S = 255 \left( 1 - \frac{3 \times \min(\text{Red}, \text{Green}, \text{Blue})}{\text{Red} + \text{Green} + \text{Blue}} \right)$
Saturation 16-bit (Sat16)	16-bit	<p>The saturation information encoded in 16 bits extracted from the RGB image. The saturation is calculated as:</p> $S = \sqrt{X^2 + Y^2}$ <p>where <math>Y = (\text{Green} - \text{Blue}) / \sqrt{2}</math> and <math>X = (2 \times \text{Red} - \text{Green} - \text{Blue}) / \sqrt{6}</math></p> <p>(StillColor only)</p>

---

# Error Codes

This appendix lists the error format and codes for the NI-IMAQ VIs.

---

## Format

---

Error format for all NI-IMAQ VIs is the same, as follows:



**error in (no error)** is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its error out cluster. If no error has occurred, then this VI executes normally and sets its own error status in error out. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.



**status** is TRUE if an error occurred before this VI was called, or FALSE if not. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code.



**code** is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.



**source** is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.



**error out** is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.





**status** is TRUE if an error occurred, or FALSE if not. If **status** is TRUE, code is a non-zero error code. If **status** is FALSE, code can be zero or a warning code.



**code** is the number identifying an error or warning. If **status** is TRUE, code is a non-zero error code. If **status** is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.



**source** is a string that indicates the origin of the error, if any. Usually **source** is the name of the VI in which the error occurred.

## Error Codes

Table B-1 lists the NI-IMAQ VI error codes.

**Table B-1.** NI-IMAQ VI Error Codes

Error Code	Status Name	Description
-1074397183	IMG_ERR_NCAP	Function not implemented
-1074397182	IMG_ERR_OVRN	Too many interfaces open
-1074397181	IMG_ERR_EMEM	Not enough memory to perform the operation
-1074397180	IMG_ERR_OSER	Operating system error occurred
-1074397179	IMG_ERR_PAR1	Function-specific, see function description
-1074397178	IMG_ERR_PAR2	Function-specific, see function description
-1074397177	IMG_ERR_PAR3	Function-specific, see function description
-1074397176	IMG_ERR_PAR4	Function-specific, see function description
-1074397175	IMG_ERR_PAR5	Function-specific, see function description
-1074397174	IMG_ERR_PAR6	Function-specific, see function description
-1074397173	IMG_ERR_PAR7	Function-specific, see function description
-1074397172	IMG_ERR_MXBF	Too many buffers already allocated
-1074397171	IMG_ERR_DLLE	DLL internal error, bad logic state
-1074397170	IMG_ERR_BSIZ	Buffer size used is too small for minimum acquisition frame

**Table B-1.** NI-IMAQ VI Error Codes (Continued)

<b>Error Code</b>	<b>Status Name</b>	<b>Description</b>
-1074397169	IMG_ERR_MXBI	Exhausted buffer IDs
-1074397168	IMG_ERR_ELCK	Cannot lock buffers down, no more memory
-1074397167	IMG_ERR_DISE	Error releasing the image buffer
-1074397166	IMG_ERR_BBUF	Bad buffer pointer in list
-1074397165	IMG_ERR_NLCK	Buffer list is not locked
-1074397164	IMG_ERR_NCAM	No camera defined for this channel
-1074397163	IMG_ERR_BINT	Bad interface
-1074397162	IMG_ERR_BROW	Rowbytes is less than region of interest
-1074397161	IMG_ERR_BROI	ROI width is greater than rowbytes
-1074397160	IMG_ERR_BCMF	Bad camera file (check syntax)
-1074397159	IMG_ERR_NVBL	Not successful because of hardware limitations
-1074397158	IMG_ERR_NCFG	Invalid action, no buffers configured for session
-1074397157	IMG_ERR_BBLF	Buffer list does not contain a valid final command
-1074397156	IMG_ERR_BBLE	Buffer list contains an invalid command
-1074397155	IMG_ERR_BBLB	A buffer list buffer is null
-1074397154	IMG_ERR_NAIP	No acquisition in progress
-1074397153	IMG_ERR_VLCK	Cannot get lock on video source
-1074397152	IMG_ERR_BDMA	Bad DMA transfer
-1074397151	IMG_ERR_AIOP	Cannot perform request, acquisition in progress
-1074397150	IMG_ERR_TIMO	Wait timed out, acquisition not complete
-1074397149	IMG_ERR_NBUF	No buffers available, too early in acquisition
-1074397148	IMG_ERR_ZBUF	Zero buffer size, no bytes filled

