NATIONAL
INSTRUMENTS™

NI-IMAQ I/O C Function Reference Help

July 2007, 371553D-01

This help file contains function reference information for developing C code to control NI-IMAQ I/O devices.

NI-IMAQ I/O is the National Instruments I/O device driver software for controlling the digital I/O on image acquisition devices.

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

Using Help

Important Information

Technical Support and Professional Services

To comment on National Instruments documentation, refer to the National Instruments Web site.

Using Help

Conventions

Navigating Help

Searching Help

Printing Help File Topics

Conventions

This help file uses the following conventions:

» 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 in the software, such as menu items and dialog box options. Bold text also denotes parameter names.

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

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

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

Navigating Help (Windows Only)

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

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

Searching Help (Windows Only)

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

Wildcards

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

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

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

Nested Expressions

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

Boolean Expressions

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

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

Search Options

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

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

Printing Help File Topics (Windows Only)

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

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

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

4. Click the **OK** button.

Printing PDF Documents

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

NI-IMAQ I/O Open/Close Functions

Use the NI-IMAQ I/O open and close functions to open and close sessions on an NI-IMAQ I/O device.

imaqIOOpen

imaqIOClose

imaqIOOpen

Format

rval = imaqIOOpen(const char* deviceName, IMAQIO_SESSION* id);

Purpose

Opens a reference to an NI-IMAQ I/O device.

Parameters

Name	Type	Direction
deviceName	const char*	input
id	IMAQIO_SESSION*	output

Parameter Discussion

deviceName is the name of the device to open a session on. This is the VISA name that is defined in Measurement & Automation Explorer (MAX). The name appears similar to RIO#::INSTR, where # represents a specific RIO device, depending on how many RIO devices you have in the system.

id is a pointer to an IMAQIO_SESSION variable. On success, the variable is populated with a valid IMAQIO_SESSION to use in subsequent functions.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOClose

Format

rval = imaqIOClose(IMAQIO_SESSION id);

Purpose

Closes a reference to an NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input

Parameter Discussion

id is the ID of the session to close.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

NI-IMAQ I/O Static I/O Functions

Use the NI-IMAQ I/O static I/O functions to manipulate the signals on an NI-IMAQ I/O device.

imaqIOGetAttribute

imaqIOSetAttribute

imaqIODrive

imaqIORead

imaqIOGetAttribute

Format

rval = imaqIOGetAttribute(IMAQIO_SESSION id, IMAQIOAttribute attribute, unsigned long *value);

Purpose

Reads one of the NI-IMAQ I/O device attributes.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
attribute	IMAQIOAttribute	input
value	unsigned long*	output

Parameter Discussion

id is the ID of the session to get an attribute for.

attribute is the attribute to retrieve. The following values are valid:

IMAQIO_ATTRIBUTE_PRODUCT_SELECT_CURRENT	Gets the **value** of the current product selection line. ISO Inputs 4 thru 0 are the product selection lines.
IMAQIO_ATTRIBUTE_PRODUCT_SELECT_LATCHED	Gets the **value** of the currently latched product selection line. ISO Inputs 4 thru 0 are the product selection lines. ISO Input 5 latches in the data on ISO Inputs 4 thru 0.
IMAQIO_ATTRIBUTE_QUADRATURE_ENCODER	Gets the **value** of the quadrature encoder. The quadrature encoder uses ISO Input 6 for its Phase A input and ISO Input 7 for its Phase B input.
IMAQIO_ATTRIBUTE_WATCHDOG_STATUS	Gets the **value** of the watchdog

	status. 0 indicates that no errors have occurred, and 1 indicates that a timeout occurred.
IMAQIO_ATTRIBUTE_ISOPOWER_PRESENT	Gets the ISO power status. A **value** of 1 indicates that ISO power is present. A **value** of 0 indicates that ISO power is not present.

value is a pointer to the value of the attribute retrieved.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOSetAttribute

Format

rval = imaqIOSetAttribute(IMAQIO_SESSION id, IMAQIOAttribute attribute, unsigned long value);

Purpose

Sets an attribute on the NI-IMAQ I/O device.

Note This function is reserved for future use. All of the valid values for the **attribute** are currently not settable.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
attribute	IMAQIOAttribute	input
value	unsigned long	input

Parameter Discussion

id is the ID of the session to set an attribute on.

attribute is the attribute to set.

