



NI-IMAQ for IEEE 1394 Cameras Function Reference Help

March 2005 Edition, Part Number 370282D-01

NI-IMAQ for IEEE 1394 Cameras driver software gives you the ability to acquire images with industrial digital video cameras. This help file describes the functions included in NI-IMAQ for IEEE 1394 Cameras.

To navigate this help file, use the **Contents**, **Index**, and **Search** tabs to the left of this window.

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

[Conventions](#)—formatting and typographical conventions in this help file

[Important Information](#)


[Technical Support and Professional Services](#)

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

© 2001–2005 National Instruments Corporation. All rights reserved.

Conventions

This help file uses the following conventions:

- < > Angle brackets that contain numbers separated by an ellipsis represent a range of values associated with a bit or signal name—for example, DBIO<3..0>.
- » 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, emphasis, or an introduction to a key concept.
- green Underlined text in this color denotes a link to a help topic, help file, or Web address.
- 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.
- monospace bold** Bold text in this font denotes the messages and responses that the computer automatically prints to the screen. This font also emphasizes lines of code that are different from the other examples.

LabWindows/CVI Function Tree

The following table shows the LabWindows/CVI function panel that corresponds to each NI-IMAQ for IEEE 1394 Cameras function.

Class/Panel Name	Function Name	Description
Low Level Acquisition		
Start Acquisition	imaq1394StartAcquisition	Starts an acquisition synchronously or asynchronously.
Stop Acquisition	imaq1394StopAcquisition	Stops an asynchronous acquisition or synchronous continuous acquisition immediately.
Configure Acquisition	imaq1394ConfigureAcquisition	Configures the acquisition mode (continuous or one-shot).
Unconfigure Acquisition	imaq1394ClearAcquisition	Unconfigures an acquisition previously configured with imaq1394ConfigureAcquisition .
Get Image	imaq1394GetImage2	Copies a decoded frame buffer data to an IMAQ Vision image.
Get Buffer	imaq1394GetBuffer2	Copies a decoded frame buffer to a user-specified buffer.
Get Raw Data	imaq1394GetImageData	Copies raw data to a user-specified buffer.
Display		
Plot to Window	imaq1394Plot	Plots a buffer to a window given a handle to a window.
Plot to Device Context	imaq1394PlotDC	Plots a buffer to a window given a handle to a device context.
Camera Attributes		
Attribute Inquiry	imaq1394AttributeInquiry2	Queries the camera to check that it supports the specified attribute.
Get Attribute	imaq1394GetAttribute	Gets an attribute for a session.
Set Attribute	imaq1394SetAttribute	Sets an attribute for a session.
Get Video Modes	imaq1394GetVideoModes	Retrieves a list of video formats, modes, and frame rates supported by the camera.
Get Features	imaq1394GetFeatures2	Retrieves a list of features supported by the camera.
Trigger		

Events

Configure Event Message	imaq1394ConfigEventMessage	Configures the NI-IMAQ for IEEE 1394 Cameras driver to send a message to your application window when an event occurs.
Install Asynchronous Callback	imaq1394InstallCallback	Configures the NI-IMAQ for IEEE 1394 Cameras driver to execute a callback function when an event occurs.
Trigger Configure	imaq1394TriggerConfigure	Configures an acquisition to start based on an external trigger.

High Level Acquisition

Grab	imaq1394Grab2	Acquires the most current frame into a previously allocated buffer. Call this function only after calling imaq1394SetupGrab .
Setup Sequence	imaq1394SetupSequence	Configures and starts a session for acquiring a full sequence into the list of previously allocated buffers.
Snap	imaq1394Snap	Performs a single frame acquisition in a memory buffer into a previously allocated buffer.
Grab Image	imaq1394GrabImage2	Acquires the most current frame into the specified IMAQ Vision image buffer. Call this function only after calling imaq1394SetupGrab .
Setup Grab	imaq1394SetupGrab	Configures and starts a continuous acquisition.
Setup Sequence Image	imaq1394SetupSequenceImage	Configures and starts a session for acquiring a full sequence into the list of buffers managed by IMAQ Vision.
Snap Image	imaq1394SnapImage	Performs a single frame acquisition in an image buffer, which is allocated using IMAQ Vision memory management.

Register Access

Read Quadlet	imaqReadQuadlet	Accesses registers on the camera and reads a 32-bit quadlet from the camera.
Write Quadlet	imaq1394WriteQuadlet	Accesses registers on the camera and writes a 32-bit quadlet from the camera.
Read Quadlet Array	imaq1394ReadBlock	Accesses registers on the camera and reads an array of contiguous 32-bit quadlets from the camera.
Write Quadlet Array	imaq1394WriteBlock	Accesses registers on the camera and writes an array of contiguous 32-bit quadlets from the camera.

Session

Open Camera	imaq1394CameraOpen2	Opens a session on a camera by name.
-------------	-------------------------------------	--------------------------------------

Session

Close Session [imaq1394Close](#) Closes a session and releases all acquisition resources.

Miscellaneous

Get Interface Files [imaq1394GetInterfaceFiles](#) Enumerates the interfaces currently on the host computer.

Save [imaq1394SaveBuffer](#) Saves a buffer of a session to disk in BMP, TIFF, or PNG format.

Show Error [imaq1394ShowError](#) Returns a null-terminated string describing the error code.

imaq1394AttributeInquiry2

Format

```
rval = imaq1394AttributeInquiry(SESSION_ID sessionId, unsigned long  
attribute, Feature2 *feature);
```

Purpose

Determines if the camera supports the specified attribute. If so, the function returns information about the attribute, including minimum and maximum values, and special operation modes.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
attribute	unsigned long	The attribute to query.
feature	Feature2 *	On return, populated with information about the camera feature being queried. Contains fields such as the minimum and maximum for both absolute and relative values.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394CameraOpen2


Format

```
rval = imaq1394CameraOpen2(char *name, unsigned long mode, SESSION_ID  
*sessionId);
```

Purpose

Opens a camera, queries the camera for its capabilities, loads a camera configuration file, and creates a unique reference to the camera. Use [IMAQ1394Close](#) when you are finished with the reference.

Parameters

Parameter	Type	Description
name	char *	<p>The name of the camera you want to open. name (cam0, cam1, ..., camN) must match the configuration file name you used to configure the camera in MAX. You can also open a camera using its 64-bit serial number (uuid:XXXXXXXXXXXXXXXXXX), where the number following uuid must be a 64-bit hexadecimal number representing the internal serial number of the camera.</p> <p> Note Specify "uuid:serial number in hexadecimal representation" for the camera name when opening in listening mode. The serial number must match the serial number used in MAX.</p>
mode	Unsigned long	<p>The control mode of the camera used during image broadcasting. Open a camera using IMG1394_CAMERA_MODE_CONTROLLER to actively configure and acquire image data. Open a camera on a different host or target computer using IMG1394_CAMERA_MODE_LISTENER to passively acquire image data from a session that was opened in controller mode.</p>
sessionId	SESSION_ID *	<p>On return, a valid Session ID.</p>

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394ClearAcquisition

Format

rval = imaq1394ClearAcquisition (SESSION_ID sessionId);

Purpose

Unconfigures an acquisition previously configured with [imaq1394ConfigureAcquisition](#).

Parameters

Parameter	Type	Description
<code>sessionId</code>	<code>SESSION_ID</code>	A valid Session ID, which you can obtain using imag1394CameraOpen2 .

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394Close

Format

rval = imaq1394Close (SESSION_ID sessionId);

Purpose

Stops an acquisition in progress, releases resources associated with the acquisition, and closes the specified IMAQ1394 Session.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394ConfigEventMessage

Format

```
rval = imaq1394ConfigEventMessage (SESSION_ID sessionID, int event,  
GUIHNDL windowHandle, int windowsMessageNumber, void *parameter);
```

Purpose

Configures the NI-IMAQ for IEEE 1394 Cameras driver to send a message to your application window when an event occurs. You must first register a callback function on a Windows message. The function you use to register the callback returns a Windows message used by the driver to notify your application. For more information, refer to the RegisterWinMsgCallback function in the following LabWindows/CVI menu:

Library»User Interface»Callback Functions»Windows Interrupt Support.

In the callback function, the **wParam** parameter contains the event that sent the message. To send a message to the main thread, pass the window handle returned by GetCVIWindowHandle. To send a message to another thread, you must have created at least one top-level panel in that thread. Pass the window handle for that panel to imaq1394ConfigEventMessage. You can obtain the window handle for the top-level panel by calling GetPanelAttribute with the ATTR_SYSTEM_WINDOW_HANDLE attribute.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
event	int	The event to monitor. The following events are valid: FRAME_DONE CAMERA_ATTACHED CAMERA_DETACHED
windowHandle	GUIHNDL	The handle of the window that receives messages from the NI-IMAQ for IEEE 1394 Cameras driver. Use the following functions to get the handle: <ul style="list-style-type: none">• GetCVIWindowHandle. The message is sent to the main thread.• GetPanelAttribute with the ATTR_SYSTEM_WINDOW_HANDLE attribute. The message is sent to the thread whose top-level panel handle was used with GetPanelAttribute.
windowsMessageNumber	int	The Windows message number returned by RegisterWinMsgCallback.
parameter	void *	A pointer to user-defined data passed to the event function.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#).

imaq1394ConfigureAcquisition

Format

rval = imaq1394ConfigureAcquisition (SESSION_ID sessionId, unsigned int continuous, unsigned int bufferCount, Rect rectangle);

Purpose

Configures a low-level acquisition previously opened with [imaq1394CameraOpen2](#). Specify the acquisition type with the **continuous** and **bufferCount** parameters.

Snap	Continuous = 0	Buffer Count = 1
Sequence	Continuous = 0	Buffer Count > 1
Grab	Continuous = 1	Buffer Count ³ 1

Use [imaq1394ClearAcquisition](#) to unconfigure the acquisition.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
continuous	unsigned int	Specifies whether the acquisition is continuous or one-shot.
bufferCount	unsigned int	For a one-shot acquisition, this parameter specifies the number of images to acquire. For a continuous acquisition, this parameter specifies the number of buffers the driver uses internally.
rectangle	Rect	The area to acquire. Set this parameter to IMAQ_NO_RECT to acquire the entire acquisition window.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GetAttribute

Format

```
rval = imaq1394GetAttribute(SESSION_ID sessionId, unsigned long attribute,  
void *value);
```

Purpose

Gets an attribute for a camera.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
attribute	unsigned long	The attribute whose value you want to obtain. Refer to Attributes for a list of attributes.
value	unsigned long * or double *	The value of the specified attributes when the function returns.

Parameter Discussion

attribute specifies the attribute whose value you want to obtain. In LabWindows/CVI, when you click the control or press <Enter>, <spacebar>, or <Ctrl-down arrow>, a dialog box opens containing a hierarchical list of the available attributes. Attributes whose values cannot be obtained are dimmed. You can access function help text for each attribute by double-clicking an attribute or by selecting the attribute and pressing <Enter>.

If the attribute shown in this ring control has named constants as valid values, you can bring up a list of them by moving to the **Attribute Value** control and pressing <ENTER>. If there is a value shown in the bottom half of the **Attribute Value** control, the corresponding constant is marked in the displayed list.

