

NI-SCOPE Driver Quick Reference Guide





Easy Programming for National Instruments High-Speed Digitizers



ICON	FUNCTION NAME AND DESCRIPTION ¹	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
Initiate and Close Functions				
	niScope_init Creates a new session to the instrument.	ViRsrc ViBoolean ViBoolean ViSession *	resourceName IDQuery resetDevice vi	DAQ::#, where # is the device number NISCOPE_VAL_TRUE, NISCOPE_VAL_FALSE NISCOPE_VAL_TRUE, NISCOPE_VAL_FALSE Reference to the new session
	niScope_close Closes the current session to the instrument.	ViSession	vi	Session handle
Configuration Functions				
	niScope_ConfigureAcquisition Configures the digitizer acquisition mode.	ViSession ViInt32	vi acquisitionType	Session handle NISCOPE_VAL_NORMAL, NISCOPE_VAL_FLEXRES, NISCOPE_VAL_DDC
	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 ViConstString ViReal64 ViReal64 ViInt32 ViReal64 ViBoolean	vi channelList range offset coupling probeAttenuation enabled	Session handle Channel to configure Volts peak-to-peak; legal values vary depending on product Location of the center of the range parameter in volts NISCOPE_VAL_AC, NISCOPE_VAL_DC Any positive real number such as 1, 10, and 100 NISCOPE_VAL_TRUE, NISCOPE_VAL_FALSE
	niScope_ConfigureChanCharacteristics Configures the specified digitizer channel.	ViSession ViConstString ViReal64 ViReal64	vi channelList inputImpedance maxInputFrequency	Session handle Channel to configure NISCOPE_VAL_50_OHM, NISCOPE_VAL_1_MEG_OHM 0—Results in the default value of the hardware; check your hardware user manual to find other supported bandwidths, in hertz

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







² In C, C++, and LabWindows/CVI, 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 online help (Show Help) for more details.









ICON	FUNCTION NAME AND DESCRIPTION ¹	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
	niScope_ConfigureHorizontalTiming Configures the horizontal subsystem for a multi-record acquisition in terms of minimum sample rate or time per record.	ViSession	vi	Session handle
		ViReal64	minSampleRate	Minimum sampling rate
		ViInt32	minNumPts	Minimum number of points you need in the record for each channel; call niScope_ActualRecordLength for the actual acquired record length
		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 NISCOPE_VAL_TRUE, NISCOPE_VAL_FALSE
	niScope_ConfigureClock Configures the properties for synchronizing the digitizer to an external clock or for sending the digitizer clock out as a synchronizing clock for other digitizers.	ViSession	vi	Session handle
		ViConstString	inputClockSource	NISCOPE_VAL_NO_SOURCE, NISCOPE_VAL_RTSI_CLOCK, NISCOPE_VAL_PFI_<1..2>, NISCOPE_VAL_EXTERNAL, NISCOPE_VAL_PXI_CLOCK
		ViConstString	outputClockSource	NISCOPE_VAL_NO_SOURCE, NISCOPE_VAL_RTSI_CLOCK, NISCOPE_VAL_PFI_<1..2>
		ViConstString	clockSyncPulseSource	NISCOPE_VAL_NO_SOURCE, NISCOPE_VAL_RTSI_<0..6>, NISCOPE_VAL_PFI_<1..2>
		ViBoolean	masterEnabled	NISCOPE_VAL_TRUE, NISCOPE_VAL_FALSE










Triggering Functions

	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; NISCOPE_VAL_NO_EVENT, NISCOPE_VAL_STOP_TRIGGER_EVENT, NISCOPE_VAL_END_OF_RECORD_EVENT
		ViConstString	triggerOutput	Hardware terminal on which the signal pulse will be generated; NISCOPE_VAL_NO_SOURCE, NISCOPE_VAL_RTSI_<0..6>, NISCOPE_VAL_PFI_<1..2>, NISCOPE_VAL_PXI_STAR
	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 ³	"0" "1" NISCOPE_VAL_EXTERNAL
		ViReal64	level	Voltage threshold for edge triggering
		ViInt32	slope	NISCOPE_VAL_POSITIVE, NISCOPE_VAL_NEGATIVE
		ViInt32	triggerCoupling	NISCOPE_VAL_AC, NISCOPE_VAL_DC
		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


³In LabVIEW, triggerSource is implemented as a string control. LabVIEW 5.1 and any subsequent versions implement a ring control for this parameter. However, if you use LabVIEW 5.0, you must enter the specific string constant. Refer to the LabVIEW online help (Show Help) for these constant names.