Note Currently, there are no settable attributes.

value is a value to set the attribute to.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIODrive

Format

rval = imaqIODrive(IMAQIO_SESSION id, IMAQIOSignalType signalType, unsigned long signalNumber, IMAQIOSignalState value);

Purpose

Drives general-purpose digital outputs with the static I/O on the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
signalType	IMAQIOSignalType	input
signalNumber	unsigned long	input
value	IMAQIOSignalState	input

Parameter Discussion

id is the ID of the session to drive.

signalType is the type of signal to drive. The following values are valid for this parameter:

Name	Constant	Description
IMAQIO_SIGNAL_ISO_OUT	3	ISO output signals are general-purpose outputs.
IMAQIO_SIGNAL_TTL_OUT	6	TTL output signals are general-purpose outputs.

signalNumber is the line number of the signal you want to drive.

value is the state you want to drive the line to. The following values are valid:

IMAQIO_SIGNAL_STATE_HIGH	Drives the line high when the signal is TRUE.
IMAQIO_SIGNAL_STATE_LOW	Drives the line low when the signal is TRUE.
IMAQIO_SIGNAL_STATE_HI_Z	Disables output on the line. This option is valid only for TTL Outs.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIORead

Format

rval = imaqIORead(IMAQIO_SESSION id, IMAQIOSignalType signalType, unsigned long signalNumber, IMAQIOSignalState* value);

Purpose

Reads general-purpose digital inputs from the static I/O of the NI-IMAQ I/O device.

Parameters

Name	**Type**	**Direction**
id	IMAQIO_SESSION	input
signalType	IMAQIOSignalType	input
signalNumber	unsigned long	input
value	IMAQIOSignalState*	output

Parameter Discussion

id is the ID of the session to read from.

signalType is the type of signal to read from. The following values are valid for this parameter:

Name	Constant	Description
IMAQIO_SIGNAL_ISO_IN	2	ISO input signals include pulse trigger inputs, a shutdown input, and a quadrature encoder input.
IMAQIO_SIGNAL_ISO_OUT	3	ISO output signals are general-purpose outputs.
IMAQIO_SIGNAL_TTL_IN	5	TTL input signals are general-purpose inputs and pulse trigger inputs.
IMAQIO_SIGNAL_TTL_OUT	6	TTL output signals are general-purpose outputs and pulse generation outputs.

signalNumber is the line number of the signal to read from.

value is a pointer to the signal state read from the NI-IMAQ I/O device. The following values are valid:

IMAQIO_SIGNAL_STATE_HIGH	Drives the line high when the signal is TRUE.
IMAQIO_SIGNAL_STATE_LOW	Drives the line low when the signal is TRUE.
IMAQIO_SIGNAL_STATE_HI_Z	Disables output on the line. This option is valid only for TTL Outs.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

NI-IMAQ I/O Pulse Generator Functions

Use the NI-IMAQ I/O pulse generator functions to manipulate pulses on the signals of the NI-IMAQ I/O device.

imaqIOPulseCreate

imaqIOPulseStart

imaqIOPulseStop

imaqIOPulseDispose

imaqIOPulseCreate

Format

```
rval = imaqIOPulseCreate(IMAQIO_SESSION id,
        IMAQIOTimebase delayTimebase, unsigned long delay,
        IMAQIOTimebase widthTimebase, unsigned long width,
        IMAQIOSignalType triggerSignalType, unsigned long triggerSignalNumber
        IMAQIOSignalType outputSignalType, unsigned long outputSignalNumber,
        IMAQIOPulseMode pulseMode, IMAQIO_PULSE_ID* pulseID);
```

Purpose

Configures one of the six pulse generators for the NI-IMAQ I/O device. Refer to your hardware documentation for more information about connecting and configuring the pulse generators.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
delayTimebase	IMAQIOTimebase	input
delay	unsigned long	input
widthTimebase	IMAQIOTimebase	input
width	unsigned long	input
triggerSignalType	IMAQIOSignalType	input
triggerSignalNumber	unsigned long	input
triggerSignalPolarity	IMAQIOPolarity	input
outputSignalType	IMAQIOSignalType	input
outputSignalNumber	unsigned long	input
outputSignalPolarity	IMAQIOPolarity	input
pulseMode	IMAQIOPulseMode	input
pulseID	IMAQIO_PULSE_ID*	output

Parameter Discussion

id is the ID of the session to create a pulse on.

delayTimeBase is the timebase of the pulse delay. The following values are valid for this parameter:

Name	**Constant**	**Description**
IMAQIO_TIMEBASE_ENCODER_TICKS	4	Defines the delay or width in encoder counts
IMAQIO_TIMEBASE_MICROSECONDS	5	Defines the delay or width in microseconds.

delay is the pulse delay.

widthTimeBase is the timebase of the pulse width. The following values are valid for this parameter:

Name	**Constant**	**Description**
IMAQIO_TIMEBASE_MICROSECONDS	5	Defines the delay or width in microseconds.

width is the width of the pulse.

triggerSignalType is the type of signal to trigger on. The following values are valid for this parameter:

Name	**Constant**	**Description**
IMAQIO_SIGNAL_ISO_IN	2	Use isolated inputs 5, 8, and 12 to trigger a pulse.
IMAQIO_SIGNAL_STATUS	4	Signal status signals are internal timing signals, including the immediate trigger mode.
IMAQIO_SIGNAL_TTL_IN	5	Use either TTL input 0 or 1 to trigger a pulse.