If the specified attribute is a camera attribute, the driver may return one of the following values:

- IMG1394_AUTOMODE: Allows the camera to always control the selected camera attribute.
- IMG1394_ONEPUSHMODE: Allows the camera to control the selected camera attribute once, and then return to the manual control state with adjusted values.
- IMG1394_OFFMODE: Disables the selected camera attribute.
- IMG1394_ABSOLUTEMODE: Sets the selected camera attribute to use 32-bit floating-point numbers that map to real-world units. In this mode, the driver refers to the absolute camera attribute for the control value.
- IMG1394_IGNOREMODE: Prevents the driver from accessing the selected camera attribute.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394GetBuffer2


Format

```
rval = imaq1394GetBuffer2 (SESSION_ID sessionID, unsigned long  
bufferNumberDesired, unsigned long *bufferNumberActual, unsigned long  
onOverwrite, void **buffer);
```

Purpose

Copies the specified frame into a previously allocated buffer. Call this function only after calling [imaq1394ConfigureAcquisition](#).

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394Camera
bufferNumberDesired	unsigned long	The cumulative buffer number for the image to retrieve. For continuation parameter to IMG1394_LASTBUFFER to get the next buffer, or to IMG1394_IMMEDIATEBUFFER to get the current buffer.
bufferNumberActual *	unsigned long	The actual cumulative buffer number of the image retrieved.
onOverwrite	unsigned long	The overwrite policy to follow if a buffer is overwritten during acquisition. <ul style="list-style-type: none">• Specify IMG1394_ONOVERWRITE_GET_OLDEST to get the oldest buffer instead of the overwritten buffer.• Specify IMG1394_ONOVERWRITE_FAIL to return an error if a buffer is overwritten.• Specify IMG1394_ONOVERWRITE_GET_NEWEST to get the newest buffer instead of the overwritten buffer.
		 Note IMG1394_ONOVERWRITE_GET_NEWEST is not implemented by NI-IMAQ for IEEE 1394 and is only presented to keep the API compatible with IMAQ 3.x.
buffer	void **	The buffer that contains the image when the function returns.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#).

imaq1394GetFeatures2

Format

```
rval = imaq1394GetFeatures2 (SESSION_ID sessionID, Feature2 *featureArray,  
unsigned long *featureArraySize);
```


Purpose

Retrieves the features supported by the camera. If you do not know in advance the number of features, complete the following steps:

1. Call this function with the **featureArray** parameter set to NULL. The necessary size is then stored in **featureArraySize**.
2. Allocate **featureArray** with the given size.
3. Call this function again using the previously allocated array.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
featureArray	Feature *	Contains a pointer to an array of feature structure elements in which the features supported by the camera are stored. Set this parameter to NULL to get the size needed by the array in the featureArraySize parameter.
featureArraySize	unsigned long *	Contains the size of the array used to store the features. If the user passes NULL as the featureArray parameter, this parameter contains the needed size.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GetImage2


Format

rval = imaq1394GetImage2 (SESSION_ID sessionID, unsigned long
bufferNumberDesired, unsigned long *bufferNumberActual, unsigned long
onOverwrite, Image *image);

Purpose

Acquires the specified frame into **Image**. Call this function only after calling [imaq1394ConfigureAcquisition](#). If the image type does not match the video format of the camera, the function changes the image type to a suitable format.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394Camera
bufferNumberDesired	unsigned long	The cumulative buffer number of the image to retrieve. For continuous set this parameter to IMG1394_LASTBUFFER to get the next buffer parameter to IMG1394_IMMEDIATEBUFFER to get the current buffer
bufferNumberActual	unsigned long *	On return, the actual cumulative buffer number of the image retrieved
onOverwrite	unsigned long	The overwrite policy to follow if a buffer is overwritten during acquisition <ul style="list-style-type: none">• Specify IMG1394_ONOVERWRITE_GET_OLDEST to get the oldest buffer instead of the overwritten buffer.• Specify IMG1394_ONOVERWRITE_FAIL to return an error if the requested buffer is overwritten.• Specify IMG1394_ONOVERWRITE_GET_NEWEST to get the newest valid buffer instead of the overwritten buffer.
		 Note The IMG1394_ONOVERWRITE_GET_NEXT policy is not implemented by NI-IMAQ for Cameras and is only presented to keep the API consistent with NI-IMAQ 3.x.
Image	Image *	The buffer that contains the IMAQ Vision image when the function returns

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GetImageData

Format

```
rval = imaq1394GetImageData (SESSION_ID sessionID, unsigned long  
bufferNumberDesired, unsigned long *bufferNumberActual, unsigned long  
onOverwrite, void **buffer);
```



Purpose

Acquires the raw image data of the specified frame into a previously allocated buffer. Call this function only after calling [imaq1394ConfigureAcquisition](#).



Note This function allows you to access raw image data. For many uncompressed formats like YUV or RGB, **buffer** is not compatible with IMAQ Vision functions. To use the IMAQ Vision functions, use [imaq1394GetImage2](#) or [imaq1394GetBuffer2](#) instead of this function.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394Camera
bufferNumberDesired	unsigned long	The cumulative buffer number of the image to retrieve. For continuation parameter to IMG1394_LASTBUFFER to get the next buffer or set to IMG1394_IMMEDIATEBUFFER to get the current buffer.
bufferNumberActual	unsigned long	On return, the actual cumulative buffer number of the image retrieved.
onOverwrite	unsigned long	The overwrite policy to follow if a buffer is overwritten during acquisition. <ul style="list-style-type: none">• Specify IMG1394_ONOVERWRITE_GET_OLDEST to get the oldest buffer instead of the overwritten buffer.• Specify IMG1394_ONOVERWRITE_FAIL to return an error if a buffer is overwritten.• Specify IMG1394_ONOVERWRITE_GET_NEWEST to get the newest buffer instead of the overwritten buffer.
		 Note IMG1394_ONOVERWRITE_GET_OLDEST is not implemented by NI-IMAQ for IEEE 1394 and is only presented to keep the API compatible with IMAQ 3.x.
buffer	void **	The buffer that contains the raw data for the image when the function returns.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GetInterfaceFiles

Format

```
rval = imaq1394GetInterfaceFiles (InterfaceFile *interfaceFileArray, unsigned  
long *interfaceFileArraySize);
```

Purpose

Returns a list of interface files currently on the host computer. If you do not know in advance the number of interfaces, complete the following steps:

1. Call this function with the **interfaceFileArray** parameter set to NULL. The necessary size is then stored in **interfaceFileArraySize**.
2. Allocate **interfaceFileArray** with the given size.
3. Call this function again using the previously allocated array.

Parameters

Parameter	Type	Description
interfaceFileArray	Interface File *	Contains a pointer to an array of interface files that are on the host computer. This includes cameras that are currently connected or that have been connected in the past. Set this parameter to NULL to get the size needed by the array in the interfaceFileArraySize parameter.
interfaceFileArraySize	unsigned long *	The size of the array used to store the interface files. If the user passes NULL as the interfaceFileArray parameter, this parameter contains the needed size.

Parameter Discussion

The interface file structure contains information about currently and previously connected interfaces. Once enumerated, check the `Flags` member of the **InterfaceFile** structure. If the value of `Flags` is 0, the camera is not currently connected. If the value of `Flags` is 1, the camera is currently connected.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GetVideoModes

Format

```
rval = imaq1394GetVideoModes (SESSION_ID sessionID, VideoMode  
*videoModeArray, unsigned long *videoModeArraySize, unsigned long  
*currentMode);
```

Purpose

Returns a list of video formats, modes, and frame rates supported by the camera. If the number of video modes is not known in advance, complete the following steps:

1. Call this function with the **videoModeArray** parameter set to NULL. The necessary size is then stored in **videoModeArraySize**.
2. Allocate the **videoModeArray** with the given size.
3. Call this function again using with the previously allocated array.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
videoModeArray	VideoMode *	Contains a pointer to an array of video modes supported by the current camera. Set this parameter to NULL to get the size needed by the array in the videoModeArraySize parameter. VideoMode contains information about the name of the video mode (size, representation, and speed) and the format, mode, and frame rate parameters as defined in the <i>IIDC 1394-based Digital Camera Specification</i> .
videoModeArraySize	unsigned long *	The size of the array used to store the video modes. If the user passes NULL as the videoModeArray parameter, this parameter then contains the needed size.
currentMode	unsigned long *	The index into the videoModeArray of the current mode used by the camera.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#).

imaq1394Grab2

Format

```
rval = imaq1394Grab2 (SESSION_ID sessionId, unsigned long  
waitForNextBuffer, unsigned long *bufferNumberActual, void **buffer);
```

Purpose

Acquires the most current frame into a previously allocated buffer. Call this function only after calling [imaq1394SetupGrab](#).

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
waitForNextBuffer	unsigned long	Specifies whether the function returns the image currently being acquired or the last completely acquired image. Specify TRUE (1) to wait for a buffer. Specify FALSE (0) to get the current buffer in memory. The default value is TRUE.
bufferNumberActual	unsigned long	On return, the actual cumulative buffer number of the image retrieved.
buffer	void **	The buffer that will contain the image when the function returns.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GrabImage2

Format

```
rval = imaq1394GrabImage2 (SESSION_ID sessionId, unsigned long  
waitForNextBuffer, unsigned long *bufferNumberActual, Image *image);
```

Purpose

Acquires the most current frame into the specified IMAQ Vision image buffer. Call this function only after calling [imaq1394SetupGrab](#). If the image type does not match the video format of the camera, this function changes the image type to a suitable format.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
waitForNextBuffer	unsigned long	Specifies whether the function returns the image currently being acquired or the last completely acquired image. Specify TRUE (1) to wait for a buffer. Specify FALSE (0) to get the current buffer in memory. The default value is TRUE.
bufferNumberActual	unsigned long *	On return, the buffer number of the image returned.
Image	Image *	The buffer that contains the IMAQ Vision image when the function returns.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394InstallCallback

Format

```
rval = imaq1394InstallCallback (SESSION_ID sessionId, int event,  
IMAQ1394_CALL_BACK_PTR callBackFunction, void *parameter);
```

Purpose

Configures the NI-IMAQ for IEEE 1394 Cameras driver to execute a callback function when an event occurs.



Note Because the callback executes in a different thread, make sure that the code inside the callback is thread safe.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
event	int	The event to monitor. The following events are valid: <ul style="list-style-type: none">- IMG1394_EVENT_FRAME_DONE: Callback fired when a frame is transferred from the camera into system memory.- IMG1394_EVENT_CAMERA_ATTACHED: Callback fired when a new camera is attached.- IMG1394_EVENT_CAMERA_DETACHED: Callback fired when the camera is detached.
callbackFunction	IMAQ1394_CALL_BACK_PTR	The address of the callback function.
parameter	void *	A pointer to user-defined data passed to the event function.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394Plot

Format

rval = imaq1394Plot (GUIHNDL window_handle, void *buffer, unsigned long srcX, unsigned long srcY, unsigned long cols, unsigned long rows, unsigned long dstX, unsigned Long dstY, unsigned long flags);

Purpose

Plots a buffer to a window given a native Windows handle. Use this function to easily display a buffer after it is acquired.

Parameters

Parameter	Type	Description
window_handle	GUIHNDL	A native window handle designating the window in which to plot the image.
buffer	void *	A pointer to an area of memory containing a video frame buffer.
srcX	unsigned long	The left offset in the buffer to start plotting.
srcY	unsigned long	The top offset in the buffer to start plotting.
cols	unsigned long	The number of columns in the image.
rows	unsigned long	The number of rows in the image.
dstX	unsigned long	The left position to start plotting in the window.
dstY	unsigned long	The top position to start plotting in the window.
flags	unsigned long	Sets the display property with the following constants: <ul style="list-style-type: none">• IMG1394_PLOT_INVERT: Invert the picture• IMG1394_PLOT_COLOR_RGB24: Display a RGB 24-bit color image• IMG1394_PLOT_COLOR_RGB32: Display a RGB 32-bit color image• IMG1394_PLOT_MONO_8: Display an 8-bit grayscale image• IMG1394_PLOT_MONO_10: Display a 10-bit grayscale image• IMG1394_PLOT_MONO_12: Display a 12-bit grayscale image• IMG1394_PLOT_MONO_14: Display a 14-bit grayscale image• IMG1394_PLOT_MONO_16: Display a 16-bit grayscale image• IMG1394_PLOT_MONO_32: Display a 32-bit grayscale image

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394PlotDC

Format

rval = imaq1394PlotDC (GUIHNDL deviceContext, void *buffer, unsigned long srcX, unsigned long srcY, unsigned long cols, unsigned long rows, unsigned long dstX, unsigned Long dstY, unsigned long flags);

Purpose

Plots a buffer to a window given a device context. Use this function to easily display a buffer after it is acquired.