**Table B-1.** NI-IMAQ VI Error Codes (Continued)

<b>Error Code</b>	<b>Status Name</b>	<b>Description</b>
-1074397147	IMG_ERR_HLPR	Bad parameter to low-level—check attributes and high-level arguments
-1074397146	IMG_ERR_BTRG	Trigger loopback problem—cannot drive trigger with action enabled
-1074397145	IMG_ERR_NINF	No interface found
-1074397144	IMG_ERR_NDLL	Unable to load DLL (LabWindows/CVI only)
-1074397143	IMG_ERR_NFNC	Unable to find API function in DLL (LabWindows/CVI only)
-1074397142	IMG_ERR_NOSR	Unable to allocate system resources (LabWindows/CVI only)
-1074397141	IMG_ERR_BTAC	No trigger action—acquisition will time out
-1074397140	IMG_ERR_FIFO	FIFO overflow caused acquisition to halt
-1074397139	IMG_ERR_MLCK	Memory lock error, cannot perform acquisition
-1074397138	IMG_ERR_ILCK	Interface locked
-1074397137	IMG_ERR_NEPK	No external pixel clock
-1074397136	IMG_ERR_SCLM	Field scaling mode not supported
-1074397135	IMG_ERR_SCC1	Channel not set to 1 when using StillColor RGB acquisition
-1074397134	IMG_ERR_SMALLALLOC	Error during small buffer allocation
-1074397133	IMG_ERR_ALLOC	Error during large buffer allocation
-1074397132	IMG_ERR_BADCAMTYPE	Bad camera type; camera needs to be of type NTSC or PAL
-1074397131	IMG_ERR_BADPIXTYPE	Camera not supported; must be an 8-bit camera
-1074397130	IMG_ERR_BADCAMPARAM	Bad camera parameter in configuration file
-1074397129	IMG_ERR_PALKEYDTCT	PAL key detection error
-1074397128	IMG_ERR_BFRQ	Bad frequency values
-1074397127	IMG_ERR_BITP	Bad interface type

**Table B-1.** NI-IMAQ VI Error Codes (Continued)

<b>Error Code</b>	<b>Status Name</b>	<b>Description</b>
-1074397126	IMG_ERR_HWNC	Hardware not capable of supporting this function
-1074397125	IMG_ERR_SERIAL	Serial port error
-1074397124	IMG_ERR_MXPI	Exhausted pulse IDs
-1074397123	IMG_ERR_BPID	Bad pulse ID
-1074397121	IMG_ERR_SERIAL_TIMO	Serial transmit/receive timeout
-1074397120	IMG_ERR_PG_TOO_MANY	Too many pattern generation transitions defined
-1074397119	IMG_ERR_PG_BAD_TRANS	Bad pattern generation transition time
-1074397118	IMG_ERR_PLNS	Pulse not started
-1074397117	IMG_ERR_BPMD	Bad pulse mode
-1074397116	IMG_ERR_NSAT	Nonsettable attribute
-1074397115	IMG_ERR_HYBRID	Cannot mix system and onboard memory buffers
-1074397114	IMG_ERR_BADFILFMT	Pixel depth not supported with this file format
-1074397113	IMG_ERR_BADFILEXT	File extension not supported
-1074397112	IMG_ERR_NRTSI	Too many RTSI triggers mapped; you can use only four RTSI lines at once
-1074397111	IMG_ERR_MXTRG	Exhausted trigger resources
-1074397110	IMG_ERR_MXRC	Exhausted resource (general)
-1074397109	IMG_ERR_OOR	Parameter out of range
-1074397108	IMG_ERR_NPROG	FPGA not programmed
-1074397107	IMG_ERR_NEOM	Not enough onboard memory to perform the operation
-1074397106	IMG_ERR_BDTYPE	Bad display type; buffer cannot be displayed with <code>imgPlot</code>
-1074397105	IMG_ERR_THRDACC DEN	Thread denied access to function
-1074397104	IMG_ERR_BADFILWRT	Could not write the file

