




QUICK REFERENCE

NI-SCOPE Instrument Driver






Initiate and Close Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
	niScope_init¹		Creates a new session to the instrument.
	ViRsrc	resourceName	For Traditional NI-DAQ devices, use DAQ::#, where # is the device number. For NI-DAQmx devices, the device name is assigned by Measurement & Automation Explorer (MAX). Optionally, for all devices you can use an IVI logical name.
	ViBoolean	IDQuery	<ul style="list-style-type: none">• NISCOPE_VAL_TRUE• NISCOPE_VAL_FALSE
	ViBoolean	resetDevice	<ul style="list-style-type: none">• NISCOPE_VAL_TRUE• NISCOPE_VAL_FALSE
	ViSession*	vi	Reference to the new session
	niScope_InitWithOptions		Creates a new session to the instrument and optionally sets the initial state of session properties.
	ViRsrc	resourceName	For Traditional NI-DAQ devices, use DAQ::#, where # is the device number. For NI-DAQmx devices, the device name is assigned by Measurement & Automation Explorer (MAX). Optionally, for all devices you can use an IVI logical name.
	ViBoolean	IDQuery	<ul style="list-style-type: none">• NISCOPE_VAL_TRUE• NISCOPE_VAL_FALSE
	ViBoolean	resetDevice	<ul style="list-style-type: none">• NISCOPE_VAL_TRUE• NISCOPE_VAL_FALSE
	ViString	Option String	Used to set the initial value of certain properties for the session, including range checking, caching, and simulation. Refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants.
	ViSession*	vi	Reference to the new session
	niScope_close		Closes the current session to the instrument.
	ViSession	vi	Session handle

¹ Function name for LabWindows™/CVI™, C, C++, and Visual Basic.

² In LabWindows/CVI, C, and C++, constant names such as NISCOPE_VAL_TRUE and NISCOPE_VAL_EDGE refer to the use of #defines in your program. In LabVIEW, these constants refer to Boolean or ring controls with corresponding entries. For example, NISCOPE_VAL_EDGE corresponds to the LabVIEW ring control entry **Edge**. Refer to *LabVIEW Help* (Show Help) for more details.

Configuration Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS	
	niScope_ConfigureAcquisition			
	Configures the digitizer acquisition mode.			
	ViSession	vi	Session handle	
	ViInt32	acquisitionType	<ul style="list-style-type: none"> NISCOPE_VAL_NORMAL NISCOPE_VAL_FLEXRES 	
	niScope_AutoSetup			
	Automatically configures the digitizer. When you call this function, the digitizer senses the input signal and configures common device settings.			
	ViSession	vi	Session handle	
	niScope_ConfigureVertical			
	Configures the vertical subsystem for the digitizer channel.			
		ViSession	vi	Session handle
		ViConstString	channelList	Channel to configure
		ViReal64	range	Volts peak-to-peak; legal values vary depending on product
		ViReal64	offset	Location of the center of the range parameter in volts
		ViInt32	coupling	<ul style="list-style-type: none"> NISCOPE_VAL_AC NISCOPE_VAL_DC NI_SCOPE_VAL_GND
		ViReal64	probeAttenuation	Any positive real number such as 1, 10, and 100
	ViBoolean	enabled	NISCOPE_VAL_TRUE, NISCOPE_VAL_FALSE	
	niScope_ConfigureChanCharacteristics			
	Configures the specified digitizer channel.			
		ViSession	vi	Session handle
		ViConstString	channelList	Channel to configure
	ViReal64	inputImpedance	<ul style="list-style-type: none"> NISCOPE_VAL_50_OHM NISCOPE_VAL_1_MEG_OHM 	
	ViReal64	maxInputFrequency	0—Results in the default value of the hardware; check your hardware documentation to find other supported bandwidths, in Hertz	
	niScope_ConfigureHorizontalTiming			
	Configures the horizontal subsystem for a single- or multi-record acquisition in terms of minimum sample rate.			
		ViSession	vi	Session handle
		ViReal64	minSampleRate	Minimum sampling rate; call niScope_SampleRate for the actual sample rate used
		ViInt32	minNumPts	Minimum number of points you need in the record for each channel; call niScope_ActualRecordLength for the actual record length acquired
		ViReal64	refPosition	Percent of waveform record that is pretriggered
		ViInt32	numRecords	Number of records to acquire
		ViBoolean	enforceRealtime	Indicates whether the digitizer enforces real-time measurements or allows equivalent-time measurements <ul style="list-style-type: none"> NISCOPE_VAL_TRUE NISCOPE_VAL_FALSE



niScope_ConfigureClock

Configures the properties for synchronizing the digitizer to an external reference clock or for sending the digitizer clock out as a synchronizing clock for other digitizers.