Parameters

Parameter	Type	Description
deviceContext	GUIHNDL	A device context in which to plot the image.
buffer	void *	A pointer to an area of memory containing a video frame buffer.
srcX	unsigned long	The left offset in the buffer to start plotting.
srcY	unsigned long	The top offset in the buffer to start plotting.
cols	unsigned long	The number of columns in the image.
rows	unsigned long	The number of rows in the image.
dstX	unsigned long	The left position to start plotting in the window.
dstY	unsigned long	The top position to start plotting in the window.
flags	unsigned long	The display property with the following constants: <ul style="list-style-type: none">• IMG1394_PLOT_INVERT: Invert the picture• IMG1394_PLOT_COLOR_RGB24: Display a RGB 24-bit color image• IMG1394_PLOT_COLOR_RGB32: Display a RGB 32-bit color image• IMG1394_PLOT_MONO_8: Display an 8-bit grayscale image• IMG1394_PLOT_MONO_10: Display a 10-bit grayscale image• IMG1394_PLOT_MONO_12: Display a 12-bit grayscale image• IMG1394_PLOT_MONO_14: Display a 14-bit grayscale image• IMG1394_PLOT_MONO_16: Display a 16-bit grayscale image• IMG1394_PLOT_MONO_32: Display a 32-bit grayscale image

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394ReadBlock

Format

```
rval = imaq1394ReadBlock(SESSION_ID sessionId, unsigned long offset,  
unsigned long count, unsigned long *array);
```

Purpose

Accesses registers on the camera and reads an array of contiguous 32-bit quadlets from the camera. Data is byte-swapped for little endian alignment after transfer.

Parameters

Parameter	Type	Description
sessionId	SESSION_Id	A valid Session ID, which you can obtain using img1394CameraOpen2 .
Offset	unsigned long	The register location to access. Use IMG1394_ATTR_BASE_ADDRESS to obtain the base register for the camera. Refer to the camera documentation for more information about camera-specific register ranges.
Count	unsigned long	The number of 32-bit quadlets to read.
Array	unsigned long *	On return, the array of 32-bit quadlets returned from the camera.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394ReadQuadlet

Format

```
rval = imaq1394ReadQuadlet(SESSION_ID sessionId, unsigned long offset,  
unsigned long *value);
```

Purpose

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

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
Offset	unsigned long	The offset from the base register to access. Use IMG1394_ATTR_BASE_ADDRESS to obtain the base register for the camera.
Value	unsigned long *	On return, the quadlet value of the specified offset.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394SaveBuffer

Format

```
rval = imaq1394SaveBuffer (SESSION_ID sessionId, void *buffer, char* filename);
```

Purpose

Saves a single image to disk. This function takes a buffer as an input.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
buffer	void *	The buffer that contains the image.
filename	char *	The filename to which you want to save the buffer. Supported formats are BMP, PNG, and JPG.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394SetAttribute

Format

rval = imaq1394SetAttribute (SESSION_ID sessionId, unsigned long attribute, unsigned long value);

Purpose

Sets an attribute for a camera.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imag1394CameraOpen2 .
attribute	unsigned long	The attribute whose value you want to set. Refer to Attributes for a list of attributes.
value	unsigned long (passed by value) or double (passed by reference)	The value of the specified attribute. Use imag1394AttributeInquiry2 to enumerate the special modes that the camera supports.

Parameter Discussion

attribute specifies the attribute whose value you want to set. In LabWindows/CVI, when you click the control or press <Enter>, <spacebar>, or <Ctrl-down arrow>, a dialog box opens containing a hierarchical list of the available attributes. Attributes whose values cannot be obtained are dimmed. Help text is shown for each attribute. Select an attribute by double-clicking it or by selecting it and then pressing <Enter>.

If the attribute shown in this ring control has named constants as valid values, you can bring up a list of them by moving to the **Attribute Value** control and pressing <ENTER>. If there is a value shown in the bottom half of the **Attribute Value** control, the corresponding constant is marked in the displayed list.

If the specified attribute is a camera attribute, the user may set one of the following values:

- IMG1394_AUTOMODE: Allows the camera to always control the selected camera attribute.
- IMG1394_ONEPUSHMODE: Allows the camera to control the selected camera attribute once, and then return to the manual control state with adjusted values.
- IMG1394_OFFMODE: Disables the selected camera attribute.
- IMG1394_ABSOLUTEMODE: Sets the selected camera attribute to use 32-bit floating-point numbers that map to real-world units. In this mode, the driver refers to the absolute camera attribute for the control value.
- IMG1394_IGNOREMODE: Prevents the driver from writing the selected camera attribute to the camera.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#) or, if you are using Microsoft Visual Basic, [`imaq1394ShowErrorCW`](#).

imaq1394SetupGrab

Format

rval = imaq1394SetupGrab (SESSION_ID sessionId, Rect rectangle);

Purpose

Configures and starts a grab acquisition. A grab performs an acquisition that loops continually on a ring of 3 buffers. Use a grab for high-speed image acquisition. Use [imaq1394Grab2](#) to copy an image out of the buffer ring. If you call this function before calling [imaq1394CameraOpen2](#), `imaq1394SetupGrab` uses `cam0` by default. Use [imaq1394ClearAcquisition](#) to unconfigure the acquisition.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
rectangle	Rect	The area of the acquisition window to acquire. Set this parameter to IMAQ_NO_RECT to acquire the entire acquisition window.



Note If your camera supports Partial Image Size Format (Format 7), you can use that setting to change the size of the image transferred over the 1394 bus. These values are coerced to the next highest multiple of the unit width or height supported by your camera.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394SetupSequence

Format

```
rval = imaq1394SetupSequence (SESSION_ID sessionID, void **buffers,  
unsigned long bufferCount, int skipCount, Rect rectangle);
```

Purpose

Configures and starts a session for acquiring a full sequence in the buffer list. A sequence acquisition acquires a full sequence of images into the image array. The acquisition is started automatically, and the function returns after the sequence has been acquired.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
buffers	void **	An array of buffers. Each element in the array must be a pointer to a valid buffer.
bufferCount	unsigned long	The number of buffers in the buffers array.
skipCount	int	Reserved for future use. Set to 0.
rectangle	Rect	The area of the acquisition window to acquire. Set this parameter to IMAQ_NO_RECT to acquire the entire acquisition window.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#).

imaq1394SetupSequenceImage

Format

```
rval = imaq1394SetupSequenceImage (SESSION_ID sessionID, Image  
**images, unsigned long bufferCount, int skipCount, Rect rectangle);
```

Purpose

Configures, starts, acquires, stops, and unconfigures a sequence acquisition. Use this function to capture multiple images. If you call this function before calling [imaq1394CameraOpen2](#), `imaq1394SetupSequenceImage` uses `cam0` by default.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
images	Image **	An array of images. Each element in the array must be a pointer to a valid image.
bufferCount	unsigned long	The number of buffers in the images array.
skipCount	int	Reserved for future use. Set to 0.
rectangle	Rect	The area of the acquisition window to acquire. Set this parameter to IMAQ_NO_RECT to acquire the entire acquisition window.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394ShowError

Format

```
rval = imaq1394ShowError (IMAQ_ERR errorCode, char *textMessage,  
unsigned long textLength);
```

Purpose

Returns a NULL terminated string describing the error code.

Parameters

Parameter	Type	Description
errorCode	IMG_ERR	A valid NI-IMAQ for IEEE 1394 Cameras error code. Refer to the Error Codes topic in this help file for a complete error code list.
textMessage	char *	A pointer to an area of memory reserved for an error string. The reserved memory must be at least the size specified by the textLength parameter.
textLength	unsigned long	The size of the C string passed as the textMessage parameter.

Return Value

Refer to [Error Codes](#) for a complete error code list.

imaq1394Snap

Format

rval = imaq1394Snap (SESSION_ID sessionId, void *buffer, Rect rectangle);

Purpose

Configures, starts, acquires, stops, and unconfigures a snap acquisition. Use a snap for low-speed or single-capture applications where ease of programming is essential. If you call this function before calling [imaq1394CameraOpen2](#), imaq1394Snap uses cam0 by default.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
buffer	void *	On return, the buffer that contains the image.
rectangle	Rect	The area of the acquisition window to acquire. Set this parameter to IMAQ_NO_RECT to acquire the entire acquisition window.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394SnapImage

Format

```
rval = imaq1394SnapImage (SESSION_ID sessionId, Image *image, Rect  
rectangle);
```

Purpose

Configures, starts, acquires, stops, and unconfigures a snap acquisition. Use a snap for low-speed or single-capture applications where ease of programming is essential. If you call this function before calling [`imaq1394CameraOpen2`](#), `imaq1394SnapImage` uses `cam0` by default. If the image type does not match the video format of the camera, this function changes the image type to a suitable format.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
image	Image *	The image into which to acquire. Use the IMAQ Vision function <code>imaqCreateImage</code> to create an image.
rectangle	Rect	The area to acquire. Set this parameter to <code>IMAQ_NO_RECT</code> to acquire the entire acquisition window.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394StartAcquisition

Format

rval = imaq1394StartAcquisition (SESSION_ID sessionID);

Purpose

Starts an acquisition that was previously configured with [imaq1394ConfigureAcquisition](#). Use [imaq1394StopAcquisition](#) to stop the acquisition.

Parameters

Parameter	Type	Description
<code>sessionId</code>	<code>SESSION_ID</code>	A valid Session ID, which you can obtain using imag1394CameraOpen2 .

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#) or, if you are using Microsoft Visual Basic, [`imaq1394ShowErrorCW`](#).

imaq1394StopAcquisition

Format

rval = imaq1394StopAcquisition (SESSION_ID sessionID);

Purpose

Stops an acquisition previously started with [imaq1394StartAcquisition](#).

Parameters

Parameter	Type	Description
<code>sessionId</code>	<code>SESSION_ID</code>	A valid Session ID, which you can obtain using imag1394CameraOpen2 .

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394TriggerConfigure

Format

rval = imaq1394TriggerConfigure (SESSION_ID sessionID, unsigned long polarity, unsigned long timeOut, unsigned long mode, unsigned long parameter);

Purpose

Configures the trigger conditions for an acquisition. Use this function before any acquisition function to set up a triggered image acquisition.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
polarity	unsigned long	The polarity of the trigger line. The following polarities are valid: <ul style="list-style-type: none">– IMG1394_TRIG_POLAR_ACTIVEL—Triggers on a falling edge.– IMG1394_TRIG_POLAR_ACTIVEH—Triggers on a rising edge.– IMG1394_TRIG_POLAR_DEFAULT—Trigger on a rising edge.
timeOut	unsigned long	The time, in milliseconds, allowed for the image acquisition to complete. If the acquisition cannot complete in time, an error returns.
mode	unsigned long	The trigger mode.
parameter	unsigned long	An optional parameter that defines <i>N</i> in trigger modes 2 and 3. Refer to the parameter discussion for more information.