**Table B-1.** NI-IMAQ VI Error Codes (Continued)

<b>Error Code</b>	<b>Status Name</b>	<b>Description</b>
1073086470	IMG_WRN_OATTR	Warning, old attribute used
1073086469	IMG_WRN_BRST	Warning, bad quality colorburst (StillColor mode)
1073086468	IMG_WRN_BLK_G	Warning, unstable blanking reference (StillColor mode)
1073086467	IMG_WRN_ILCK	Warning, interface still locked
1073086466	IMG_WRN_CONF	Warning, change requires reconfiguration to take effect
1073086465	IMG_WRN_BCAM	Warning, corrupt camera file detected



---

# Technical Support Resources

This appendix describes the comprehensive resources available to you in the Technical Support section of the National Instruments Web site and provides technical support telephone numbers for you to use if you have trouble connecting to our Web site or if you do not have internet access.

## NI Web Support

---

To provide you with immediate answers and solutions 24 hours a day, 365 days a year, National Instruments maintains extensive online technical support resources. They are available to you at no cost, are updated daily, and can be found in the Technical Support section of our Web site at [www.natinst.com/support](http://www.natinst.com/support).

### Online Problem-Solving and Diagnostic Resources

- **KnowledgeBase**—A searchable database containing thousands of frequently asked questions (FAQs) and their corresponding answers or solutions, including special sections devoted to our newest products. The database is updated daily in response to new customer experiences and feedback.
- **Troubleshooting Wizards**—Step-by-step guides lead you through common problems and answer questions about our entire product line. Wizards include screen shots that illustrate the steps being described and provide detailed information ranging from simple getting started instructions to advanced topics.
- **Product Manuals**—A comprehensive, searchable library of the latest editions of National Instruments hardware and software product manuals.
- **Hardware Reference Database**—A searchable database containing brief hardware descriptions, mechanical drawings, and helpful images of jumper settings and connector pinouts.
- **Application Notes**—A library with more than 100 short papers addressing specific topics such as creating and calling DLLs, developing your own instrument driver software, and porting applications between platforms and operating systems.

## Software-Related Resources

- **Instrument Driver Network**—A library with hundreds of instrument drivers for control of standalone instruments via GPIB, VXI, or serial interfaces. You also can submit a request for a particular instrument driver if it does not already appear in the library.
- **Example Programs Database**—A database with numerous, non-shipping example programs for National Instruments programming environments. You can use them to complement the example programs that are already included with National Instruments products.
- **Software Library**—A library with updates and patches to application software, links to the latest versions of driver software for National Instruments hardware products, and utility routines.

## Worldwide Support

---

National Instruments has offices located around the globe. Many branch offices maintain a Web site to provide information on local services. You can access these Web sites from [www.natinst.com/worldwide](http://www.natinst.com/worldwide).

If you have trouble connecting to our Web site, please contact your local National Instruments office or the source from which you purchased your National Instruments product(s) to obtain support.

For telephone support in the United States, dial 512 795 8248. For telephone support outside the United States, contact your local branch office:

Australia 03 9879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20,  
Brazil 011 284 5011, Canada (Ontario) 905 785 0085,  
Canada (Québec) 514 694 8521, China 0755 3904939,  
Denmark 45 76 26 00, Finland 09 725 725 11, France 01 48 14 24 24,  
Germany 089 741 31 30, Hong Kong 2645 3186, India 91805275406,  
Israel 03 6120092, Italy 02 413091, Japan 03 5472 2970,  
Korea 02 596 7456, Mexico (D.F.) 5 280 7625,  
Mexico (Monterrey) 8 357 7695, Netherlands 0348 433466,  
Norway 32 27 73 00, Singapore 2265886, Spain (Madrid) 91 640 0085,  
Spain (Barcelona) 93 582 0251, Sweden 08 587 895 00,  
Switzerland 056 200 51 51, Taiwan 02 2377 1200,  
United Kingdom 01635 523545

# Glossary

---

Prefix	Meanings	Value
p-	pico-	$10^{-12}$
n-	nano-	$10^{-9}$
$\mu$ -	micro-	$10^{-6}$
m-	milli-	$10^{-3}$
k-	kilo-	$10^3$
M-	mega-	$10^6$
G-	giga-	$10^9$

## Numbers/Symbols

– Negative of, or minus.