ICON	FUNCTION NAME AND DESCRIPTION ¹	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
	<p>niScope_ConfigureTriggerHysteresis</p> <p>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 plus the hysteresis parameter and then crosses the voltage of the level parameter.</p>	ViSession ViConstString ViReal64 ViReal64 ViInt32 ViInt32 ViReal64 ViReal64	vi triggerSource ³ level hysteresis slope triggerCoupling holdoff delay	Session handle "0", "1" NISCOPE_VAL_EXTERNAL Voltage threshold for the trigger Size of the hysteresis window in volts NISCOPE_VAL_POSITIVE, NISCOPE_VAL_NEGATIVE NISCOPE_VAL_AC, NISCOPE_VAL_DC The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data
	<p>niScope_ConfigureTriggerDigital</p> <p>Configures the digital trigger. A digital trigger occurs when the trigger signal has the slope that you specify with the slope parameter.</p>	ViSession ViConstString ViInt32 ViReal64 ViReal64	vi triggerSource ³ slope holdoff delay	Session handle NISCOPE_VAL_RTSL_<0..6>, NISCOPE_VAL_PFI_<1..2>, NISCOPE_VAL_PXI_STAR NISCOPE_VAL_POSITIVE, NISCOPE_VAL_NEGATIVE The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data
	<p>niScope_ConfigureTriggerImmediate</p> <p>Configures the common properties for immediate triggering. Immediate triggering means the digitizer triggers itself.</p>	ViSession	vi	Session handle
	<p>niScope_ConfigureTriggerSoftware</p> <p>Configures the common properties for software triggering. Use <code>niScope_SendSWTrigger</code> to trigger the acquisition.</p>	ViSession ViReal64 ViReal64	vi holdoff delay	Session handle The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data
	<p>niScope_SendSWTrigger</p> <p>Sends a command to trigger the digitizer; Use <code>niScope_ConfigureTriggerSoftware</code> first to configure the trigger.</p>	ViSession	vi	Session handle
	<p>niScope_ConfigureTriggerWindow</p> <p>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.</p>	ViSession ViConstString ViReal64 ViReal64 ViInt32 ViInt32 ViReal64 ViReal64	vi triggerSource ³ lowLevel highLevel windowMode triggerCoupling holdoff delay	Session handle "0", "1" NISCOPE_VAL_EXTERNAL Voltage threshold you want the digitizer to use for the lower trigger level Voltage threshold you want the digitizer to use for the upper trigger level Specifies if you want the trigger to occur when the signal enters or leaves a window; NISCOPE_VAL_ENTERING_WINDOW NISCOPE_VAL_LEAVING_WINDOW NISCOPE_VAL_AC, NISCOPE_VAL_DC The length of time the digitizer waits, in seconds, after detecting a trigger before enabling NI-SCOPE to detect another trigger The length of time the digitizer waits, in seconds, after it receives the trigger to start acquiring posttrigger data

ICON	FUNCTION NAME AND DESCRIPTION ¹	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
	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 ViConstString ViReal64 ViInt32 ViReal64 * niScope_wfmInfo *	vi channelList timeout numSamples wfm wfmInfo	Session handle List of channels, separated by commas, to retrieve waveforms from Maximum length of time, in seconds, to wait for the data acquisition to complete The number of samples to fetch for each waveform An array containing the scaled waveforms from the digitizer 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 ViConstString ViReal64 ViInt32 ViInt8 * niScope_wfmInfo *	vi channelList timeout numSamples wfm wfmInfo	Session handle List of channels, separated by commas, to retrieve waveforms from Maximum length of time, in seconds, to wait for the data acquisition to complete. Pass zero to return the currently available data The number of samples to fetch for each waveform An array containing the binary waveforms from the digitizer 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 ViConstString ViReal64 ViInt32 ViReal64 * niScope_wfmInfo *	vi channelList timeout numSamples wfm wfmInfo	Session handle List of channels, separated by commas, to retrieve waveforms from Maximum length of time, in seconds, to wait for the data acquisition to complete. Pass zero to return the currently available data The number of samples to fetch for each waveform An array containing the scaled waveforms from the digitizer An array containing timing and scaling information for each returned waveform