Parameter Discussion

mode specifies the trigger mode. The following descriptions are valid for low-active polarity. All states are inverted for high-active polarity.

- **IMG1394_TRIG_DISABLED**—No trigger. The camera starts integration immediately.
- **IMG1394_TRIG_MODE0**—The camera starts integration on every falling edge of the external trigger input. Integration time is defined in the shutter attribute.
- **IMG1394_TRIG_MODE1**—The camera starts integration on every falling edge of the external trigger input. Integration time is equal to the low state of the external trigger.
- **IMG1394_TRIG_MODE2**—The camera starts integration on the first falling edge of the external trigger input. At the *N*th external trigger input falling edge, integration stops. *N* is defined in **parameter**.
- **IMG1394_TRIG_MODE3**—This is the internal trigger mode. The camera issues a trigger internally. The cycle time is *N* times the cycle time of the fastest frame rate. *N* is defined in **parameter**. The integration time is specified with the shutter attribute.
- **IMG1394_TRIG_MODE4**—The camera starts frame integration when the external trigger input changes to an active value. Each frame is exposed for a duration specified in the shutter attribute. The number of frames is specified in **parameter**, which must have a value of 1 or more.
- **IMG1394_TRIG_MODE5**—The camera starts frame integration when the external trigger input changes to an active value. Each frame is exposed while the external trigger is active. The number of frames is specified in **parameter**, which must have a value of 1 or more.



Note Some cameras may not support every trigger mode.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394WriteBlock

Format

```
rval = imaq1394WriteBlock(SESSION_ID sessionId, unsigned long offset,  
unsigned long count, unsigned long *array);
```

Purpose

Accesses registers on the camera and writes an array of contiguous 32-bit quadlets to the camera. Data is byte-swapped for big endian alignment before transfer.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
offset	unsigned long	The offset register location to access. Use IMG1394_ATTR_BASE_ADDRESS to obtain the base register for the camera. Refer to the camera documentation for more information about camera-specific register ranges.
count	unsigned long	The number of 32-bit quadlets to write.
array	unsigned long *	The array of 32-bit quadlets to write at the specified offset.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394WriteQuadlet

Format

rval = imaq1394WriteQuadlet(SESSION_ID sessionId, unsigned long offset, unsigned long value);

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

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
offset	unsigned long	The offset register location to access. Use IMG1394_ATTR_BASE_ADDRESS to obtain the base register for the camera. Refer to the camera documentation for more information about camera-specific register ranges.
value	unsigned long	The quadlet to write to the specified offset.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or, if you are using Microsoft Visual Basic, [img1394ShowErrorCW](#).

imaq1394AttributeInquiry2CW

Format

imaq1394AttributeInquiry2CW(sessionId As SESSION_ID, Attribute As Attribute, Feature As Feature2) As IMG_ERR

Purpose

Determines if the camera supports the specified attribute. If so, the function returns information about the attribute, including minimum and maximum values, and special operation modes.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
attribute	Attribute	The Attribute you want to query. Refer to Attributes for a list of attributes.
feature	Feature2 (passed by reference)	Information about the camera feature being queried. Contains fields such as the minimum and maximum for both absolute and relative values.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394ConfigureAcquisitionCW

Format

imaq1394ConfigureAcquisitionCW (sessionId As SESSION_ID, continuous As Long, bufferCount As Long) As IMG_ERR

Purpose

Configures a low-level acquisition previously opened with [imaq1394CameraOpen2](#). The acquisition type is specified with the parameters **continuous** and **bufferCount**.

Snap	Continuous = 0	Buffer Count = 1
Sequence	Continuous = 0	Buffer Count > 1
Grab	Continuous = 1	Buffer Count ³ 1

Use [imaq1394ClearAcquisition](#) to unconfigure the acquisition.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
continuous	long	Set this parameter to 1 for a continuous acquisition.
bufferCount	long	For a non-continuous acquisition, this parameter specifies the number of images to acquire. For a continuous acquisition, use this parameter to specify the number of buffers the driver uses internally.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394GetAttributeCW

Format

imaq1394GetAttributeCW (sessionId As SESSION_ID, Attribute As Attribute, value) As IMG_ERR

Purpose

Gets an attribute for a camera.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
attribute	Attribute	The Attribute whose value you want to obtain. Refer to Attributes for a list of attributes.
value	VARIANT (passed by reference)	The value of the specified attributes when the function returns.

Parameter Discussion

attribute specifies the attribute whose value you want to obtain. In LabWindows/CVI, when you click the control or press <Enter>, <spacebar>, or <Ctrl-down arrow>, a dialog box opens containing a hierarchical list of the available attributes. Attributes whose values cannot be obtained are dimmed. You can access function help text for each attribute by double-clicking an attribute or by selecting the attribute and pressing <Enter>.

If the attribute shown in this ring control has named constants as valid values, you can bring up a list of them by moving to the **Attribute Value** control and pressing <ENTER>. If there is a value shown in the bottom half of the **Attribute Value** control, the corresponding constant is marked in the displayed list.

If the specified attribute is a camera attribute, the driver may return one of the following values:

- IMG1394_AUTOMODE: Allows the camera to always control the selected camera attribute.
- IMG1394_ONEPUSHMODE: Allows the camera to control the selected camera attribute once, and then return to the manual control state with adjusted values.
- IMG1394_OFFMODE: Disables the selected camera attribute.
- IMG1394_ABSOLUTEMODE: Sets the selected camera attribute to use 32-bit floating-point numbers that map to real-world units. In this mode, the driver refers to the absolute camera attribute for the control value.
- IMG1394_IGNOREMODE: Prevents the driver from accessing the selected camera attribute.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394GetFeatures2CW

Format

imaq1394GetFeatures2CW(sessionId As SESSION_ID, psaFeatureArray() As Feature2) As IMG_ERR

Purpose

Retrieves the features supported by the camera.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
psaFeatureArray	SAFEARRAY (Feature2)	An array of Feature2 structure elements in which the features supported by the camera are stored.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowErrorCW`](#).

imaq1394GetImageCW


Format

imaq1394GetImageCW (sessionID As SESSION_ID, bufferNumberDesired As Long, bufferNumberActual As Long, onOverwrite As OverwritePolicy, dispatch As Object) As IMG_ERR

Purpose

Acquires the specified frame into **Image**. Call this function only after calling [imaq1394ConfigureAcquisition](#). If the image type does not match the video format of the camera, this function changes the image type to a suitable format.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394Cam
bufferNumberDesired	long	The cumulative buffer number of the image to retrieve. For con set this parameter to IMG1394_LASTBUFFER to get the next I parameter to IMG_IMMEDIATEBUFFER to get the current buff
bufferNumberActual	long (passed by reference)	The actual cumulative buffer number of the image retrieved.
onOverwrite	OverwritePolicy	The overwrite policy to follow if a buffer is overwritten during ac <ul style="list-style-type: none">• Specify the IMG1394_ONOVERWRITE_GET_OLDES oldest valid buffer instead of the overwritten buffer.• Specify the IMG1394_ONOVERWRITE_FAIL policy to requested buffer is overwritten.• Specify the IMG1394_ONOVERWRITE_GET_NEWE most recent valid buffer instead of the overwritten buff
		 Note The IMG1394_ONOVERWRITE_GET_NEX policy is not implemented by NI-IMAQ for Cameras and is only presented to keep consistent with NI-IMAQ.
dispatch	Object	A valid CWIMAQImage.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394GetInterfaceFilesCW

Format

imaq1394GetInterfaceFilesCW (psaInterfaceFileArray() As InterfaceFile) As
IMG_ERR

Purpose

Enumerates the interfaces currently on the host computer.

Parameters

Parameter	Type	Description
psaInterfaceFileArray	SAFEARRAY (InterfaceFile)	An array of InterfaceFile structure elements in which the interfaces supported by the system are stored.

Parameter Discussion

The interface file structure contains information about currently and previously connected interfaces. Once enumerated, check the `Flags` member of the **InterfaceFile** structure. If the value of `Flags` is 0, the camera is not currently connected. If the value of `Flags` is 1, the camera is currently connected.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394GetVideoModesCW

Format

imaq1394GetVideoModesCW(sessionId As SESSION_ID,
psaVideoModeArray() As VideoMode, currentMode As Long) As IMG_ERR

Purpose

Retrieves the list of video formats, modes, and frame rates supported by the camera.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
psaVideoModeArray	SAFEARRAY (VideoMode)	An array of VideoMode structure elements in which the video modes supported by the camera are stored. Video mode contains information about the name of the video mode (size, representation, and speed) and the format, mode, and frame rate parameters as defined in the IEEE 1394-based digital camera specifications.
currentMode	Long (passed by reference)	The index of the current mode used by the camera in psaVideoModeArray .

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394Grab2CW

Format

imaq1394Grab2CW(sessionId As SESSION_ID, waitForNextBuffer As Long, actualBufferNumber As Long, dispatch As Object) As IMG_ERR

Purpose

Acquires the most current frame into a valid CWIMAQImage. Call this function only after calling [imaq1394SetupGrabCW](#).

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2.
waitForNextBuffer	long	Specifies whether the function returns the image currently being acquired or the last completely acquired image. Specify TRUE (1) to wait for a buffer. Specify FALSE (0) to get the current buffer in memory.
bufferNumberActual	long (passed by reference)	The actual cumulative buffer number of the image retrieved.
dispatch	Object	A valid CWIMAQImage.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394SetAttributeCW

Format

imaq1394SetAttributeCW (sessionId As SESSION_ID, Attribute As Attribute, value) As IMG_ERR

Purpose

Sets an attribute for a camera.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using img1394CameraOpen2 .
attribute	Attribute	The Attribute whose value you want to set. Refer to Attributes for a list of attributes.
value	variant	The value for the specified attribute. If the camera supports a special operation mode, specify one of the following constants as the value: IMG1394_AUTOMODE IMG1394_ONEPUSHMODE IMG1394_OFFMODE IMG1394_ABSOLUTEMODE IMG1394_IGNOREMODE Use img1394AttributeInquiry2CW to enumerate the special modes that the camera supports.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394SetupGrabCW

Format

imaq1394SetupGrabCW(sessionId As SESSION_ID) As IMG_ERR

Purpose

Configures and starts a grab acquisition. A grab performs an acquisition that loops continually on a ring of 3 buffers. Use [imaq1394Grab2CW](#) to copy an image out of the buffer. Use [imaq1394ClearAcquisition](#) to unconfigure the acquisition.

Parameters

Parameter	Type	Description
<code>sessionId</code>	<code>SESSION_ID</code>	A valid Session ID, which you can obtain using imag1394CameraOpen2 .

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394SetupSequenceCW

Format

imaq1394SetupSequenceCW(sessionId As SESSION_ID, dispatchArray() As Object, bufferCount As Long, skipCount As Long) As IMG_ERR

Purpose

Configures, starts, acquires, stops, and unconfigures a sequence acquisition. Use this function to capture multiple images. If you call this function before calling [imaq1394CameraOpen2](#), `imaq1394SetupSequenceImage` uses `cam0` by default.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imaq1394CameraOpen2 .
dispatchArray	SAFEARRAY (Object)	An valid CWIMAQImage array.
bufferCount	Long	The number of images in the CWIMAQImage array. This value must be less than or equal to the number of allocated images in the CWIMAQImage array.
skipCount	Long	Reserved for future use. Set to 0.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394ShowErrorCW

Format

imaq1394ShowErrorCW(errorCode As IMG_ERR, errorMessage As String) As
IMG_ERR

Purpose

Returns a string describing the error code.