ViSession	vi	Session handle
ViConstString	inputClockSource	<ul style="list-style-type: none"> • NISCOPE_VAL_NO_SOURCE • NISCOPE_VAL_RTSL_CLOCK • NISCOPE_VAL_PFI_<1..2> • NISCOPE_VAL_EXTERNAL • NISCOPE_VAL_PXI_CLOCK • NISCOPE_VAL_CLK_IN
ViConstString	outputClockSource	<ul style="list-style-type: none"> • NISCOPE_VAL_NO_SOURCE • NISCOPE_VAL_RTSL_CLOCK • NISCOPE_VAL_PFI_<0..2> • NISCOPE_VAL_CLK_OUT
ViConstString	clockSyncPulseSource	<ul style="list-style-type: none"> • NISCOPE_VAL_NO_SOURCE • NISCOPE_VAL_RTSL_<0..6> • NISCOPE_VAL_PFI_<0..2>
ViBoolean	masterEnabled	<ul style="list-style-type: none"> • NISCOPE_VAL_TRUE • NISCOPE_VAL_FALSE

Triggering Functions

ICON

TYPE

PARAMETER

VALUE TO SET, COMMENTS



niScope_ConfigureTriggerImmediate

Configures the common properties for immediate triggering. Immediate triggering means the digitizer triggers itself.

ViSession	vi	Session handle
-----------	----	----------------



niScope_ConfigureTriggerEdge

Configures the edge trigger. An edge trigger occurs when the trigger signal passes through the voltage threshold that you specify with the level parameter. Its slope is specified with the slope parameter.

ViSession	vi	Session handle
ViConstString	triggerSource	"0", "1", NISCOPE_VAL_EXTERNAL
ViReal64	level	Voltage threshold for edge triggering
ViInt32	slope	<ul style="list-style-type: none"> • NISCOPE_VAL_POSITIVE • NISCOPE_VAL_NEGATIVE
ViInt32	triggerCoupling	<ul style="list-style-type: none"> • NISCOPE_VAL_AC • NISCOPE_VAL_DC • NISCOPE_VAL_LF_REJECT • NISCOPE_VAL_HF_REJECT • NISCOPE_VAL_AC_PLUS_HF_REJECT
ViReal64	holdoff	The length of time the digitizer waits, in seconds, after detecting a trigger or acquisition before enabling NI-SCOPE to detect another trigger
ViReal64	delay	The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data



niScope_ConfigureTriggerDigital

Configures the digital trigger. A digital trigger occurs when the trigger signal has the slope that you specify with the slope parameter.

ViSession	vi	Session handle
ViConstString	triggerSource	<ul style="list-style-type: none"> • NISCOPE_VAL_RTSL_<0..6> • NISCOPE_VAL_PFI_<0..2> • NISCOPE_VAL_PXI_STAR
ViInt32	slope	<ul style="list-style-type: none"> • NISCOPE_VAL_POSITIVE • NISCOPE_VAL_NEGATIVE

continued on p. 4

Triggering Functions (continued)

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	ViReal64	holdoff	The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger
	ViReal64	delay	The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data



niScope_ConfigureTriggerSoftware

Configures the common properties for software triggering.

Use `niScope_SendSoftwareTriggerEdge` to trigger the acquisition.

ViSession	vi	Session handle
ViReal64	holdoff	The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger
ViReal64	delay	The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data



niScope_ConfigureTriggerVideo

Configures the common properties for video triggering.

ViSession	vi	Session handle
ViConstString	triggerSource	"0", "1", NISCOPE_VAL_EXTERNAL
ViBoolean	enabledDCRestore	Enabling DC restore sets the zero volt reference of video signal to zero volts <ul style="list-style-type: none"> • NISCOPE_VAL_TRUE • NISCOPE_VAL_FALSE
ViInt32	signalFormat	<ul style="list-style-type: none"> • NISCOPE_VAL_NTSC • NISCOPE_VAL_PAL • NISCOPE_VAL_SECAM
ViInt32	event	<ul style="list-style-type: none"> • NISCOPE_VAL_TV_EVENT_FIELD1 • NISCOPE_VAL_TV_EVENT_FIELD2 • NISCOPE_VAL_TV_EVENT_ANY_LINE • NISCOPE_VAL_TV_EVENT_LINE_NUMBER
ViInt32	lineNumber	Line number to trigger on. For NTSC, the valid range is 1 to 525. For PAL and SECAM, the valid range is 1 to 625.
ViInt32	polarity	<ul style="list-style-type: none"> • NISCOPE_VAL_TV_POSITIVE • NISCOPE_VAL_TV_NEGATIVE
ViInt32	triggerCoupling	<ul style="list-style-type: none"> • NISCOPE_VAL_AC • NISCOPE_VAL_DC • NISCOPE_VAL_LF_REJECT • NISCOPE_VAL_HF_REJECT • NISCOPE_VAL_AC_PLUS_HF_REJECT
ViReal64	holdoff	The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger
ViReal64	delay	The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data