ICON	FUNCTION NAME AND DESCRIPTION ¹	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
Waveform Measurement Functions				
	niScope_AddWaveformProcessing Adds one array measurement to the list of processing steps. It is completed before any other measurements.	ViSession ViConstString ViInt32	vi channelList measFunction	Session handle List of channels, separated by commas, to add processing to Array measurement to add; see <i>NI-SCOPE Function Reference Help</i> for constants
	niScope_ClearWaveformProcessing Clears the waveform processing on the channels you specify.	ViSession ViConstString	vi channelList	Session handle 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 ViConstString ViInt32	vi channelList measFunction	Session handle List of channels, separated by commas, to clear statistics from Any measurement; see <i>NI-SCOPE Function Reference Help</i> for constants
	niScope_FetchMeasurement Fetches the waveform from a previously initiated acquisition and performs the scalar measurement you specify.	ViSession ViConstString ViReal64	vi channelList timeout	Session handle List of channels, separated by commas, to retrieve waveform measurements from Maximum length of time, in seconds, to wait for the data acquisition to complete.
	In LabVIEW, use niScope Fetch Measurement to fetch a single scalar measurement and niScope Multi Fetch Measurement to fetch multiple measurements at one time.	ViInt32 ViReal64 *	scalarMeasFunction result	Scalar measurement to perform; see <i>NI-SCOPE Function Reference Help</i> for constants 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.	ViSession ViConstString ViReal64 ViInt32 ViReal64 *	vi channelList timeout scalarMeasFunction result	Session handle List of channels, separated by commas, to retrieve waveform measurements from Maximum length of time, in seconds, to wait for the data acquisition to complete. Scalar measurement to perform; see <i>NI-SCOPE Function Reference Help</i> for constants An array containing the scalar measurement result for all the fetched waveforms
	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.	ViReal64 * ViReal64 * ViReal64 * ViReal64 * ViInt32 *	mean stdev min max numInStats	Mean of last numInStats measurements Standard deviation of last numInStats measurements Minimum of last numInStats measurements Maximum of last numInStats measurements Number of measurements used for statistics
	niScope_FetchArrayMeasurement Fetches a waveform from a previously initiated acquisition and performs the specified array measurement.	ViSession ViConstString ViReal64 ViInt32	vi channelList timeout arrayMeasFunction	Session handle List of channels, separated by commas, to retrieve waveform measurements from Maximum length of time, in seconds, to wait for the data acquisition to complete. Array measurement to perform; see <i>NI-SCOPE Function Reference Help</i> for constants
	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.	ViInt32 ViReal64 * niScope_wfmlnfo *	measWfmSize measWfm measWfmInfo	The number of samples to return for each waveform measurement in the measWfm array An array containing the measurement waveforms An array containing timing and scaling information for each returned waveform

Error Function







	niScope_errorHandler Translates an error code and its source into a detailed error description.	ViSession ViInt32 ViChar ViChar	vi errorCode errorSource [MAX_FUNCTION_NAME_SIZE] errorDescription [MAX_ERROR_DESCRIPTION]	Session handle Error code to translate Function returning the error code; can be VI_NULL Translated description
------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------	------------------------------------------	-----------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------

ICON	FUNCTION NAME AND DESCRIPTION ¹	TYPE	PARAMETER	VALUE TO SET, COMMENTS ²
	niScope_ActualMeasWfmSize Queries the number of samples in the resulting waveform measurement after performing the specified array measurement function.	ViSession ViConstString ViInt32 *	vi arrayMeasFunction measWfmSize	Session handle Array measurement to perform; see <i>NI-SCOPE Function Reference Help</i> for constants The number of samples available after performing the specified array measurement
	niScope_ActualNumWfms Queries the number of waveforms available on a specified channel.	ViSession ViConstString ViInt32 *	vi channelList numWfms	Session handle List of channels, separated by commas, from which to query the number of available waveforms The number of waveforms available for the specified channels
	niScope_ActualRecordLength Queries the actual number of points the digitizer acquires for each channel.	ViSession ViInt32 *	vi actualRecordLength	Session handle 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 ViReal64 *	vi sampleRate	Session handle The effective sample rate of the waveform acquired for each channel
	niScope_SampleMode Queries the sample mode the digitizer is currently using.	ViSession ViInt32 *	vi sampleMode	Session handle The sample mode the digitizer is currently using. NISCOPE_VAL_REAL_TIME, NISCOPE_VAL_EQUIVALENT_TIME