Parameters

Parameter	Type	Description
errorCode	IMG_ERR	A valid NI-IMAQ error code. Refer to Error Codes for a complete error code list.
errorMessage	String	The string describing the error that occurred.

Return Value

Refer to [Error Codes](#) for a complete error code list.

imaq1394SnapCW

Format

imaq1394SnapCW(sessionId As SESSION_ID, dispatch As Object) As
IMG_ERR

Purpose

Configures, starts, acquires, stops, and unconfigures a snap acquisition. Use a snap for low-speed or single-capture applications where ease of programming is essential. If you call this function before calling [`imaq1394CameraOpen2`](#), `imaq1394SnapImage` uses `cam0` by default. If the image type does not match the video format of the camera, this function changes the image type to a suitable format.

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain using imag1394CameraOpen2 .
dispatch	Object	A valid CWIMAQImage.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

Obsolete

Obsolete functions are functions from a previous version of NI-IMAQ for IEEE 1394 Cameras that have been replaced by newer functions.

Though the current version of NI-IMAQ for IEEE 1394 Cameras still supports these functions, you should use the newer functions whenever possible.

Obsolete Functions

The following functions are obsolete.

Function Name

[imaq1394AttributeInquiry](#)

[imaq1394CameraOpen](#)

[imaq1394GetBuffer](#)

[imaq1394GetFeatures](#)

[imaq1394GetFeaturesCW](#)

[imaq1394GetImage](#)

[imaq1394Grab](#)

[imaq1394GrabCW](#)

[imaq1394GrabImage](#)

Attributes

You can look for attributes by attribute [name](#) or [code](#).

Attributes by Code

The following table, sorted by attribute code, describes the attributes you can use with the attribute functions.

Code	Attribute Name	Attribute	Type	
0x01	Brightness	IMG1394_ATTR_BRIGHTNESS	int	Sets/gets th
0x02	Auto Exposure	IMG1394_ATTR_AUTO_EXPOSURE	int	Sets/gets r camera to a exposure le is controller Shutter attr
0x03	Sharpness	IMG1394_ATTR_SHARPNESS	int	Sets/gets th
0x04	White Balance U	IMG1394_ATTR_WHITE_BALANCE_U_B	int	Sets/gets th component
0x05	White Balance V	IMG1394_ATTR_WHITE_BALANCE_V_R	int	Sets/gets th component
0x06	Hue	IMG1394_ATTR_HUE	int	Sets/gets th
0x07	Saturation	IMG1394_ATTR_SATURATION	int	Sets/gets th
0x08	Gamma	IMG1394_ATTR_GAMMA	int	Defines the light level a
0x09	Shutter	IMG1394_ATTR_SHUTTER	int	Sets/gets th incoming liq
0x0A	Gain	IMG1394_ATTR_GAIN	int	Controls th
0x0B	Iris	IMG1394_ATTR_IRIS	int	Controls th
0x0C	Focus	IMG1394_ATTR_FOCUS	int	Controls th
0x0D	Temperature	IMG1394_ATTR_TEMPERATURE	int	Gets the te and/or cont
0x0F	Zoom	IMG1394_ATTR_ZOOM	int	Controls th
0x10	Pan	IMG1394_ATTR_PAN	int	Controls th
0x11	Tilt	IMG1394_ATTR_TILT	int	Controls th
0x12	Optical Filter	IMG1394_ATTR_OPTICAL_FILTER	int	Changes th function.
0x13	Vendor Name	IMG1394_ATTR_VENDOR_NAME	string	Read only. NULL-term a user-alloc to hold the
0x14	Model Name	IMG1394_ATTR_MODEL_NAME	string	Read only. NULL-term a user-alloc to hold the
0x15	Camera Serial Number	IMG1394_ATTR_SERIAL_NO	string	Read only.
0x16	Camera Model	IMG1394_ATTR_MODEL_ID	string	Read only.

	Number			
0x17	Video Format	IMG1394_ATTR_VIDEO_FORMAT	int	Sets/gets the value between img1394C supported formats
0x18	Video Mode	IMG1394_ATTR_VIDEO_MODE	int	Sets/gets the value between 0 and img1394C supported modes
0x19	Video Frame Rate	IMG1394_ATTR_VIDEO_FRAME_RATE	int	Sets/gets the value between img1394C supported rates
0x1A	Image Representation	IMG1394_ATTR_IMAGE_REP	int	Read only. for the acquisition are possible IMG1394_I IMG1394_I IMG1394_I IMG1394_I IMG1394_I IMG1394_I
0x1B	Time Out	IMG1394_ATTR_TIMEOUT	int	Sets/gets the time an acquisition could not be completed
0x1C	Nonstandard Unit Width	IMG1394_ATTR_FORMAT7_UNIT_WIDTH	int	Read only. only if the camera Size Format region of interest in the region of interest Width .
0x1D	Nonstandard Unit Height	IMG1394_ATTR_FORMAT7_UNIT_HEIGHT	int	Read only. only if the camera Size Format region of interest in the region of interest Height .
0x1E	Unique ID Low	IMG1394_ATTR_UNIQUE_ID_LOW	int	Read only. 16 bit unique ID camera assigned
0x1F	Unique ID High	IMG1394_ATTR_UNIQUE_ID_HIGH	int	Read only. 16 bit unique ID camera assigned
0x20	Number of lost buffers	IMG1394_ATTR_LOST_BUFFER_NB	int	Read only. number of buffers lost during an acquisition

0x21	Nonstandard Left Offset	IMG1394_ATTR_FORMAT7_LEFT	int	Sets/gets the left offset of the video window. Valid only if Partial Image Mode is enabled.
0x22	Nonstandard Top Offset	IMG1394_ATTR_FORMAT7_TOP	int	Sets/gets the top offset of the video window. Valid only if Partial Image Mode is enabled.
0x23	Nonstandard Width	IMG1394_ATTR_FORMAT7_WIDTH	int	Sets/gets the width of the video window. Valid only if Partial Image Mode is enabled.
0x24	Nonstandard Height	IMG1394_ATTR_FORMAT7_HEIGHT	int	Sets/gets the height of the video window. Valid only if Partial Image Mode is enabled.
0x25	Nonstandard Color Coding	IMG1394_ATTR_FORMAT7_COLORCODING	int	Sets/gets the current video color coding. Read-only if Partial Image Mode is enabled. Valid only if the video is following the current video color coding. IMG1394_ATTR_FORMAT7_COLORCODING_0 IMG1394_ATTR_FORMAT7_COLORCODING_1 IMG1394_ATTR_FORMAT7_COLORCODING_2 IMG1394_ATTR_FORMAT7_COLORCODING_3 IMG1394_ATTR_FORMAT7_COLORCODING_4 IMG1394_ATTR_FORMAT7_COLORCODING_5 IMG1394_ATTR_FORMAT7_COLORCODING_6 IMG1394_ATTR_FORMAT7_COLORCODING_7 IMG1394_ATTR_FORMAT7_COLORCODING_8 IMG1394_ATTR_FORMAT7_COLORCODING_9 IMG1394_ATTR_FORMAT7_COLORCODING_10 IMG1394_ATTR_FORMAT7_COLORCODING_11 IMG1394_ATTR_FORMAT7_COLORCODING_12 IMG1394_ATTR_FORMAT7_COLORCODING_13 IMG1394_ATTR_FORMAT7_COLORCODING_14 IMG1394_ATTR_FORMAT7_COLORCODING_15
0x26	Image Width	IMG1394_ATTR_IMAGE_WIDTH	int	Read only. The width of the image in the video window.
0x27	Image Height	IMG1394_ATTR_IMAGE_HEIGHT	int	Read only. The height of the image in the video acquisition.
0x28	Bytes per Pixel	IMG1394_ATTR_BYTES_PER_PIXEL	int	Read only. The number of bytes per pixel in the video.
0x29	Frame Interval	IMG1394_ATTR_FRAME_INTERVAL	int	Read only. The frame interval in milliseconds.
0x2A	Nonstandard Bytes Per Packet	IMG1394_ATTR_FORMAT7_BYTES_PER_PACKET	int	Sets/gets the number of bytes per packet in the video. Valid only if Partial Image Mode is enabled.

				Image Size
0x101	Absolute Brightness	IMG1394_ATTR_ABSOLUTE_BRIGHTNESS	Double	Sets/gets the value is rep
0x102	Absolute Auto Exposure	IMG1394_ATTR_ABSOLUTE_AUTO_EXPOSURE	Double	Sets/gets re camera to a exposure le relative to C
0x103	Absolute Sharpness	IMG1394_ATTR_ABSOLUTE_SHARPNESS	Double	Sets/gets the
0x104	Absolute White Balance	IMG1394_ATTR_ABSOLUTE_WHITE_BALANCE	Double	Sets/gets the of the imag Kelvin (K).
0x106	Absolute Hue	IMG1394_ATTR_ABSOLUTE_HUE	Double	Sets/gets the The value i
0x107	Absolute Saturation	IMG1394_ATTR_ABSOLUTE_SATURATION	Double	Sets/gets the The value i
0x108	Absolute Gamma	IMG1394_ATTR_ABSOLUTE_GAMMA	Double	Defines the light level a
0x109	Absolute Shutter	IMG1394_ATTR_ABSOLUTE_SHUTTER	Double	Sets/gets the incoming lig seconds.
0x10A	Absolute Gain	IMG1394_ATTR_ABSOLUTE_GAIN	Double	Controls the is represen
0x10B	Absolute Iris	IMG1394_ATTR_ABSOLUTE_IRIS	Double	Controls the value is rep
0x10C	Absolute Focus	IMG1394_ATTR_ABSOLUTE_FOCUS	Double	Controls the represente
0x10D	Absolute Temperature	IMG1394_ATTR_ABSOLUTE_TEMPERATURE	Double	Gets the te and/or cont
0x10F	Absolute Zoom	IMG1394_ATTR_ABSOLUTE_ZOOM	Double	Controls the represente wide end.
0x110	Absolute Pan	IMG1394_ATTR_ABSOLUTE_PAN	Double	Controls the represente
0x111	Absolute Tilt	IMG1394_ATTR_ABSOLUTE_TILT	Double	Controls the represente
0x112	Absolute Optical Filter	IMG1394_ATTR_ABSOLUTE_OPTICAL_FILTER	Double	Changes the functions.
0x11B	ROI Left	IMG1394_ATTR_ROI_LEFT	int	Sets/gets the interest. The subset of the
0x11C	ROI Top	IMG1394_ATTR_ROI_TOP	int	Sets/gets the interest. The subset of the
0x11D	ROI Width	IMG1394_ATTR_ROI_WIDTH	int	Sets/gets the interest. The subset of the