$\Omega$  Ohm.

/ Per.

% Percent.

$\pm$  Plus or minus.

+ Positive of, or plus.

## A

A Amperes.

AC Alternating current.

acquisition window The image size specific to a video standard or camera resolution.

active line region The region of lines actively being stored; defined by a line start (relative to VSYNC) and a line count.



active pixel region	The region of pixels actively being stored; defined by a pixel start (relative to HSYNC) and a pixel count.
A/D	Analog-to-digital.
ADC	Analog-to-digital converter. An electronic device, often an integrated circuit, that converts an analog voltage to a digital number.
address	Character code that identifies a specific location (or series of locations) in memory.
ANSI	American National Standards Institute.
antichrominance filter	Removes the color information from the video signal.
API	Application programming interface.
area	A rectangular portion of an acquisition window or frame that is controlled and defined by software.
array	Ordered, indexed set of data elements of the same type.
ASIC	Application-Specific Integrated Circuit. A proprietary semiconductor component designed and manufactured to perform a set of specific functions for a specific customer.
aspect ratio	The ratio of a picture or image's width to its height.
<b>B</b>	
b	Bit. One binary digit, either 0 or 1.
B	Byte. Eight related bits of data, an eight-bit binary number; also used to denote the amount of memory required to store one byte of data.
back porch	The area of the video signal between the rising edge of the horizontal sync signal and the active video information.
black reference level	The level that represents the darkest an image can get. <i>See also</i> white reference level.
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

**C** Celsius.

**cache** High-speed processor memory that buffers commonly used instructions or data to increase processing throughput.

**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.

dB	Decibel. The unit for expressing a logarithmic measure of the ratio of two signal levels: $dB = 20\log_{10} V1/V2$ , for signals in volts.
DC	Direct current.
default setting	A default parameter value recorded in the driver; in many cases, the default input of a control is a certain value (often 0) that means <i>use the current default setting</i> .
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.
drivers	Software that controls a specific hardware device such as an IMAQ or DAQ device.
dynamic range	The ratio of the largest signal level a circuit can handle to the smallest signal level it can handle (usually taken to be the noise level), normally expressed in decibels.

## E

EEPROM	Electrically erasable programmable read-only memory. ROM that can be erased with an electrical signal and reprogrammed.
external trigger	A voltage pulse from an external source that triggers an event such as A/D conversion.

## F

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 the odd-numbered lines, 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; FIFOs are used on IMAQ devices to temporarily store incoming data until that data can be retrieved.

flash ADC	An ADC whose output code is determined in a single step by a bank of comparators and encoding logic.
frame	A complete image; in interlaced formats, a frame is composed of two fields.
front porch	The area of a video signal between the start of the horizontal blank and the start of the horizontal sync.
ft	Feet.
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; examples of functions are:  $y = \text{COS}(x)$ <b>status</b> = AO_config( <b>board, channel, range</b> )

## G

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.
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.

## H

h	Hour.
hardware	The physical components of a computer system, such as the circuit boards, plug-in boards, chassis, enclosures, peripherals, cables, and so on.
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. <i>See also</i> RGB.
Hz	Hertz. The number of scans read or updates written per second.
<b>I</b>	
IC	Integrated circuit.
ID	Identification.
IEEE	Institute of Electrical and Electronics Engineers.
in.	Inches.
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 driver	A set of high-level software functions, such as NI-IMAQ, 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.
interpreter	A software utility that executes source code from a high-level language such as Basic, C or Pascal, by reading one line at a time and executing the specified operation. <i>See also</i> compiler.
interrupt	A computer signal indicating that the CPU should suspend its current task to service a designated activity.
interrupt level	The relative priority at which a device can interrupt.
I/O	Input/output. The transfer of data to/from a computer system involving communications channels, operator interface devices, and/or data acquisition and control interfaces.