niScope_ConfigureTriggerHysteresis

Configures the hysteresis trigger. If the slope parameter is set to positive, a trigger occurs when a signal starts at a voltage below the level parameter minus the hysteresis parameter, and then crosses the voltage of the level parameter. If slope is set to negative, a trigger occurs when a signal starts at a voltage above the level parameter plus the hysteresis parameter, and then crosses the voltage of the level parameter.

ViSession	vi	Session handle
ViConstString	triggerSource	"0", "1", NISCOPE_VAL_EXTERNAL
ViReal64	level	Voltage threshold for the trigger
ViReal64	hysteresis	Size of the hysteresis window in volts
ViInt32	slope	<ul style="list-style-type: none"> NISCOPE_VAL_POSITIVE NISCOPE_VAL_NEGATIVE
ViInt32	triggerCoupling	<ul style="list-style-type: none"> NISCOPE_VAL_AC NISCOPE_VAL_DC NISCOPE_VAL_LF_REJECT NISCOPE_VAL_HF_REJECT NISCOPE_VAL_AC_PLUS_HF_REJECT
ViReal64	holdoff	The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger
ViReal64	delay	The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data



niScope_ConfigureTriggerWindow

Configures common properties for window triggering. A window trigger occurs when a signal enters or leaves a window you specify with the high level or low level parameters.

ViSession	vi	Session handle
ViConstString	triggerSource	"0", "1", NISCOPE_VAL_EXTERNAL
ViReal64	lowLevel	Voltage threshold you want the digitizer to use for the lower trigger level
ViReal64	highLevel	Voltage threshold you want the digitizer to use for the upper trigger level
ViInt32	windowMode	Specifies whether you want the trigger to occur when the signal enters or leaves a window <ul style="list-style-type: none"> NISCOPE_VAL_ENTERING_WINDOW NISCOPE_VAL_LEAVING_WINDOW
ViInt32	triggerCoupling	<ul style="list-style-type: none"> NISCOPE_VAL_AC NISCOPE_VAL_DC NISCOPE_VAL_LF_REJECT NISCOPE_VAL_HF_REJECT NISCOPE_VAL_AC_PLUS_HF_REJECT
ViReal64	holdoff	The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger
ViReal64	delay	The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data




niScope_SendSoftwareTriggerEdge

Sends the selected trigger to the digitizer. Call this function if you called **niScope_ConfigureTriggerSoftware** when you want the Reference trigger to occur. If you have configured an Acquisition Arm source, ArmReference Trigger source, or Advance Trigger source, call this function when you want to send the corresponding trigger to the digitizer.





ViSession	vi	Session handle
ViInt32	whichTrigger	<ul style="list-style-type: none"> NISCOPE_VAL_SOFTWARE_TRIGGER_START NISCOPE_VAL_SOFTWARE_TRIGGER_ARM_REFERENCE NISCOPE_VAL_SOFTWARE_TRIGGER_REFERENCE NISCOPE_VAL_SOFTWARE_TRIGGER_ADVANCE

continued on p. 6

Triggering Functions (continued)

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	niScope_ConfigureTriggerOutput		
			Configures the digitizer to generate a signal pulse that other digitizers configured for digital triggering can detect.
	ViSession	vi	Session handle
	ViInt32	triggerEvent	Condition that causes the digitizer to generate the signal pulse <ul style="list-style-type: none"> • NISCOPE_VAL_NO_EVENT • NISCOPE_VAL_STOP_TRIGGER_EVENT • NISCOPE_VAL_END_OF_RECORD_EVENT • NISCOPE_VAL_START_TRIGGER_EVENT • NISCOPE_VAL_END_OF_ACQUISITION_EVENT
	ViConstString	triggerOutput	Hardware terminal on which the signal pulse is generated <ul style="list-style-type: none"> • NISCOPE_VAL_NO_SOURCE • NISCOPE_VAL_RTSL_<0..6> • NISCOPE_VAL_PFI_<0..2> • NISCOPE_VAL_PXI_STAR