0x11E	ROI Height	IMG1394_ATTR_ROI_HEIGHT	int	Sets/gets the ROI height. The ROI is a subset of the image.
0x121	Bus Transfer Speed	IMG1394_ATTR_FORMAT7_SPEED	int	Sets/gets the bus transfer speed for acquisition. The camera does not support Format 7.
0x123	BufferNumber	IMG1394_ATTR_LAST_TRANSFERRED_BUFFER_NUM	int	Read only. The number of buffers transferred.
0x124	FrameCount	IMG1394_ATTR_FRAME_COUNT	int	Read only. The number of frames.
0x125	Acquisition in Progress indicator	IMG1394_ATTR_ACQ_IN_PROGRESS	int	Read only. TRUE if acquisition is in progress. FALSE otherwise.
0x126	Ignore First Frame	IMG1394_ATTR_IGNORE_FIRST_FRAME	int	Sets/gets the number of frames to ignore when starting acquisition.
0x127	Downshift 16-bit Images	IMG1394_ATTR_SHIFT_PIXEL_BITS	int	Sets/gets the number of bits to downshift the multibyte pixel data for acquisition.
0x128	Swap 16-bit Images Bytes	IMG1394_ATTR_SWAP_PIXEL_BYTES	int	Sets/gets the number of bytes to swap for the camera's 16-bit images.
0x129	Nonstandard Unit Bytes Per Packet	IMG1394_ATTR_FORMAT7_UNIT_BYTES_PER_PACKET	int	Read only. The number of bytes per packet transfer size for this value. The camera supports Packet Transfer (Format 7).
0x12A	Nonstandard Max Bytes Per Packet	IMG1394_ATTR_FORMAT7_MAX_BYTES_PER_PACKET	int	Read only. The maximum number of bytes per packet transfer size for this value. Valid for Packet Transfer (Format 7).
0x12C	Bits Per Pixel	IMG1394_ATTR_BITS_PER_PIXEL	int	Sets/gets the number of bits per pixel component. The camera supports 1-bit and 8-bit depth (1-bit).
0x202	Trigger Delay	IMG1394_ATTR_TRIGGER_DELAY	int	Sets/gets the trigger delay for incoming external frame exposure.
0x203	White Shading Compensation (Red)	IMG1394_ATTR_WHITE_SHADING_R	int	Sets/gets the white shading compensation for the red channel.
0x204	White Shading Compensation (Green)	IMG1394_ATTR_WHITE_SHADING_G	int	Sets/gets the white shading compensation for the green channel.
0x205	White Shading Compensation (Blue)	IMG1394_ATTR_WHITE_SHADING_B	int	Sets/gets the white shading compensation for the blue channel.
0x206	Frame Rate	IMG1394_ATTR_FRAME_RATE	int	Sets/gets the frame rate.

				current vide
0x207	Absolute Trigger Delay	IMG1394_ATTR_ABSOLUTE_TRIGGER_DELAY	Double	Sets/gets the incoming exposure frame exposure in millisecond
0x208	Absolute Frame Rate	IMG1394_ATTR_ABSOLUTE_FRAME_RATE	Double	Sets/gets the current video frame rate
0x209	Available Bayer Pattern	IMG1394_ATTR_COLOR_FILTER_INQ	int	Read only. Returns the available Bayer pattern to use
0x20A	Current Bayer Pattern	IMG1394_ATTR_COLOR_FILTER	int	Sets/gets the current Bayer pattern. The following values are supported: <img1394_attr_color_filter_0 </img1394_attr_color_filter_0 <img1394_attr_color_filter_1 </img1394_attr_color_filter_1 <img1394_attr_color_filter_2 </img1394_attr_color_filter_2 <img1394_attr_color_filter_3 </img1394_attr_color_filter_3 <img1394_attr_color_filter_4< td=""> </img1394_attr_color_filter_4<>
0x20B	Bayer Gain (Red)	IMG1394_ATTR_COLOR_FILTER_GAIN_R	Double	Sets/gets the Bayer gain for the red component. Valid value: 0.0-1.0
0x20C	Bayer Gain (Green)	IMG1394_ATTR_COLOR_FILTER_GAIN_G	Double	Sets/gets the Bayer gain for the green component. Valid value: 0.0-1.0
0x20D	Bayer Gain (Blue)	IMG1394_ATTR_COLOR_FILTER_GAIN_B	Double	Sets/gets the Bayer gain for the blue component. Valid value: 0.0-1.0
0x20E	Max Bus Transfer Speed	IMG1394_ATTR_FORMAT7_MAX_SPEED	int	Read only. Returns the maximum bus transfer speed for the camera supported by the current Format (Format 7)
0x21B	Packets Per Image	IMG1394_ATTR_FORMAT7_PACKETS_PER_IMAGE	int	Read only. Returns the number of packets transferred per image
0x21D	Base Address	IMG1394_ATTR_BASE_ADDRESS	int	Read only. Returns the base address of the camera

Attributes by Name

The following table, sorted by attribute name, describes the attributes you can use with the attribute functions.

Attribute Name	Attribute	Code	Type	
Absolute Auto Exposure	IMG1394_ATTR_ABSOLUTE_AUTO_EXPOSURE	0x102	Double	Sets/gets the camera to a exposure level relative to C
Absolute Brightness	IMG1394_ATTR_ABSOLUTE_BRIGHTNESS	0x101	Double	Sets/gets the value is rep
Absolute Focus	IMG1394_ATTR_ABSOLUTE_FOCUS	0x10C	Double	Controls the represente
Absolute Frame Rate	IMG1394_ATTR_ABSOLUTE_FRAME_RATE	0x208	Double	Sets/gets the current vide
Absolute Gain	IMG1394_ATTR_ABSOLUTE_GAIN	0x10A	Double	Controls the is represen
Absolute Gamma	IMG1394_ATTR_ABSOLUTE_GAMMA	0x108	Double	Defines the light level a
Absolute Hue	IMG1394_ATTR_ABSOLUTE_HUE	0x106	Double	Sets/gets the The value i
Absolute Iris	IMG1394_ATTR_ABSOLUTE_IRIS	0x10B	Double	Controls the value is rep
Absolute Optical Filter	IMG1394_ATTR_ABSOLUTE_OPTICAL_FILTER	0x112	Double	Changes the functions.
Absolute Pan	IMG1394_ATTR_ABSOLUTE_PAN	0x110	Double	Controls the represente
Absolute Saturation	IMG1394_ATTR_ABSOLUTE_SATURATION	0x107	Double	Sets/gets the The value i
Absolute Sharpness	IMG1394_ATTR_ABSOLUTE_SHARPNESS	0x103	Double	Sets/gets the
Absolute Shutter	IMG1394_ATTR_ABSOLUTE_SHUTTER	0x109	Double	Sets/gets the incoming liq seconds.
Absolute Temperature	IMG1394_ATTR_ABSOLUTE_TEMPERATURE	0x10D	Double	Gets the te and/or cont
Absolute Tilt	IMG1394_ATTR_ABSOLUTE_TILT	0x111	Double	Controls the represente
Absolute Trigger Delay	IMG1394_ATTR_ABSOLUTE_TRIGGER_DELAY	0x207	Double	Sets/gets the incoming e. frame expo millisecond
Absolute White Balance	IMG1394_ATTR_ABSOLUTE_WHITE_BALANCE	0x104	Double	Sets/gets the of the imag Kelvin (K).

Absolute Zoom	IMG1394_ATTR_ABSOLUTE_ZOOM	0x10F	Double	Controls the represente wide end.
Acquisition in Progress Indicator	IMG1394_ATTR_ACQ_IN_PROGRESS	0x125	int	Read only. acquisition. FALSE.
Auto Exposure	IMG1394_ATTR_AUTO_EXPOSURE	0x02	int	Sets/gets re camera to a exposure le is controller Shutter attr
Available Bayer Pattern	IMG1394_ATTR_COLOR_FILTER_INQ	0x209	int	Read only. pattern to u
Base Address	IMG1394_ATTR_BASE_ADDRESS	0x21D	int	Read only. the camera
Bayer Gain (Blue)	IMG1394_ATTR_COLOR_FILTER_GAIN_B	0x20D	Double	Sets/gets th blue compo
Bayer Gain (Green)	IMG1394_ATTR_COLOR_FILTER_GAIN_G	0x20C	Double	Sets/gets th green comp
Bayer Gain (Red)	IMG1394_ATTR_COLOR_FILTER_GAIN_R	0x20B	Double	Sets/gets th red compor
Bits Per Pixel	IMG1394_ATTR_BITS_PER_PIXEL	0x12C	int	Sets/gets th bit compon bit depth (1
Brightness	IMG1394_ATTR_BRIGHTNESS	0x01	int	Sets/gets th
BufferNumber	IMG1394_ATTR_LAST_TRANSFERRED_BUFFER_NUM	0x123	int	Read only. number tra
Bus Transfer Speed	IMG1394_ATTR_FORMAT7_SPEED	0x121	int	Sets/gets th acquisition. camera doe Size Forme
Bytes per Pixel	IMG1394_ATTR_BYTES_PER_PIXEL	0x28	int	Read only. for each pix
Camera Model Number	IMG1394_ATTR_MODEL_ID	0x16	string	Read only.
Camera Serial Number	IMG1394_ATTR_SERIAL_NO	0x15	string	Read only.
Current Bayer Pattern	IMG1394_ATTR_COLOR_FILTER	0x20A	int	Sets/gets th following va IMG1394_ IMG1394_ IMG1394_ IMG1394_ IMG1394_

Downshift 16-bit Images	IMG1394_ATTR_SHIFT_PIXEL_BITS	0x127	int	Sets/gets the number of multibyte pixel acquisitions.
Focus	IMG1394_ATTR_FOCUS	0x0C	int	Controls the focus.
FrameCount	IMG1394_ATTR_FRAME_COUNT	0x124	int	Read only. Number of frames.
Frame Interval	IMG1394_ATTR_FRAME_INTERVAL	0x29	int	Read only. Frame interval in milliseconds.
Frame Rate	IMG1394_ATTR_FRAME_RATE	0x206	int	Sets/gets the current video frame rate.
Gain	IMG1394_ATTR_GAIN	0x0A	int	Controls the gain.
Gamma	IMG1394_ATTR_GAMMA	0x08	int	Defines the gamma (light level adjustment).
Hue	IMG1394_ATTR_HUE	0x06	int	Sets/gets the hue.
Ignore First Frame	IMG1394_ATTR_IGNORE_FIRST_FRAME	0x126	int	Sets/gets the ignore first frame flag when starting an acquisition.
Image Height	IMG1394_ATTR_IMAGE_HEIGHT	0x27	int	Read only. Image height in pixels.
Image Representation	IMG1394_ATTR_IMAGE_REP	0x1A	int	Read only. Image representation for the acquisition. Possible values: IMG1394_I_RGB, IMG1394_I_YUV422, IMG1394_I_YUV420, IMG1394_I_MONO, IMG1394_I_MONO8, IMG1394_I_MONO16.
Image Width	IMG1394_ATTR_IMAGE_WIDTH	0x26	int	Read only. Image width in pixels.
Iris	IMG1394_ATTR_IRIS	0x0B	int	Controls the iris.
Max Bus Transfer Speed	IMG1394_ATTR_FORMAT7_MAX_SPEED	0x20E	int	Read only. Maximum bus transfer speed for the camera supported by the Format (Format 7).
Model Name	IMG1394_ATTR_MODEL_NAME	0x14	string	Read only. Model name. NULL-terminates a pointer to a large enough buffer to hold the camera model name.
Nonstandard Bytes Per Packet	IMG1394_ATTR_FORMAT7_BYTES_PER_PACKET	0x2A	int	Sets/gets the number of bytes per packet for isochronous transfer. Valid only if Image Size is set to 1.