triggerSignalNumber is the line number of the signal to trigger on. If you set **triggerSignalType** to IMAQIO_SIGNAL_STATUS, use

IMAQIO_STATUS_SIGNAL_NONE to cause the pulse to start immediately upon calling imaqIOPulseStart.

triggerSignalPolarity is the polarity of the trigger as defined by the following constants:

IMAQIO_POLARITY_HIGH_TRUE	Triggers the pulse on a rising edge.
IMAQIO_POLARITY_LOW_TRUE	Triggers the pulse on a falling edge.

outputSignalType is the type of signal to create a pulse on. The following values are valid for this parameter:

Name	Constant	Description
IMAQIO_SIGNAL_ISO_OUT	3	ISO output signals are general-purpose outputs.
IMAQIO_SIGNAL_TTL_OUT	6	TTL output signals are general-purpose outputs.

outputSignalNumber is the line number of the signal to create the pulse on.

outputSignalPolarity is the polarity of the pulse output as defined by the following constants:

IMAQIO_POLARITY_HIGH_TRUE	Drives the line high during the pulse assertion, which is dictated by the **width**.
IMAQIO_POLARITY_LOW_TRUE	Drives the line low during the pulse assertion, which is dictated by the **width**.

pulseMode is a value that indicates if the pulse is generated once or continuously. **pulseMode** can be one of the following constants:

PULSE_MODE_TRAIN	Pulse is generated continuously after the trigger is asserted. Choose this option to generate a continuous pulse train that is inactive for the time specified in the **delay** parameter and active for the time specified in the **width** parameter. When the pulse train is started, it continues

	periodically until you call imaqIOPulseStop, imaqIOPulseDispose, or imaqIOClose. This mode is valid only if you set **triggerSignalType** to IMAQIO_SIGNAL_STATUS, and **triggerSignalNumber** is set to IMAQIO_STATUS_SIGNAL_NONE.
PULSE_MODE_SINGLE	Pulse is generated only one time in response to the triggering condition being satisfied. If **triggerSignalType** is set to IMAQIO_SIGNAL_STATUS and **triggerSignalNumber** is set to IMAQIO_STATUS_SIGNAL_NONE, pulse generation will start immediately after calling imaqIOPulseStart. Otherwise, the pulse will start after the respective trigger line is asserted. In either case, after the trigger condition is satisfied, the output line stays inactive for the time specified in the delay parameter and becomes active for the time specified in the width parameter.
PULSE_MODE_SINGLE_REARM	Pulse occurs one time on each trigger occurrence. Choose this option to generate a rearmed single-shot pulse. On every occurrence of the trigger, the output line stays inactive for the time specified in the **delay** parameter and becomes active for the time specified in the **width** parameter. When the pulse is started, output toggles for each occurrence of the trigger until you call imaqIOPulseStop, imaqIOPulseDispose, or

imaqIOClose. This mode works only when the application is configured to trigger on the TTL In or ISO In lines.

pulseID is a pointer to a variable to receive the pulse ID. If the function succeeds, the variable is populated with a valid PULSE_ID that you can use in subsequent functions.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOPulseStart

Format

rval = imaqIOPulseStart(IMAQIO_SESSION id, IMAQIO_PULSE_ID PulseID);

Purpose

Starts a pulse generator session for the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
PulseID	IMAQIO_PULSE_ID	input

Parameter Discussion

id is the ID of the session to start a pulse on.

PulseID is the pulse to start.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOPulseStop

Format

rval = imaqIOPulseStop(IMAQIO_SESSION id, IMAQIO_PULSE_ID PulseID);

Purpose

Stops a pulse generator session for the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
PulseID	IMAQIO_PULSE_ID	input

Parameter Discussion

id is the ID of the session to stop a pulse on.

PulseID is the pulse to stop.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOPulseDispose

Format

rval = imaqIOPulseDispose(IMAQIO_SESSION id, IMAQIO_PULSE_ID PulseID);

Purpose

Disposes of a pulse ID, relinquishing its resources.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
PulseID	IMAQIO_PULSE_ID	input

Parameter Discussion

id is the ID of the session to dispose a pulse on.

PulseID is the pulse to dispose of.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

Change Detectors

Use the NI-IMAQ I/O change detector functions to detect signal state changes on the various input lines of the NI-IMAQ I/O device.

imaqIOChangeDetectConfigure

imaqIOChangeDetectQuery

imaqIOChangeDetectReset

imaqIOChangeDetectConfigure

Format

```
rval = imaqIOChangeDetectConfigure(IMAQIO_SESSION id,
IMAQIOSignalType triggerType, unsigned long triggerNumber,
IMAQIOChangeDetectMode detectMode, unsigned long filter);
```

Purpose

Configures the signal state change detector logic of an input line on the NI-IMAQ I/O device. Use this function to arm the detection of rising edges, falling edges, or both on the specified input trigger line.