Acquisition Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	niScope_InitiateAcquisition		
			Initiates a waveform acquisition. After you call this function, the digitizer leaves the idle state and starts acquiring data. The digitizer acquires a waveform for each channel you have enabled with niScope_ConfigureVertical .
	ViSession	vi	Session handle
	niScope_Abort		
			Aborts an acquisition and returns the digitizer to the idle state. The niScope_Read or niScope_InitiateAcquisition functions initiate an acquisition.
	ViSession	vi	Session handle
	niScope_Read		
			Enables the specified channel, initiates an acquisition, waits for it to complete, and returns the scaled waveform.
			In LabVIEW, use niScope Read Cluster to acquire one waveform and niScope Multi Read Cluster to acquire multiple waveforms at one time.
	ViSession	vi	Session handle
	ViConstString	channelList	List of channels, separated by commas, to retrieve waveforms from
	ViReal64	timeout	Maximum length of time, in seconds, to wait for the data acquisition to complete
	ViInt32	numSamples	The number of samples to fetch for each waveform
	ViReal64*	wfm	An array containing the scaled waveforms from the digitizer
	niScope_wfmInfo*	wfmInfo	An array containing timing and scaling information for each returned waveform



niScope_FetchBinary8

Returns the waveform the digitizer acquires for the channel you specify in 8-bit binary form. The waveform is from a previously initiated acquisition.



In LabVIEW, use **niScope Fetch Binary 8** to fetch a single binary waveform or **niScope Multi Fetch Binary 8** to fetch multiple waveforms at one time.

For 16-bit binary form, use **niScope_FetchBinary16**. In LabVIEW, use **niScope Fetch Binary 16** or **niScope Multi Fetch Binary 16**.

For 32-bit binary form, use **niScope_FetchBinary32**. In LabVIEW, use **niScope Fetch Binary 32** or **niScope Multi Fetch Binary 32**.

ViSession	vi	Session handle
ViConstString	channelList	List of channels, separated by commas, to retrieve waveforms from
ViReal64	timeout	Maximum length of time, in seconds, to wait for the data acquisition to complete. Pass zero to return the currently available data
ViInt32	numSamples	The number of samples to fetch for each waveform
ViInt8*	wfm	An array containing the binary waveforms from the digitizer
niScope_wfmInfo*	wfmInfo	An array containing timing and scaling information for each returned waveform



niScope_Fetch

Returns the scaled waveform the digitizer acquires for the channel you specify. The waveform is from a previously initiated acquisition.



In LabVIEW, use **niScope Fetch Cluster** to fetch a single waveform or **niScope Multi Fetch Cluster** to fetch multiple waveforms at one time.

ViSession	vi	Session handle
ViConstString	channelList	List of channels, separated by commas, to retrieve waveforms from
ViReal64	timeout	Maximum length of time, in seconds, to wait for the data acquisition to complete. Pass zero to return the currently available data
ViInt32	numSamples	The number of samples to fetch for each waveform
ViReal64*	wfm	An array containing the scaled waveforms from the digitizer
niScope_wfmInfo*	wfmInfo	An array containing timing and scaling information for each returned waveform



niScope_Commit

Commits to hardware all the parameter settings associated with the task.

ViSession	vi	Session handle
-----------	----	----------------

Waveform Measurement Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
------	------	-----------	------------------------









niScope_AddWaveformProcessing

Adds one array measurement to the list of processing steps. This function completes before any other measurements.

ViSession	vi	Session handle
ViConstString	channelList	List of channels, separated by commas, to add processing to
ViInt32	measFunction	Array measurement to add; refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants

Waveform Measurement Functions (continued)