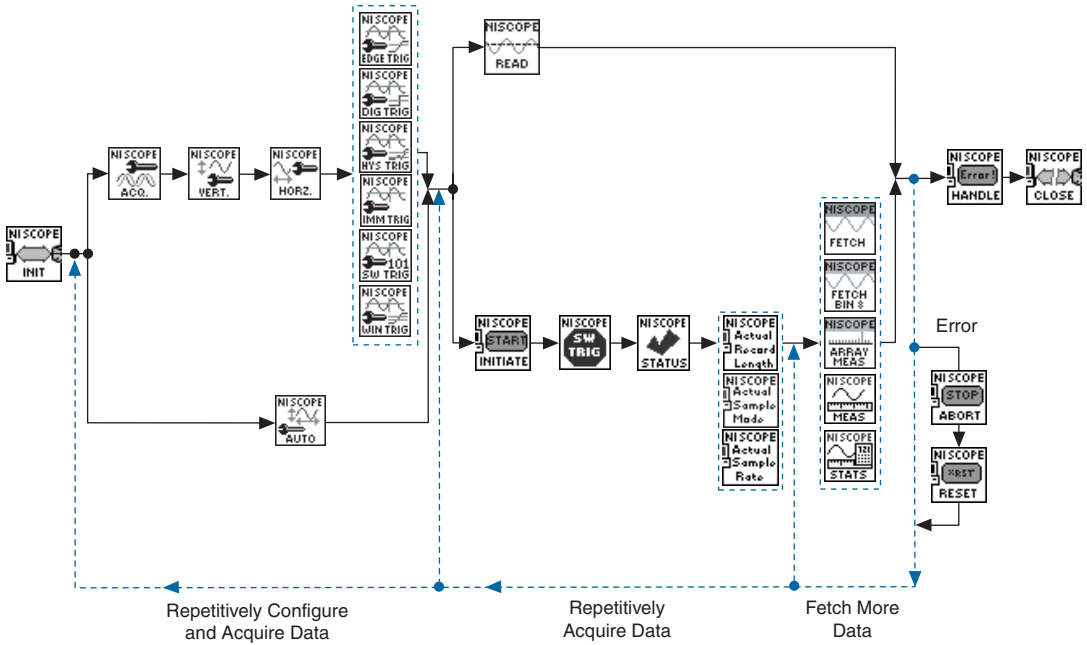
Check Status Functions

	niScope_AcquisitionStatus Shows if an acquisition is in progress or complete.	ViSession ViInt32 *	vi status	Session handle NISCOPE_VAL_ACQ_IN_PROGRESS (0) NISCOPE_VAL_ACQ_COMPLETE (1), NISCOPE_VAL_ACQ_STATUS_UNKNOWN (-1)
----------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------	------------------------	--------------	---------------------------------------------------------------------------------------------------------------------------

Utility Functions

	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 ViInt16 * ViChar	vi selfTestResult selfTestMessage [VI_MAX_MESSAGE_BUF_SIZE]	Session handle 0–Self test passed, 1–Self test failed Self-test response string from the digitizer; see your device user manual for an explanation of the string's contents
	niScope_CalSelfCalibrate Performs a self-calibration of all channels on the digitizer.	ViSession ViConstString ViInt32	vi channelList option	Session handle Currently ignored; use VI_NULL NISCOPE_VAL_SELF_CALIBRATION, NISCOPE_VAL_RESTORE_FACTORY_CALIBRATION
	niScope_revision_query Returns the revision numbers of the instrument driver and instrument firmware.	ViSession ViChar ViChar	vi driverRev[VI_MAX_MESSAGE_BUF_SIZE] instrRev[VI_MAX_MESSAGE_BUF_SIZE]	Session handle Driver software revision numbers in the form of a string Firmware revision numbers 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

Programming Flow





322575C-01

Jun01