Parameters

Name	**Type**	**Direction**
id	IMAQIO_SESSION	input
triggerType	IMAQIOSignalType	input
triggerNumber	unsigned long	input
detectMode	IMAQIOChangeDetectMode	input
filter	unsigned	input

Parameter Discussion

id is the ID of the session to close.

triggerType is the type of trigger line to detect a change on.

triggerNumber is the trigger number to detect a change on.

detectMode is the change detection mode. This specifies what type of edge to be sensitive to.

filter is used to suppress high-frequency noise from the change detection. After receiving an edge, the trigger line must be held steady in the new logic state for filter × (25) ns in order for the device to register a successful change detection. Set this parameter to 0 to disable filtering.

imaqIOChangeDetectQuery

Format

rval = imaqIOChangeDetectQuery(IMAQIO_SESSION id, IMAQIOSignalType triggerType, unsigned long triggerNumber, unsigned long* changed);

Purpose

Queries the NI-IMAQ I/O device for information about whether the given trigger line has changed. For this function to behave properly, you must have previously configured the change detector for the same trigger line using imaqIOChangeDetectConfigure. After a change has been registered for a trigger line, this function will continually report that the trigger line has changed until you reset it by calling imaqIOChangeDetectReset.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
triggerType	IMAQIOSignalType	input
triggerNumber	unsigned long	input
changed	unsigned long*	output

Parameter Discussion

id is the ID of the NI-IMAQ I/O device that will be detecting trigger line changes.

triggerType is the type of trigger line to detect a change on.

triggerNumber is the trigger number to detect a change on.

detectMode is the change detection mode. This specifies what type of edge to be sensitive to.

changed is a pointer to a variable to receive the trigger changed information. If no change was detected, the variable will be set to 0. Otherwise, the variable will be set to 1.

imaqIOChangeDetectReset

Format

rval = imaqIOChangeDetectReset(IMAQIO_SESSION id, IMAQIOSignalType triggerType, unsigned long triggerNumber);

Purpose

Resets the change detection mechanism for the given trigger line. This rearms the change detector logic for detection of a new trigger edge.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
triggerType	IMAQIOSignalType	input
triggerNumber	unsigned long	input

Parameter Discussion

id is the ID of the NI-IMAQ I/O device that will be detecting trigger line changes.

triggerType is the type of trigger line to detect a change on.

triggerNumber is the trigger number to detect a change on.

NI-IMAQ I/O Watchdog Functions

Use the NI-IMAQ I/O watchdog functions to set up a timer for the NI-IMAQ I/O device.

imaqIOWatchdogConfigure

imaqIOWatchdogArm

imaqIOWatchdogClear

imaqIOWatchdogWhack

imaqIOWatchdogConfigure

Format

rval = imaqIOWatchdogConfigure(IMAQIO_SESSION id, unsigned long timeout, IMAQIOExpirationAction expirationAction);

Purpose

Configures a watchdog timer on the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
timeout	unsigned long	input
expirationAction	IMAQIOExpirationAction	input

Parameter Discussion

id is the ID of the session to configure the watchdog on.

timeout is the time, in milliseconds, that can occur without a reset before the watchdog occurs.

expirationAction is the action to take when the watchdog occurs:

IMAQIO_EXPIRATION_ACTION_INDICATOR_ONLY	Sets the watch status attribute.
IMAQIO_EXPIRATION_ACTION_ASSERT_TTL_OUT_0	Drives the TTL line high when watchdog occu
IMAQIO_EXPIRATION_ACTION_SHUTDOWN	Shuts down the IMAQ I/O devic the watchdog o Refer to imaqIOEnableS for information enabling shutd the NI-IMAQ I/

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOWatchdogArm

Format

rval = imaqIOWatchdogArm(IMAQIO_SESSION id);

Purpose

Arms the watchdog timer on the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input

Parameter Discussion

id is the ID of the session to arm the watchdog on.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOWatchdogWhack

Format

rval = imaqIOWatchdogWhack(IMAQIO_SESSION id);

Purpose

Resets the watchdog timer on the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input

Parameter Discussion

id is the ID of the session to reset the watchdog timer on.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOWatchdogClear

Format

```
rval = imaqIOWatchdogClear(IMAQIO_SESSION id);
```

Purpose

Clears the watchdog timer on the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input

Parameter Discussion

id is the ID of the session to clear the watchdog on.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

NI-IMAQ I/O Shutdown States Functions

Use the NI-IMAQ I/O shutdown states functions to enable/disable and get/set shutdown states on an NI-IMAQ I/O device. Refer to your hardware documentation for information about the NI-IMAQ I/O device shutdown configuration settings and effects.

imaqIOSetShutdownState

imaqIOGetShutdownState

imaqIOEnableShutdown

imaqIOQueryShutdown

imaqIOEnableShutdown

Format

rval = imaqIOEnableShutdown(IMAQIO_SESSION id, unsigned long enableShutdown);

Purpose

Enables or disables the shutdown functionality of the NI-IMAQ I/O device.