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.
IRQ	Interrupt request.
<b>K</b>	
k	Kilo. The standard metric prefix for 1,000, or $10^3$ , used with units of measure such as volts, hertz, and meters.
K	Kilo. The prefix for 1,024, or $2^{10}$ , used with B in quantifying data or computer memory.
kbytes/s	A unit for data transfer that means 1,000 or $10^3$ bytes/s.
Kword	1,024 words of memory.
<b>L</b>	
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.
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	Look-up table. A selection in the Measurement & Automation Explorer for IMAQ that contains formulas that let you implement simple imaging operations such as contrast enhancement, data inversion, gamma manipulation, or other nonlinear transfer functions.

## M

m	Meters.
M	(1) Mega, the standard metric prefix for 1 million or $10^6$ , when used with units of measure such as volts and hertz; (2) mega, the prefix for 1,048,576, or $2^{20}$ , when used with B to quantify data or computer memory.
MB	Megabytes of memory.
Mbytes/s	A unit for data transfer that means 1 million or $10^6$ bytes/s.
memory buffer	<i>See</i> buffer.
memory window	Continuous blocks of memory that can be accessed quickly by changing addresses on the local processor.
MSB	Most significant bit.
MTBF	Mean time between failure.
mux	Multiplexer. Switching device with multiple inputs that selectively connects one of its inputs to its output.

## N

NI-IMAQ	Driver software for National Instruments IMAQ hardware.
noninterlaced	A video frame where all the lines are scanned sequentially, instead of 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. <i>See also</i> PAL.
NVRAM	Nonvolatile RAM. RAM that is not erased when a device loses power or is turned off.

## O

operating system	Base-level software that controls a computer, runs programs, interacts with users, and communicates with installed hardware or peripheral devices.
------------------	--

**P**

PAL	Phase Alternation Line. One of the European video color standards; uses 625 lines per frame. <i>See also</i> 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 132 Mbytes/s.
PCLK	Pixel clock signal. Times the sampling of pixels on a video line.
PFI	Programmable function input.
PGIA	Programmable gain instrumentation amplifier.
picture aspect ratio	The ratio of the active pixel region to the active line region; for standard video signals like RS-170 or CCIR, the full-size picture aspect ratio normally is 4/3 (1.33).
pixel	Picture element. The smallest division that makes up the 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 aspect ratio	The ratio between the physical horizontal size and the vertical size of the region covered by the pixel; an acquired pixel should optimally be square, thus the optimal value is 1.0, but typically it falls between 0.95 and 1.05, depending on camera quality.
pixel clock	Divides the incoming horizontal video line into pixels.
pixel count	The total number of pixels between two HYSNCs; the pixel count determines the frequency of the pixel clock.
PLL	Phase-locked loop. Circuitry that provides a very stable pixel clock that is referenced to another signal, for example, an incoming HSYNC signal.
protocol	The exact sequence of bits, characters, and control codes used to transfer data between computers and peripherals through a communications channel.
pts	Points.
PXI	PCI eXtensions for Instrumentation. An open specification that builds off the CompactPCI specification by adding instrumentation-specific features.



## 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 accuracy	A measure in LSB of the accuracy of an ADC; it includes all 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 deliver three signals, one for each primary.
ribbon cable	A flat cable in which the conductors are side by side.
ROI	Region-of-interest. 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 IMAQ and DAQ boards directly, by means of connectors on top of the boards, for precise synchronization of functions.

## S

s	Seconds.
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 down circuitry	Circuitry that scales down the resolution of a video signal.
scatter-gather DMA	A type of DMA that allows the DMA controller to reconfigure on-the-fly.

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 board.
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	The set of rules to which statements must conform in a particular programming language.
system RAM	RAM installed on a personal computer and used by the operating system, as contrasted with onboard RAM.
<b>T</b>	
transfer rate	The rate, measured in bytes/s, at which data is moved from source to destination after software initialization and set up operations; the maximum rate at which the hardware can operate.
trigger	Any event that causes or starts some form of data capture.
trigger control and mapping circuitry	Circuitry that routes, monitors, and drives the external and RTSI bus trigger lines; you can configure each of these lines to start or stop acquisition on a rising or falling edge.
TTL	Transistor-transistor logic.
<b>U</b>	
UV plane	<i>See</i> YUV.
<b>V</b>	
V	Volts.
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. (1) 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 line** A video line consists of a HSYNC, back porch, 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.