Nonstandard Color Coding	IMG1394_ATTR_FORMAT7_COLORCODING	0x25	int	Sets/gets the current video read-only if Partial Image following value. IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING IMG1394_ATTR_FORMAT7_COLORCODING
Nonstandard Height	IMG1394_ATTR_FORMAT7_HEIGHT	0x24	int	Sets/gets the window. Value Partial Image
Nonstandard Left Offset	IMG1394_ATTR_FORMAT7_LEFT	0x21	int	Sets/gets the window. Value Partial Image
Nonstandard Max Bytes Per Packet	IMG1394_ATTR_FORMAT7_MAX_BYTES_PER_PACKET	0x12A	int	Read only. bytes per packet transfer size value. Valid Partial Image
Nonstandard Top Offset	IMG1394_ATTR_FORMAT7_TOP	0x22	int	Sets/gets the window. Value Partial Image
Nonstandard Unit Bytes Per Packet	IMG1394_ATTR_FORMAT7_UNIT_BYTES_PER_PACKET	0x129	int	Read only. bytes per packet transfer size this value. Value supports Partial Image (Format 7).
Nonstandard Unit Height	IMG1394_ATTR_FORMAT7_UNIT_HEIGHT	0x1D	int	Read only. only if the current Size Format region of in

				the region r Height.
Nonstandard Unit Width	IMG1394_ATTR_FORMAT7_UNIT_WIDTH	0x1C	int	Read only. only if the c Size Forma region of in the region r Width.
Nonstandard Width	IMG1394_ATTR_FORMAT7_WIDTH	0x23	int	Sets/gets th window. Va partial scar
Number of Lost Buffers	IMG1394_ATTR_LOST_BUFFER_NB	0x20	int	Read only. during an a
Optical Filter	IMG1394_ATTR_OPTICAL_FILTER	0x12	int	Changes th function.
Packets Per Image	IMG1394_ATTR_FORMAT7_PACKETS_PER_IMAGE	0x21B	int	Read only. transferred
Pan	IMG1394_ATTR_PAN	0x10	int	Controls th
ROI Height	IMG1394_ATTR_ROI_HEIGHT	0x11E	int	Sets/gets th interest. Th subset of th
ROI Left	IMG1394_ATTR_ROI_LEFT	0x11B	int	Sets/gets th interest. Th subset of th
ROI Top	IMG1394_ATTR_ROI_TOP	0x11C	int	Sets/gets th interest. Th subset of th
ROI Width	IMG1394_ATTR_ROI_WIDTH	0x11D	int	Sets/gets th interest. Th subset of th
Saturation	IMG1394_ATTR_SATURATION	0x07	int	Sets/gets th
Sharpness	IMG1394_ATTR_SHARPNESS	0x03	int	Sets/gets th
Shutter	IMG1394_ATTR_SHUTTER	0x09	int	Sets/gets th incoming liq
Swap 16-bit Images Bytes	IMG1394_ATTR_SWAP_PIXEL_BYTES	0x128	int	Sets/gets th of 16-bit ca the camera
Temperature	IMG1394_ATTR_TEMPERATURE	0x0D	int	Gets the te and/or cont
Tilt	IMG1394_ATTR_TILT	0x11	int	Controls th
Time Out	IMG1394_ATTR_TIMEOUT	0x1B	int	Sets/gets th an acquisiti could not b
Trigger Delay	IMG1394_ATTR_TRIGGER_DELAY	0x202	int	Sets/gets th incoming e. frame expo
Unique ID Low	IMG1394_ATTR_UNIQUE_ID_LOW	0x1E	int	Read only.

				bit unique r associated
Unique ID High	IMG1394_ATTR_UNIQUE_ID_HIGH	0x1F	int	Read only. bit unique r associated
Vendor Name	IMG1394_ATTR_VENDOR_NAME	0x13	string	Read only. NULL-term a pointer to large enoug the camera
Video Format	IMG1394_ATTR_VIDEO_FORMAT	0x17	int	Sets/gets th value betwe img1394C supported l
Video Frame Rate	IMG1394_ATTR_VIDEO_FRAME_RATE	0x19	int	Sets/gets th value betwe img1394C rate suppor
Video Mode	IMG1394_ATTR_VIDEO_MODE	0x18	int	Sets/gets th between 0 img1394C supported l
White Balance U	IMG1394_ATTR_WHITE_BALANCE_U_B	0x04	int	Sets/gets th component
White Balance V	IMG1394_ATTR_WHITE_BALANCE_V_R	0x05	int	Sets/gets th component
White Shading Compensation (Blue)	IMG1394_ATTR_WHITE_SHADING_B	0x205	int	Sets/gets th for the blue
White Shading Compensation (Green)	IMG1394_ATTR_WHITE_SHADING_G	0x204	int	Sets/gets th for the gree
White Shading Compensation (Red)	IMG1394_ATTR_WHITE_SHADING_R	0x203	int	Sets/gets th for the red
Zoom	IMG1394_ATTR_ZOOM	0x0F	int	Controls th

Error Codes

The following table describes the error codes used in NI-IMAQ for IEEE 1394 Cameras.

Error Code	Status Name	Description
0	IMG1394_ERR_GOOD=1	Success
-1074364416	IMG1394_ERR_EMEM	Not enough memory
-1074364415	IMG1394_ERR_EDRV	Cannot load the driver
-1074364414	IMG1394_ERR_TIMO	Timeout
-1074364413	IMG1394_ERR_NIMP	Function not implemented
-1074364412	IMG1394_ERR_INTL	Internal error
-1074364411	IMG1394_ERR_BMOD	Invalid mode
-1074364410	IMG1394_ERR_INIT	Session not initialized
-1074364409	IMG1394_ERR_BATT	Invalid attribute
-1074364408	IMG1394_ERR_FTNP	Feature not present in the camera
-1074364407	IMG1394_ERR_ESYS	System error
-1074364406	IMG1394_ERR_HEAP	Allocation error
-1074364405	IMG1394_ERR_UNINITIALIZED	Allocator not initialized
-1074364404	IMG1394_ERR_ORNG	Value out of range
-1074364403	IMG1394_ERR_BCAM	Invalid camera file
-1074364402	IMG1394_ERR_BSID	Invalid Session ID
-1074364401	IMG1394_ERR_NSUP	Attribute not supported by the camera
-1074364400	IMG1394_ERR_INVF	Invalid format
-1074364399	IMG1394_ERR_INVM	Invalid mode
-1074364398	IMG1394_ERR_INVR	Invalid frame rate
-1074364397	IMG1394_ERR_INVC	Invalid color ID
-1074364396	IMG1394_ERR_NOAP	No acquisition in progress
-1074364395	IMG1394_ERR_AOIP	Acquisition already in progress
-1074364394	IMG1394_ERR_IRES	Insufficient resources available for the required video mode
-1074364393	IMG1394_ERR_TBUF	Too many buffers used
-1074364392	IMG1394_ERR_INVP	Invalid parameter
-1074364391	IMG1394_ERR_NSAT	Non-writeable attribute
-1074364390	IMG1394_ERR_NGAT	Non-readable attribute
-1074364389	IMG1394_ERR_CMNF	Camera not found
-1074364388	IMG1394_ERR_CRMV	Camera removed

-1074364387	IMG1394_ERR_BNRD	Buffer not ready
-1074364386	IMG1394_ERR_BRST	Bus reset occurred during a transaction
-1074364385	IMG1394_ERR_NLIC	No license for NI-IMAQ for IEEE 1394 Cameras
-1074364384	IMG1394_ERR_NDLL	DLL could not be found (LabWindows/CVI only)
-1074364383	IMG1394_ERR_NFNC	Function not found in DLL (LabWindows/CVI only)
-1074364382	IMG1394_ERR_NOSR	No resource available (LabWindows/CVI only)
-1074364381	IMG1394_ERR_NCFG	Session not configured
-1074364380	IMG1394_ERR_IOER	I/O error
-1074364379	IMG1394_ERR_CAIU	Camera already in use
-1074364378	IMG1394_ERR_BAD_POINTER	Invalid pointer. The pointer may be NULL when it should be non-NULL, or non-NULL when it should be NULL.
-1074364377	IMG1394_EXCEPTION	Exception occurred. Refer to the NI-PAL debug log for more information.
-1074364376	IMG1394_ERR_BAD_DEVICE_TYPE	Invalid device type. Unable to create an instance.
-1074364375	IMG1394_ERR_ASYNC_READ	Unable to perform asynchronous register read. Camera may be busy or broken.
-1074364374	IMG1394_ERR_ASYNC_WRITE	Unable to perform asynchronous register write. Camera may be busy or broken.
-1074364373	IMG1394_ERR_VIDEO_NOT_SUPPORTED	Combination of video format, mode, and rate is not supported for this camera. Refer to your camera documentation.
-1074364372	IMG1394_ERR_BUFFER_INDEX	Index into the buffer list is incorrect. Reconfigure and try again.
-1074364371	IMG1394_ERR_BAD_USER_ROI	Camera cannot acquire the user-defined ROI. Resize and try again.
-1074364370	IMG1394_ERR_BUFFER_LIST_ALREADY_LOCKED	Buffer list already locked. Reconfigure acquisition and try again.
-1074364369	IMG1394_ERR_BUFFER_LIST_NOT_LOCKED	No buffer list. Reconfigure acquisition and try again.

-1074364368	IMG1394_ERR_RESOURCES_ALREADY_ALLOCATED	Isochronous resources already allocated. Reconfigure acquisition and try again.
-1074364367	IMG1394_ERR_BUFFER_LIST_EMPTY	Buffer list empty. Add at least one buffer.
-1074364366	IMG1394_ERR_FLAG_1	For Format 7, combination of speed, image position, image size, and color coding is incorrect.
-1074364365	IMG1394_ERR_BUFFER_NOT_AVAILABLE	Requested buffer unavailable. Contents of current buffer overwritten by the acquisition.
-1074364364	IMG1394_ERR_IMAGE_REP_NOT_SUPPORTED	Requested image representation not supported for current color coding.
-1074364363	IMG1394_ERR_BAD_OCCURRENCE	Invalid given occurrence. Unable to complete acquisition.

Important Information

[Warranty](#)

[Copyright](#)

[Trademarks](#)

[Patents](#)

[Warning Regarding Use of NI Products](#)

Warranty

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

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

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

Except as specified herein, National Instruments makes no warranties, express or implied, and specifically disclaims any warranty of merchantability or fitness for a particular purpose. Customer's right to recover damages caused by fault or negligence on the part of National Instruments shall be limited to the amount theretofore paid by the customer. National Instruments will not be liable for damages resulting from loss of data, profits, use of products, or incidental or consequential damages, even if advised of the possibility thereof. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action

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

Copyright

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

Trademarks

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

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

Handle Graphics®, MATLAB®, Real-Time Workshop®, Simulink®, and Stateflow® are registered trademarks, and TargetBox²™, xPC TargetBox²™, and Target Language Compiler™ are trademarks of The MathWorks, Inc.

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

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

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

Patents

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

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) National Instruments products are not designed with components and testing for a level of reliability suitable for use in or in connection with surgical implants or as critical components in any life support systems whose failure to perform can reasonably be expected to cause significant injury to a human.

(2) In any application, including the above, reliability of operation of the software products can be impaired by adverse factors, including but not limited to fluctuations in electrical power supply, computer hardware malfunctions, computer operating system software fitness, fitness of compilers and development software used to develop an application, installation errors, software and hardware compatibility problems, malfunctions or failures of electronic monitoring or control devices, transient failures of electronic systems (hardware and/or software), unanticipated uses or misuses, or errors on the part of the user or applications designer (adverse factors such as these are hereafter collectively termed "system failures"). Any application where a system failure would create a risk of harm to property or persons (including the risk of bodily injury and death) should not be reliant solely upon one form of electronic system due to the risk of system failure. To avoid damage, injury, or death, the user or application designer must take reasonably prudent steps to protect against system failures, including but not limited to back-up or shut down mechanisms. Because each end-user system is customized and differs from National Instruments' testing platforms and because a user or application designer may use National Instruments products in combination with other products in a manner not evaluated or contemplated by National Instruments, the user or application designer is ultimately responsible for verifying and validating the suitability of National Instruments products whenever National Instruments products are incorporated in a system or application, including, without limitation, the appropriate design, process and safety level of such system or application.