When the shutdown functionality is enabled, the device reaches a shutdown state under two conditions:

- If ISO_IN_11 is Low
- If the Watchdog Timer expires and the expiration action is IMAQIO_EXPIRATION_ACTION_SHUTDOWN

When the device is in a shutdown state, it asserts the outputs with the values that have been specified through imaqIOSetShutdownState. The default value for the outputs is Disabled for TTL Outputs and Low for ISO Outputs. Shutdown is a fatal condition. Clear shutdown by powering off and then powering on your system.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
enableShutdown	unsigned long	input

Parameter Discussion

id is the ID of the session to enable or disable shutdown on.

enableShutdown determines if shutdown is enabled for the NI-IMAQ I/O device. 1 enables shutdown, and 0 disables shutdown.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOSetShutdownState

Format

rval = imaqIOSetShutdownState(IMAQIO_SESSION id, IMAQIOSignalType signalType, unsigned long signalNumber, IMAQIOSignalState value);

Purpose

Sets the shutdown state of one of the output signals on the NI-IMAQ I/O device. The default shutdown state is IMAQIO_SIGNAL_STATE_LOW.

Parameters

Name	**Type**	**Direction**
id	IMAQIO_SESSION	input
signalType	IMAQIOSignalType	input
signalNumber	unsigned long	input
value	IMAQIOSignalState	input

Parameter Discussion

id is the ID of the session to set the shutdown state on.

signalType is the type of signal to set the shutdown state on. The following values are valid for this parameter:

Name	Constant	Description
IMAQIO_SIGNAL_ISO_OUT	3	ISO output signals are general-purpose outputs.
IMAQIO_SIGNAL_TTL_OUT	6	TTL output signals are general-purpose outputs.

signalNumber is the line number to set the shutdown state on.

value is the shutdown state to set the device to. The following values are valid:

IMAQIO_SIGNAL_STATE_HIGH	Drives the line high when the signal is TRUE.
IMAQIO_SIGNAL_STATE_LOW	Drives the line low when the signal is TRUE.
IMAQIO_SIGNAL_STATE_HI_Z	Disables output on the line. This option is valid only for TTL output signals.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOGetShutdownState

Format

rval = imaqIOGetShutdownState(IMAQIO_SESSION id, IMAQIOSignalType signalType, unsigned long signalNumber, IMAQIOSignalState* value);

Purpose

Gets the shutdown state of one of the output signals on the NI-IMAQ I/O device.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
signalType	IMAQIOSignalType	input
signalNumber	unsigned long	input
value	IMAQIOSignalState*	output

Parameter Discussion

id is the ID of the session to get the shutdown state for.

signalType is the type of signal to associated with the NI-IMAQ I/O device. The following values are valid for this parameter:

Name	Constant	Description
IMAQIO_SIGNAL_ISO_OUT	3	ISO output signals are general-purpose outputs.
IMAQIO_SIGNAL_TTL_OUT	6	TTL output signals are general-purpose outputs.

signalNumber is the line number of the signal associated with the NI-IMAQ I/O device.

value is a pointer to the state of the NI-IMAQ I/O device. The following values are valid:

IMAQIO_SIGNAL_STATE_HIGH	Drives the line high when the signal is TRUE.
IMAQIO_SIGNAL_STATE_LOW	Drives the line low when the signal is TRUE.
IMAQIO_SIGNAL_STATE_HI_Z	Disables output on the line. This option is valid only for TTL output signals.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

imaqIOQueryShutdown

Format

rval = imaqIOQueryShutdown(IMAQIO_SESSION id, unsigned long* shutdownEnabled);

Purpose

Queries the shutdown functionality of the NI-IMAQ I/O device to determine if it is enabled or disabled.

Parameters

Name	Type	Direction
id	IMAQIO_SESSION	input
shutdownEnabled	unsigned long*	output

Parameter Discussion

id is the ID of the session to query.

shutdownEnabled is a pointer to return data that indicates if the shutdown functionality of the NI-IMAQ I/O device is enabled. 0 indicates that the shutdown functionality is disabled, and 1 indicates that the shutdown functionality is enabled.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

NI-IMAQ I/O Error Handling Function

Use the NI-IMAQ I/O error handling function to retrieve error information.

imaqIOGetErrorText

imaqIOGetErrorText

Format

```
rval = imaqIOGetErrorText(IMAQIO_ERR errorCode, unsigned long*
errorTextLength, char *errorText);
```

Purpose

Gets the error text associated with an error code.