## **Y**

**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 representing the coordinates in a color plane.

# Index

---

## A

### acquisition types

- grab, 1-7 to 1-8
- ring, 1-9
- sequence, 1-8
- snap, 1-7

acquisition VIs. *See* high-level VIs; low-level VIs.

analog properties, IMAQ Property Node, 3-2

attributes. *See* camera attributes.

## B

board information properties, IMAQ Property Node, 3-2 to 3-3

BridgeVIEW Functions palette with IMAQ palette (figure), 1-3

BridgeVIEW software, 1-1

buffer management, 1-6

buffer management VIs

- IMAQ Create VI, 1-6, 1-11, 2-36
- IMAQ Dispose VI, 1-6, 2-37
- IMAQ Extract Buffer VI, 1-10, 2-30
- IMAQ Get Buffer VI, 1-10, 2-29

## C

camera attribute VIs

- IMAQ Get Camera Attribute VI, 1-16, 2-11
- IMAQ Set Camera Attribute VI, 1-16, 2-10

camera attributes, 1-15 to 1-16

color image acquisition. *See* StillColor acquisition.

color image representation, A-1 to A-3

color properties, IMAQ Property Node, 3-4 to 3-8

composite snap, 1-10 to 1-11

configuration VIs

- IMAQ Configure Buffer, 1-9 to 1-10, 2-27
- IMAQ Configure List, 1-9, 2-26
- IMAQ Configure Trigger VI, 1-12, 2-13 to 2-14
- IMAQ Init, 1-9, 1-11, 2-8
- IMAQ Occurrence Config, 2-23 to 2-24

conventions used in manual, ix

## E

error codes

- format, B-1 to B-2
  - list of codes (table), B-2 to B-5
- error handling, 1-4 to 1-5
- error clusters (figure), 1-5
  - Simple Error Handler VI, 1-5

examples for NI-IMAQ, 1-2

## F

Functions palettes

- BridgeVIEW Functions palette with IMAQ palette (figure), 1-3
- LabVIEW Functions palette with IMAQ palette (figure), 1-2

## G

grab acquisition type, 1-7 to 1-8. *See also* IMAQ Grab Acquire VI; IMAQ Grab Setup VI.

## H

high-level VIs, 2-2 to 2-11

- IMAQ Close, 1-10, 2-9
- IMAQ Get Camera Attribute, 1-16, 2-11
- IMAQ Grab Acquire, 2-5

- IMAQ Grab Setup, 2-4
- IMAQ Init, 1-11, 2-8
- IMAQ Sequence, 2-6 to 2-7
- IMAQ Set Camera Attribute, 1-16, 2-10
- IMAQ Snap, 1-7, 1-11, 1-12, 2-3
- overview, 1-9
- palette (figure), 2-2