Technical Support and Professional Services

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

- [Support](#)—Online technical support resources at ni.com/support include the following:
 - **Self-Help Resources**—For answers and solutions, visit the award-winning National Instruments Web site for software drivers and updates, a searchable [KnowledgeBase](#), [product manuals](#), step-by-step troubleshooting wizards, thousands of example programs, tutorials, application notes, instrument drivers, and so on.
 - **Free Technical Support**—All registered users receive free Basic Service, which includes access to hundreds of Applications Engineers worldwide in the [NI Developer Exchange](#) at ni.com/exchange. National Instruments Applications Engineers make sure every question receives an answer.

For information about other [technical support options](#) in your area, visit ni.com/services or [contact](#) your local office at ni.com/contact.

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

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

Feature2

Describes a camera attribute.

Member	Type	Description
Enable	uInt32	Returns TRUE if the current camera attribute is present.
Absolute	uInt32	Returns TRUE if the current camera attribute supports the Absolute operation mode.
OnePush	uInt32	Returns TRUE if the current camera attribute supports the One Push operation mode.
Readable	uInt32	Returns TRUE if the current camera attribute is readable.
OnOff	uInt32	Returns TRUE if the current camera attribute supports the Off operation mode.
Auto	uInt32	Returns TRUE if the current camera attribute supports the Auto operation mode.
Manual	uInt32	Returns TRUE if the current camera attribute supports the Relative operation mode.
Relative_Min_Value	uInt32	Gets the minimum value for the Relative operation mode.
Relative_Max_Value	uInt32	Gets the maximum value for the Relative operation mode.
Relative_Current_Value	uInt32	Gets the current value for the Relative operation mode.
Relative_Default_Value	uInt32	Gets the default value for the Relative operation mode.
Relative_Attribute	uInt32	The camera attribute used to get/set the relative values.
Absolute_Min_Value	double	Gets the minimum value for the Absolute operation mode.
Absolute_Max_Value	double	Gets the maximum value for the Absolute operation mode.
Absolute_Current_Value	double	Gets the current value for the Absolute operation mode.
Absolute_Default_Value	double	Gets the default value for the Absolute operation mode.
Absolute_Attribute	uInt32	The camera attribute used to get/set the absolute values.
FeatureName	char[64]	Returns the name of the current camera attribute.

Interface File

Describes an interface file.

Member	Type	Description
Type	uInt32	Designates an NI-IMAQ IEEE 1394 interface file. Type has a value of 3 .
Version	uInt32	The version of the interface file. This number may increment with different versions of the driver as the format of the interface file changes.
Flags	uInt32	A bitwise mask of the current interface status. If bit 0 is on (value=1), the interface represents a camera that is currently connected. If bit 0 is off (value=0), the interface represents a disconnected camera.
SerialNumberHi	uInt32	The upper 32 bits of the interface serial number. Every camera has a unique value for Serial Number Hi .
SerialNumberLo	uInt32	The lower 32 bits of the interface serial number. Every camera has a unique value for Serial Number Lo .
InterfaceName	char[64]	The name of the interface. Use this name when opening the interface.
VendorName	char[64]	The vendor name of the camera designated for this interface. Vendor Name varies from camera to camera.
ModelName	char[64]	The model name of the camera designated for this interface. Model Name varies from camera to camera.
CameraFileName	char[64]	The name of the camera file that this interface uses. The camera file contains all the settings for a given camera. You can configure and save these settings in Measurement & Automation Explorer (MAX).

Video Mode

Describes a video mode.

Member	Type	Description
Format	uint32	The video format. Must be a value between 0 and 7.
Mode	uint32	The video mode. Must be a value between 0 and 7.
FrameRate	uint32	The video frame rate. Must be a value between 0 and 7.
VideoModeName	char[64]	The name of the video mode.

imaq1394AttributeInquiry

Format

```
rval = imaq1394AttributeInquiry(SESSION_ID sessionId, unsigned long  
attribute, unsigned long *minimumValue, unsigned long *maximumValue,  
unsigned long *readable, unsigned long *autoMode, unsigned long *enable);
```

Purpose

Determines if the camera supports the specified attribute. If so, the function returns information about the attribute, including minimum and maximum values, whether the attribute can be read, and if the attribute can be set in automatic mode.

Parameters

Parameter	Type	Description
sessionID	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen .
attribute	unsigned long	Specifies the attribute whose information you want to query.
minimumValue	unsigned long (passed by reference)	Minimum value of the attribute if it is supported by the camera (if enable is set to TRUE). Set this parameter to NULL if you do not need the return value.
maximumValue	unsigned long (passed by reference)	Maximum value of the attribute if it is supported by the camera (if enable is set to TRUE). Set this parameter to NULL if you do not need the return value.
readable	unsigned long (passed by reference)	Set to TRUE if the software can read the value of the specified attribute. Set this parameter to NULL if you do not need the return value.
autoMode	unsigned long (passed by reference)	Set to TRUE if the value of the specified attribute can be set to automatic mode. Set this parameter to NULL if you do not need the return value. To set an attribute in automatic mode, set its value to IMAQ1394_AUTOMODE using imaq1394SetAttribute .
enable	unsigned long (passed by reference)	Set to TRUE if the camera supports the specified attributes.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or [img1394ShowErrorCW](#), if you are using Microsoft Visual Basic.

imaq1394CameraOpen

Format

```
rval = imaq1394CameraOpen(char *camera_name, SESSION_ID *sessionId);
```

Purpose

Opens a camera by name, as configured in Measurement & Automation Explorer (MAX). Camera names follow the convention (cam0, cam1, ..., camN)

Parameters

Parameter	Type	Description
camera_name	char *	Name of the camera you want to use. camera_name must contain the camera identifier (cam0, cam1, ..., camN) or it must match the configuration file name you used to configure the camera in MAX.
sessionId	SESSION_ID (passed by reference)	A valid Session ID.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#) or [img1394ShowErrorCW](#), if you are using Microsoft Visual Basic.

imaq1394GetBuffer

Format

```
rval = imaq1394GetBuffer (SESSION_ID sessionID, unsigned long imageIndex,  
void **buffer);
```

Purpose

Gets a single image from a low-level acquisition setting. This function takes a buffer as an input. The buffer must be large enough to contain the entire image.

Parameters

Parameter	Type	Description
sessionID	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen .
imageIndex	unsigned long	The index of the image to retrieve. This parameter is used only in non-continuous acquisitions using multiple buffers (sequence acquisitions). Set this parameter to 0 for continuous acquisitions.
Buffer	void * (passed by reference)	The buffer that contains the image when the function returns.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure, this function returns an error code. You can obtain a more detailed error message with [`imaq1394ShowError`](#).

imaq1394GetFeatures

Format

```
rval = imaq1394GetFeatures (SESSION_ID sessionID, Feature *featureArray,  
unsigned long *feature_array_size);
```

Purpose

Retrieves the features supported by the camera. If you do not know in advance the number of features, perform the following steps:

1. Call this function with the **featureArray** parameter set to NULL. The needed size is then stored in **feature_array_size**.
2. Allocate **featureArray** with the given size.
3. Call this function one more time with the previously allocated array.

Parameters

Parameter	Type	Description
sessionID	SESSION_ID	A valid Session ID, which you can obtain with the function imag1394CameraOpen .
featureArray	Feature (passed by reference)	Contains a pointer to an array of feature structure elements in which the features supported by the camera are stored. Set this parameter to NULL to get the size needed by the array in the feature_array_size parameter.
feature_array_size	unsigned long (passed by reference)	Contains the size of the array used to store the features. If the user passes NULL as the featureArray parameter, this parameter contains the needed size.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GetFeaturesCW

Format

imaq1394GetFeaturesCW(sessionId As SESSION_ID, psaFeatureArray() As Feature) As IMG_ERR

Purpose

Retrieves the features supported by the camera.

Parameters

Description

Parameter	Type	
sessionId	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen <code>imaq1394CameraOpen.html</code> .
psaFeatureArray	SAFEARRAY (Feature)	An array of feature structure elements in which the features supported by the camera are stored.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394GetImage

Format

```
rval = imaq1394GetImage (SESSION_ID sessionID, unsigned long imageIndex,  
Image *image);
```

Purpose

Gets a single image from a low-level acquisition session using IMAQ Vision memory management. If the image type does not match the video format of the camera, `imaq1394GetImage` changes it to a suitable format.

Parameters

Parameter	Type	Description
sessionID	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen .
imageIndex	unsigned long	The index of the image to retrieve. This parameter is used only in non-continuous acquisitions using multiple buffers (sequence acquisitions). Set this parameter to 0 for continuous acquisitions.
Image	Image (passed by reference)	The buffer that contains the IMAQ Vision image when the function returns.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure, this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394Grab

Format

rval = imaq1394Grab (SESSION_ID sessionId, void **buffer);

Purpose

Acquires the most current frame into a previously allocated buffer. Call this function only after calling [imaq1394SetupGrab](#).

Parameters

Parameter	Type	Description
sessionID	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen .
image	void * (passed by reference)	The image into which to acquire. If image is NULL, imaq1394Grab creates a new image.

Return Value

On success, this function returns IMG1394_ERR_GOOD. On failure this function returns an error code. You can obtain a more detailed error message with [img1394ShowError](#).

imaq1394GrabCW

Format

imaq1394GrabCW(sessionId As SESSION_ID, dispatch As Object) As
IMG_ERR

Purpose

Acquires the most current frame into a valid CWIMAQImage. Call this function only after calling [imaq1394SetupGrabCW](#).

Parameters

Parameter	Type	Description
sessionId	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen <code>imaq1394CameraOpen.html</code> .
dispatch	Object	A valid CWIMAQImage.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure this function returns an error code. You can obtain a more detailed error message with [img1394ShowErrorCW](#).

imaq1394GrabImage

Format

```
rval = imaq1394GrabImage (SESSION_ID sessionId, Image *image);
```

Purpose

Acquires the most current frame into the specified IMAQ Vision image buffer. Call this function only after calling [imaq1394SetupGrab](#). If the image type does not match the video format of the camera, this function changes it to a suitable format.

Parameters

Parameter	Type	Description
sessionID	SESSION_ID	A valid Session ID, which you can obtain with the function imaq1394CameraOpen .
Image	Image * (passed by reference)	The image into which to acquire.

Return Value

On success, this function returns `IMG1394_ERR_GOOD`. On failure this function returns an error code. You can obtain more detailed error message with [`imaq1394ShowError`](#).

Branch Offices

Office	Telephone Number
Australia	1800 300 800
Austria	43 0 662 45 79 90 0
Belgium	32 0 2 757 00 20
Brazil	55 11 3262 3599
Canada	800 433 3488
China	86 21 6555 7838
Czech Republic	420 224 235 774
Denmark	45 45 76 26 00
Finland	385 0 9 725 725 11
France	33 0 1 48 14 24 24
Germany	49 0 89 741 31 30
India	91 80 51190000
Israel	972 0 3 6393737
Italy	39 02 413091
Japan	81 3 5472 2970
Korea	82 02 3451 3400
Lebanon	961 0 1 33 28 28
Malaysia	1800 887710
Mexico	01 800 010 0793
Netherlands	31 0 348 433 466
New Zealand	0800 553 322
Norway	47 0 66 90 76 60
Poland	48 22 3390150
Portugal	351 210 311 210
Russia	7 095 783 68 51
Singapore	1800 226 5886
Slovenia	386 3 425 4200
South Africa	27 0 11 805 8197
Spain	34 91 640 0085
Sweden	46 0 8 587 895 00
Switzerland	41 56 200 51 51
Taiwan	886 02 2377 2222
Thailand	662 992 7519
United Kingdom	44 0 1635 523545
United States (Corporate)	512 683 0100