Parameters

Name	Type	Direction
errorCode	IMAQIO_ERR	input
errorTextLength	unsigned long*	input/output
errorText	char*	output

Parameter Discussion

errorCode is the error code to get text for.

errorTextLength is the size of the buffer to store error text in. If **errorTextLength** is smaller than the buffer size you pass in here, IMAQIO_ERR_ERROR_TEXT_TOO_SHORT is returned. Regardless of whether an error occurs, this function returns the error text length in **errorTextLength**.

errorText is a pointer to a buffer to write the error text into.

Return Value

This function returns 0 on success. On failure, this function returns an error code. For information about the error code, call imaqIOGetErrorText.

Constants and Enumerations

The following NI-IMAQ I/O constants and enumerations provide the valid values for parameters in the NI-IMAQ I/O methods:

IMAQIO_ERR

IMAQIO_PULSE_ID

IMAQIO_SESSION

IMAQIOAttribute

imaqIOChangeDetectMode

IMAQIOExpirationAction

IMAQIOPolarity

IMAQIOPulseMode

IMAQIOSignalState

IMAQIOSignalType

IMAQIOTimebase

IMAQIOAttribute Constants

IMAQIOAttribute constants are the constants for the **attribute** parameter of the imaqIOGetAttribute and imaqIOSetAttribute methods in the NI-IMAQ I/O library.

Note Currently, these values are not settable.

- IMAQIO_ATTRIBUTE_PRODUCT_SELECT_CURRENT—Gets the value of the current product selection line. ISO Inputs 4 thru 0 are the product selection lines.
- IMAQIO_ATTRIBUTE_PRODUCT_SELECT_LATCHED—Gets the value of the currently latched product selection line. ISO Inputs 4 thru 0 are the product selection lines. ISO Input 5 latches in the data on ISO Inputs 4 thru 0.
- IMAQIO_ATTRIBUTE_QUADRATURE_ENCODER—Gets the value of the quadrature encoder. The quadrature encoder uses ISO Input 6 for its Phase A input and ISO Input 7 for its Phase B input.
- IMAQIO_ATTRIBUTE_WATCHDOG_STATUS—Gets the value of the watchdog status. 1 indicates that no errors occurred, and 0 indicates that a timeout occurred.
- IMAQIO_ATTRIBUTE_ISOPOWER_PRESENT—Gets the ISO power status. A value of 1 indicates that ISO power is present. A value of 0 indicates that ISO power is absent.

IMAQIOChangeDetectMode

IMAQIOChangeDetectMode constants are the constants for the imaqIOChangeDetectConfigure method in the NI-IMAQ I/O library:

- IMAQIO_CHANGE_DETECT_ANY_EDGE—Detects a change on any edge.
- IMAQIO_CHANGE_DETECT_RISING_EDGE—Detects a change on a rising edge.
- IMAQIO_CHANGE_DETECT_FALLING_EDGE—Detects a change on a falling edge.

IMAQIOExpirationAction Constants

IMAQIOExpirationAction constants are the constants for the **expirationAction** parameter of the imaqIOWatchdogConfigure method in the NI-IMAQ I/O library. Use these constants to specify what action to take when the watchdog timer expires.

- IMAQIO_EXPIRATION_ACTION_INDICATOR_ONLY—Displays a message when the watchdog occurs.
- IMAQIO_EXPIRATION_ACTION_ASSERT_TTL_OUT_0—Drives the TTL Out 0 line when the watchdog occurs.
- IMAQIO_EXPIRATION_ACTION_SHUTDOWN—Shuts down the NI-IMAQ I/O device when the watchdog occurs. Refer to imaqIOEnableShutdown for information about enabling shutdown.

IMAQIOPolarity Constants

IMAQIOPolarity constants are the constants for the imaqIOPulseCreate method in the NI-IMAQ I/O library.

outputSignalPolarity

For the **outputSignalPolarity** parameter, these constants behave as follows:

- IMAQIO_POLARITY_HIGH_TRUE—Drives the line high during the pulse assertion, which is dictated by the **width** parameter.
- IMAQIO_POLARITY_LOW_TRUE—Drives the line low during the pulse assertion, which is dictated by the **width** parameter.

triggerSignalPolarity

For the **triggerSignalPolarity** parameter, these constants behave as follows:

- IMAQIO_POLARITY_HIGH_TRUE—Triggers the pulse when the signal is high.
- IMAQIO_POLARITY_LOW_TRUE—Triggers the pulse when the signal is low.

IMAQIOPulseMode Constants

IMAQIOPulseMode constants are the constants for the imaqIOPulseCreate method in the NI-IMAQ I/O library.