## I

- Image Acquisition palette, 1-3
- image display, 1-13 to 1-15
  - Intensity Graph (figure), 1-14
  - using IMAQ WindDraw (example), 1-13
- Image in parameter, 1-4
- Image out parameter, 1-4
- image properties, IMAQ Property Node, 3-8 to 3-10
- IMAQ Close VI
  - description, 2-9
  - low-level acquisition procedure, 1-10
- IMAQ ColorImageToArray VI
  - description, 2-39
  - image display (example), 1-15
- IMAQ Configure Buffer VI
  - description, 2-27
  - low-level acquisition steps, 1-9 to 1-10
- IMAQ Configure List VI
  - description, 2-26
  - low-level acquisition steps, 1-9
- IMAQ Configure Trigger VI
  - description, 2-13 to 2-14
  - triggering example, 1-12 to 1-13
- IMAQ Copy VI
  - description, 2-31
  - low-level acquisition procedure, 1-10
- IMAQ Create VI
  - buffer management (figure), 1-6
  - composite snap (example), 1-11
  - description, 2-36
- IMAQ Dispose VI
  - buffer management (figure), 1-6
  - description, 2-37
- IMAQ Extract Buffer VI
  - description, 2-30
  - low-level acquisition procedure, 1-10
- IMAQ Generate Pulse VI, 2-18 to 2-20
- IMAQ Get Buffer VI
  - description, 2-29
  - low-level acquisition procedure, 1-10
- IMAQ Get Camera Attribute VI
  - description, 2-11
  - getting value of attribute (example), 1-16
- IMAQ Grab Acquire VI
  - description, 2-5
  - example (figure), 1-8
  - purpose and use, 1-7 to 1-8
- IMAQ Grab Setup VI
  - description, 2-4
  - example (figure), 1-8
  - purpose and use, 1-7 to 1-8
- IMAQ ImageToArray VI
  - description, 2-38
  - image display (example), 1-14
- IMAQ Init VI
  - composite snap (example), 1-11
  - description, 2-8
  - low-level acquisition steps, 1-9
- IMAQ Low Level palette, 1-3
- IMAQ Occurrence Config VI, 2-23 to 2-24
- IMAQ Property Node, 3-1 to 3-14
  - analog properties, 3-2
  - board information properties, 3-2 to 3-3
  - color properties, 3-4 to 3-8
  - composite snap, 1-11
  - image properties, 3-8 to 3-10
  - overview, 3-1 to 3-2
  - Property Node icon (figure), 3-1
  - session information properties, 3-11

- status information properties, 3-11 to 3-12
- StillColor properties, 3-12 to 3-14
- IMAQ Sequence VI
  - description, 2-6 to 2-7
  - sequence acquisition (example), 1-8
- IMAQ Session In parameter, 1-4
- IMAQ Session Out parameter, 1-4
- IMAQ Session parameter, 1-4
- IMAQ Set Camera Attribute VI
  - description, 2-10
  - setting values (example), 1-16
- IMAQ Set User LUT VI, 2-34
- IMAQ Signal I/O palette, 1-3
- IMAQ Snap VI
  - composite snap (example), 1-11
  - description, 2-3
  - RGB snap (example), 1-12
  - simplified block diagram (figure), 1-7
  - snap acquisition (example), 1-7
- IMAQ Start VI
  - description, 2-28
  - low-level acquisition procedure, 1-10
- IMAQ Status VI, 2-32
- IMAQ Stop VI
  - description, 2-33
  - low-level acquisition procedure, 1-10
- IMAQ Trigger Drive VI, 2-15 to 2-16
- IMAQ Trigger Read VI, 2-17
- IMAQ Vision VIs, 2-35 to 2-39
  - IMAQ ColorImageToArray, 1-15, 2-39
  - IMAQ Create, 1-6, 1-11, 2-36
  - IMAQ Dispose, 2-37
  - IMAQ ImageToArray, 1-14, 2-38
  - palette (figure), 2-35
- IMAQ Wait Signal VI, 2-21 to 2-22
- IMAQ WindDraw (example), 1-13
- Intensity Graph, for displaying images (figure), 1-14

## L

- LabVIEW Functions Palette with IMAQ palette (figure), 1-2
- LabVIEW software, 1-1
- low-level VIs, 2-25 to 2-34
  - general steps for low-level acquisitions, 1-9 to 1-10
  - IMAQ Configure Buffer, 1-9 to 1-10, 2-27
  - IMAQ Configure List, 1-9, 2-26
  - IMAQ Copy, 1-10, 2-31
  - IMAQ Extract Buffer, 1-10, 2-30
  - IMAQ Get Buffer, 1-10, 2-29
  - IMAQ Set User LUT, 2-34
  - IMAQ Start, 1-10, 2-28
  - IMAQ Status, 2-32
  - IMAQ Stop, 1-10, 2-33
  - overview, 1-9 to 1-10
  - palette (figure), 2-25

## N

- NI Web support, C-1 to C-2
- NI-IMAQ acquisition types, 1-6 to 1-9
  - grab, 1-7 to 1-8
  - ring, 1-9
  - sequence, 1-8
  - snap, 1-7
- NI-IMAQ VIs
  - basic G IMAQ concepts, 1-1
  - high-level VIs, 2-2 to 2-11
    - IMAQ Close, 1-10, 2-9
    - IMAQ Get Camera Attribute, 1-16, 2-11
    - IMAQ Grab Acquire, 2-5
    - IMAQ Grab Setup, 2-4
    - IMAQ Init, 1-11, 2-8
    - IMAQ Sequence, 2-6 to 2-7
    - IMAQ Set Camera Attribute, 1-16, 2-10
    - IMAQ Snap, 1-7, 1-11, 1-12, 2-3