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	niScope_ClearWaveformProcessing		
	Clears the waveform processing on the channels you specify.		
	ViSession	vi	Session handle
	ViConstString	channelList	List of channels, separated by commas, to clear processing on
	niScope_ClearWaveformMeasurementStats		
	Clears the waveform measurement statistics for the channels and measurement you specify.		
	ViSession	vi	Session handle
	ViConstString	channelList	List of channels, separated by commas, to clear statistics from
	ViInt32	measFunction	Any measurement; refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants
	niScope_FetchMeasurement		
	Fetches the waveform from a previously initiated acquisition and performs the scalar measurement you specify.		
	In LabVIEW, use niScope Fetch Measurement to fetch a single scalar measurement and niScope Multi Fetch Measurement to fetch multiple measurements at one time.		
	ViSession	vi	Session handle
	ViConstString	channelList	List of channels, separated by commas, to retrieve waveform measurements from
	ViReal64	timeout	Maximum length of time, in seconds, to wait for the data acquisition to complete
	ViInt32	scalarMeasFunction	Scalar measurement to perform; refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants
	ViReal64*	result	An array containing the scalar measurement result for all the fetched waveforms
	niScope_FetchMeasurementStats		
	Fetches a waveform from a previously initiated acquisition and returns the statistics from multiple acquisitions. The statistics are updated once per acquisition if the measurement is fetched.		
	In LabVIEW, use niScope Fetch Measurement Stats to fetch a single scalar measurement and niScope Multi Fetch Measurement Stats to fetch multiple measurements at one time.		
	ViSession	vi	Session handle
	ViConstString	channelList	List of channels, separated by commas, to retrieve waveform measurements from
	ViReal64	timeout	Maximum length of time, in seconds, to wait for the data acquisition to complete
	ViInt32	scalarMeasFunction	Scalar measurement to perform; refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants
	ViReal64*	result	An array containing the scalar measurement result for all the fetched waveforms
	ViReal64*	mean	Mean of last numInStats measurements
	ViReal64*	stdev	Standard deviation of last numInStats measurements
	ViReal64*	min	Minimum of last numInStats measurements
	ViReal64*	max	Maximum of last numInStats measurements
	ViInt32*	numInStats	Number of measurements used for statistics



niScope_FetchArrayMeasurement

Fetches a waveform from a previously initiated acquisition and performs the specified array measurement.



In LabVIEW, use **niScope Fetch Array Measurement** or **niScope Fetch Array Measurement Cluster** to fetch a single measurement, or **niScope Multi Fetch Array Measurement** or **niScope Multi Fetch Array Measurement Cluster** to fetch multiple measurements at one time.

ViSession	vi	Session handle
ViConstString	channelList	List of channels, separated by commas, to retrieve waveform measurements from
ViReal64	timeout	Maximum length of time, in seconds, to wait for the data acquisition to complete
ViInt32	arrayMeasFunction	Array measurement to perform; refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants
ViInt32	measWfmSize	The number of samples to return for each waveform measurement in the measWfm array
ViReal64*	measWfm	An array containing the measurement waveforms
niScope_wfmInfo*	measWfmInfo	An array containing timing and scaling information for each returned waveform

Error Function

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
------	------	-----------	------------------------



niScope_errorMessage

Takes the returned error code and returns the interpretation as a user-readable string.

ViSession	vi	Session handle
ViStatus	errorCode	Error code returned by any NI-SCOPE function
ViChar	errorMessage [IVI_MAX_MESSAGE_BUF_SIZE]	Interpreted error code as a user-readable string

Actual Value Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
------	------	-----------	------------------------




niScope_ActualMeasWfmSize


Queries the number of samples in the resulting waveform measurement after performing the specified array measurement function.


Note: Not required for LabVIEW because the fetch VIs in LabVIEW default to fetch all data.


ViSession	vi	Session handle
ViConstString	arrayMeasFunction	Array measurement to perform; refer to <i>NI-SCOPE Function Reference Help</i> or <i>NI-SCOPE VI Reference Help</i> for constants
ViInt32*	measWfmSize	The number of samples available after performing the specified array measurement

Actual Value Functions (continued)


ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	niScope_ActualNumWfms Queries the number of waveforms available on a specified channel. Note: Not required for LabVIEW because the fetch VIs in LabVIEW default to fetch all data.		
	ViSession	vi	Session handle
	ViConstString	channelList	List of channels, separated by commas, from which to query the number of available waveforms
	ViInt32*	numWfms	The number of waveforms available for the specified channels

	niScope_ActualRecordLength Queries the actual number of points the digitizer acquires for each channel.		
	ViSession	vi	Session handle
	ViInt32*	actualRecordLength	Value is equal to or greater than the minNumPts you specify with niScope_ConfigureHorizontalTiming ; length of record is available for each channel








	niScope_SampleRate Queries the effective sample rate of the acquired waveform using the current configuration in samples per second.		
	ViSession	vi	Session handle
	ViReal64*	sampleRate	The effective sample rate of the waveform acquired for each channel

	niScope_SampleMode Queries the sample mode the digitizer is currently using.		
	ViSession	vi	Session handle
	ViInt32*	sampleMode	The sample mode the digitizer is currently using <ul style="list-style-type: none"> • NISCOPE_VAL_REAL_TIME • NISCOPE_VAL_EQUIVALENT_TIME