- PULSE_MODE_TRAIN—Pulse is generated continuously after the trigger is asserted. Choose this option to generate a continuous pulse train that is inactive for the time specified in the delay parameter, and active for the time specified in the width parameter. When the pulse train is started, it continues periodically until you call imaqIOPulseStop, imaqIOPulseDispose, or imaqIOClose. This mode is valid only if you set **triggerSignalType** to IMAQIO_SIGNAL_STATUS, and triggerSignalNumber is set to IMAQIO_STATUS_SIGNAL_NONE.
- PULSE_MODE_SINGLE—This option is not supported.
- PULSE_MODE_SINGLE_REARM—Pulse occurs one time on each trigger occurrence. Choose this option to generate a rearmed single shot pulse. On every occurrence of the trigger, the output line stays inactive for the time specified in the delay parameter, and becomes active for the time specified in the width parameter. When the pulse is started, output toggles for each occurrence of the trigger until you call imaqIOPulseStop, imaqIOPulseDispose, or imaqIOClose. This mode works only when the application is configured to trigger on the TTL In or ISO In lines.

IMAQIOSignalState Constants

IMAQIOSignalState constants are the constants for the **value** parameter of the following methods:

- imaqIODrive
- imaqIORead
- imaqIOGetShutdownState
- imaqIOSetShutdownState

- IMAQIO_SIGNAL_STATE_HIGH—Drives the line high when the signal is True.
- IMAQIO_SIGNAL_STATE_LOW—Drives the line low when the signal is True.
- IMAQIO_SIGNAL_STATE_HI_Z—Disables output on the line. This option is valid only for TTL output signals.

IMAQIOSignalType Enumeration

IMAQIOSignalType enumerations are the constants for the **signalType** parameter on the methods listed in the following table.

Attribute	Description	Applicable Methods
IMAQIO_SIGNAL_ISO_IN	Pulse trigger inputs, a shutdown input, and a quadrature encoder input	imaqIORead imaqIOPulseCreate (triggerSi imaqIOChangeDetectMode
IMAQIO_SIGNAL_ISO_OUT	General-purpose outputs	imaqIODrive imaqIORead imaqIOPulseCreate (outputTri imaqIOSetShutdownState imaqIOGetShutdownState
IMAQIO_SIGNAL_STATUS	Internal timing signals, including the immediate trigger mode	imaqIOPulseCreate (triggerSi
IMAQIO_SIGNAL_TTL_IN	General-purpose inputs	imaqIORead imaqIOPulseCreate (triggerSi imaqIOChangeDetectMode
IMAQIO_SIGNAL_TTL_OUT	General-purpose	imaqIODrive imaqIORead

outputs	imaqIOPulseCreate (outputTri imaqIOSetShutdownState imaqIOGetShutdownState

IMAQIOStatusSignal Enumeration

Purpose

This enumeration provides options you can use for the signal number when you set signal type to status.

IMAQIO_STATUS_SIGNAL_NONE—Sets the signal number to none, which results in an immediate trigger.

IMAQIOTimebase Enumeration

IMAQIOTimebase enumerations are the constants for the **delayTimebase** and **widthTimebase** parameters on the imaqIOPulseCreate method.

You can use the following constants with this data type:

- IMAQIO_TIMEBASE_ENCODER_TICKS—Defines the delay or width in encoder counts.
- IMAQIO_TIMEBASE_MICROSECONDS—Defines the delay or width in microseconds.

Note The IMAQIO_TIMEBASE_ENCODER_TICKS enumeration is not applicable to the **widthTimebase** parameter.

IMAQIO_ERR

IMAQIO_ERR is the typedef for the **errorCode** parameter of the imaqIOGetErrorText method. The **errorCode** parameter is returned by each method in the NI-IMAQ I/O library.

IMAQIO_PULSE_ID

IMAQIO_PULSE_ID is the structure that defines the functionality of the **pulseID** parameter of the following methods:

- imaqIOPulseCreate
- imaqIOPulseDispose
- imaqIOPulseStart
- imaqIOPulseStop

IMAQIO_SESSION

IMAQIO_SESSION is the typedef structure that defines the functionality of the **id** parameter of the following methods:

- imaqIOClose
- imaqIODrive
- imaqIOEnableShutdown
- imaqIOGetAttribute
- imaqIOOpen
- imaqIORead
- imaqIOGetShutdownState
- imaqIOPulseCreate
- imaqIOPulseDispose
- imaqIOPulseStart
- imaqIOPulseStop
- imaqIOQueryShutdown
- imaqIOSetAttribute
- imaqIOSetShutdownState
- imaqIOWatchdogArm
- imaqIOWatchdogClear
- imaqIOWatchdogConfigure
- imaqIOWatchdogWhack
- imaqIOChangeDetectConfigure
- imaqIOChangeDetectQuery
- imaqIOChangeDetectReset

Data Types for imaqIO

Visual Basic	C/C++	Description
Long	long	32-bit signed integer.
Picture	LPPICTUREDISP	An OLE Automation Interface to a picture or image.
String	BSTR	A string.
Void	void	No data.