- overview, 1-9
- palette (figure), 2-2
- IMAQ Vision VIs, 2-35 to 2-39
  - IMAQ ColorImageToArray, 1-15, 2-39
  - IMAQ Create, 1-6, 1-11, 2-36
  - IMAQ Dispose, 2-37
  - IMAQ Dispose function, 1-6
  - IMAQ ImageToArray, 1-14, 2-38
  - palette (figure), 2-35
- location
  - in BridgeVIEW Functions palette (figure), 1-3
  - of examples, 1-2
  - in LabVIEW Functions palette (figure), 1-2
- low-level VIs, 2-25 to 2-34
  - IMAQ Configure Buffer, 1-9 to 1-10, 2-27
  - IMAQ Configure List, 1-9, 2-26
  - IMAQ Copy, 1-10, 2-31
  - IMAQ Extract Buffer, 1-10, 2-30
  - IMAQ Get Buffer, 1-10, 2-29
  - IMAQ Set User LUT, 2-34
  - IMAQ Start, 1-10, 2-28
  - IMAQ Status, 2-32
  - IMAQ Stop, 1-10, 2-33
  - overview, 1-9 to 1-10
  - palette (figure), 2-25
- overview, 1-1, 2-1
- signal I/O VIs, 2-12 to 2-24
  - IMAQ Configure Trigger, 1-12 to 1-13, 2-13 to 2-14
  - IMAQ Generate Pulse, 2-18 to 2-20
  - IMAQ Occurrence Config, 2-23 to 2-24
  - IMAQ Trigger Drive, 2-15 to 2-16
  - IMAQ Trigger Read, 2-17
  - IMAQ Wait Signal, 2-21 to 2-22
  - palette (figure), 2-12
- Simple Error Handler VI, 1-5

## O

- online problem-solving and diagnostic resources, C-1

## P

- parameters, common, 1-4
- Property Node. *See* IMAQ Property Node.
- pulse generation. *See* IMAQ Generate Pulse VI.

## R

- Region of Interest input, 1-4
- RGB snap, 1-12
- ring acquisition type, 1-9

## S

- sequence acquisition type, 1-8. *See also* IMAQ Sequence VI.
- session information properties, IMAQ Property Node, 3-11
- signal I/O VIs, 2-12 to 2-24
  - IMAQ Configure Trigger, 1-12 to 1-13, 2-13 to 2-14
  - IMAQ Generate Pulse, 2-18 to 2-20
  - IMAQ Occurrence Config, 2-23 to 2-24
  - IMAQ Trigger Drive, 2-15 to 2-16
  - IMAQ Trigger Read, 2-17
  - IMAQ Wait Signal, 2-21 to 2-22
  - palette (figure), 2-12
- Simple Error Handler VI, 1-5
- snap
  - composite snap, 1-10 to 1-11
  - RGB snap, 1-12
- snap acquisition type, 1-7. *See also* IMAQ Snap VI.
- software-related support resources, C-2
- status information properties, IMAQ Property Node, 3-11 to 3-12

- Step x input, 1-4
- Step y input, 1-4
- StillColor acquisition, 1-10 to 1-12
  - composite snap, 1-10 to 1-11
  - RGB snap, 1-12
- StillColor image representation, A-1 to A-3
- StillColor properties, IMAQ Property Node,  
3-12 to 3-14

## T

- technical support resources, C-1 to C-2
  - NI Web support, C-1 to C-2
  - worldwide support, C-2
- telephone support numbers, C-2
- triggering
  - IMAQ Configure Trigger VI,  
1-12 to 1-13, 2-13 to 2-14
  - IMAQ Trigger Drive VI, 2-15 to 2-16
  - IMAQ Trigger Read VI, 2-17
  - overview, 1-12 to 1-13

## W

- Web support for NI products, C-1 to C-2