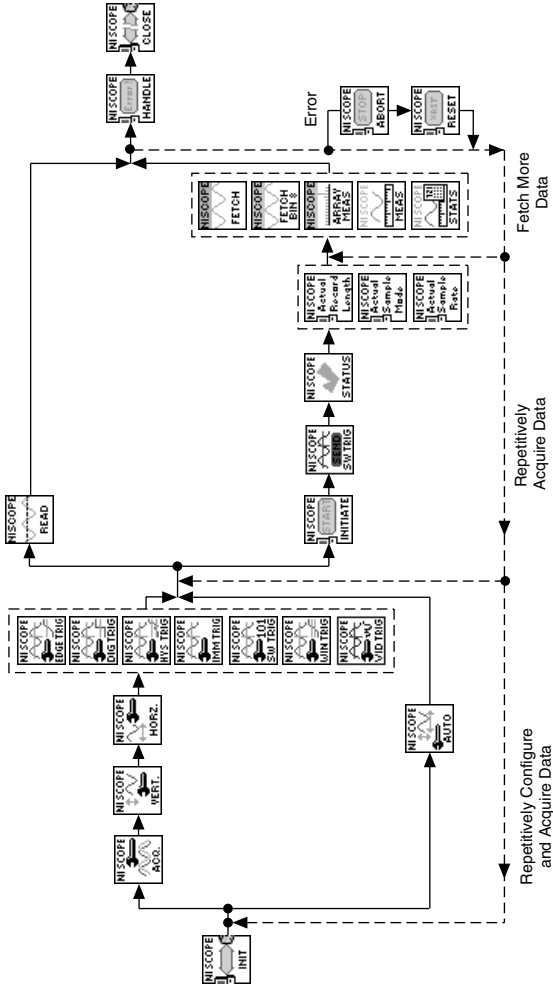
Check Status Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	niScope_AcquisitionStatus Shows whether an acquisition is in progress or complete.		
	ViSession	vi	Session handle
	ViInt32*	status	<ul style="list-style-type: none"> • NISCOPE_VAL_ACQ_IN_PROGRESS (0) • NISCOPE_VAL_ACQ_COMPLETE (1) • NISCOPE_VAL_ACQ_STATUS_UNKNOWN (-1)

Utility Functions

ICON	TYPE	PARAMETER	VALUE TO SET, COMMENTS
	niScope_reset		
	Resets the instrument to a known state.		
	ViSession	vi	Session handle
	niScope_self_test		
	Runs the instrument's self-test routine and returns the test result(s).		
	ViSession	vi	Session handle
	ViInt16*	selfTestResult	0–Self test passed, 1–Self test failed
	ViChar	selfTestMessage [IVI_MAX_MESSAGE_BUF_SIZE]	Self-test response string from the digitizer
	niScope_CalSelfCalibrate		
	Performs a self-calibration of all channels on the digitizer.		
	ViSession	vi	Session handle
	ViConstString	channelList	Currently ignored; use VI_NULL
	ViInt32	option	<ul style="list-style-type: none"> NISCOPE_VAL_SELF_CALIBRATION NISCOPE_VAL_RESTORE_FACTORY_CALIBRATION
	niScope_revision_query		
	Returns the revision numbers of the instrument driver and instrument firmware.		
	ViSession	vi	Session handle
	ViChar	driverRev [IVI_MAX_MESSAGE_BUF_SIZE]	Driver software revision number in the form of a string
	ViChar	instrRev [IVI_MAX_MESSAGE_BUF_SIZE]	Firmware revision number in the form of a string
	niScope_ProbeCompensationSignalStart		
	Starts the square wave output on PFI 1 for probe compensation.		
	ViSession	vi	Session handle
	niScope_ProbeCompensationSignalStop		
	Stops the square wave output on PFI 1 for probe compensation.		
	ViSession	vi	Session handle
	niScope_ResetDevice		
	Performs a hard reset on the device.		
	ViSession	vi	Session handle

Programming Flow



CVI™, IVI™, LabVIEW™, National Instruments™, NI™, ni.com™, NI-DAQ™, and RTSI™ are trademarks of National Instruments Corporation. Product and company names mentioned herein are trademarks or trade names of their respective companies. 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.