Error Codes

Every NI-IMAQ I/O function is of the following form:

rval = Function_Name(parameter 1, parameter 2, ... parameter n);

Each function returns a status code (rval) that indicates the success or failure of the function. The following table describes the error codes returned by each NI-IMAQ I/O function.

Code	Name
-301521	IMAQIO_ERR_INVALID_PULSE_MODE
-301520	IMAQIO_ERR_ATTRIBUTE_NOT_SETTABLE
-301519	IMAQIO_ERR_TIMEOUT_OUT_OF_RANGE
-301518	IMAQIO_ERR_INVALID_WATCHDOG_TIMER_EXPIRATION_A
-301517	IMAQIO_ERR_INVALID_ATTRIBUTE_TO_READ
-301516	IMAQIO_ERR_INVALID_PULSE_ID

-301515	IMAQIO_ERR_INVALID_TIMEBASE
-301514	IMAQIO_ERR_INVALID_POLARITY
-301513	IMAQIO_ERR_PULSE_OUTPUT_ALREADY_IN_USE
-301512	IMAQIO_ERR_INVALID_PULSE_TRIGGER_SIGNAL
-301511	IMAQIO_ERR_INVALID_PULSE_OUTPUT_SIGNAL
-301510	IMAQIO_ERR_INVALID_SIGNAL_STATE_FOR_THIS_CHANN
-301509	IMAQIO_ERR_INVALID_SIGNAL_FOR_THIS_OPERATION
-301508	IMAQIO_ERR_PULSE_ALREADY_RUNNING
-301507	IMAQIO_ERR_INVALID_SESSION
-301506	IMAQIO_ERR_INTERNAL_ERROR
-301505	IMAQIO_ERR_ERROR_NOT_FOUND

-301504	IMAQIO_ERR_ERROR_TEXT_TOO_SHORT
-301503	IMAQIO_ERR_OUT_OF_MEMORY
-301502	IMAQIO_ERR_INVALID_POINTER
-301501	IMAQIO_ERR_DEVICE_IN_USE
-301500	IMAQIO_ERR_DEVICE_NOT_FOUND

Important Information

Warranty

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

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

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

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT 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

Trademarks

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 Discussion Forums at ni.com/forums. 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.

Related Documentation

Some NI-IMAQ for IEEE 1394 manuals are available as PDFs. You must have Adobe Acrobat Reader with Search and Accessibility 5.0.5 or later installed to view the PDFs. Refer to the Adobe Systems Incorporated Web site to download Acrobat Reader. Refer to the National Instruments Product Manuals Library for updated documentation resources.

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

- *NI-IMAQdx Function Reference Help*—Contains reference information about the LabWindows/CVI and Microsoft Visual Basic functions for NI-IMAQdx driver software.
- *NI-IMAQdx User Manual*—Describes how to create image acquisition applications using the NI-IMAQdx driver software; the fundamentals of creating applications for Windows 2000/XP and LabVIEW Real-Time targets; and advanced programming concepts, such as broadcasting, triggering, and register-level programming.
- *NI Vision Acquisition Software Release Notes*—Outlines new functionality, system requirements, installation procedures, and descriptions of the documentation included with NI Vision Acquisition Software.
- *LabVIEW Help*—Contains information about using NI-IMAQ I/O devices with LabVIEW and the LabVIEW FPGA Module. This document also provides reference information for the NI-IMAQ I/O parameters and shutdown VIs used to create applications to control I/O lines on NI-IMAQ I/O devices.
- Your NI-IMAQ I/O device documentation

IMAQIOAttribute Constants

IMAQIOAttribute constants are the constants for the **attribute** parameter of the imaqIOGetAttribute and imaqIOSetAttribute methods in the NI-IMAQ I/O library.

Note Currently, these values are not settable.

- IMAQIO_ATTRIBUTE_PRODUCT_SELECT_CURRENT—Gets the value of the current product selection line. ISO Inputs 4 thru 0 are the product selection lines.
- IMAQIO_ATTRIBUTE_PRODUCT_SELECT_LATCHED—Gets the value of the currently latched product selection line. ISO Inputs 4 thru 0 are the product selection lines. ISO Input 5 latches in the data on ISO Inputs 4 thru 0.
- IMAQIO_ATTRIBUTE_QUADRATURE_ENCODER—Gets the value of the quadrature encoder. The quadrature encoder uses ISO Input 6 for its Phase A input and ISO Input 7 for its Phase B input.
- IMAQIO_ATTRIBUTE_WATCHDOG_STATUS—Gets the value of the watchdog status. 1 indicates that no errors occurred, and 0 indicates that a timeout occurred.
- IMAQIO_ATTRIBUTE_ISOPOWER_PRESENT—Gets the ISO power status. A value of 1 indicates that ISO power is present. A value of 0 indicates that ISO power is absent.

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	358 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