

NI 5102

Introduction

This document contains information and step-by-step instructions for calibrating the National Instruments (NI) 5102 digitizer. This calibration procedure is intended for metrology labs. It includes specific programming instructions necessary for external calibration of the NI 5102 using Measurement Studio, LabVIEW, C, or Visual Basic programming environments.

What Is Calibration?

Calibration consists of verifying the measurement accuracy of a device and correcting for any measurement error. *Verification* is measuring the performance of a device and comparing the results to the factory specifications of the device. NI calibrates every NI 5102 digitizer at the factory. During the factory calibration process, the calibration constants are stored on the EEPROM. These values are loaded from memory and used as needed by the digitizer.

NI digitizers have three types of calibration: external, internal, and external restore. These three types of calibration are described in the next three sections.

External Calibration

External calibration is generally performed with a high-precision oscilloscope calibrator at either NI or a metrology lab. This procedure replaces all calibration constants in the EEPROM and is equivalent to a factory calibration at NI. Because the external calibration procedure changes all EEPROM constants, it invalidates the original National Institute of Standards and Technology (NIST)-traceability certificate. If an external calibration is done with a NIST-certified voltage source, a new NIST-traceability certificate can be issued.

Self-Calibration

Self-calibration, or internal calibration, uses a software command and requires no external connections.

External Restore Calibration

External restore calibration restores the previous external calibration settings for any calibration constants computed during external calibration or self-calibration. You should use this option only during a self-calibration failure. For example, if you lose power during a self-calibration, the digitizer may be rendered unusable. External restore allows the digitizer to function for a self-calibration.

Why Should You Calibrate?

The accuracy of electronic components drifts with time and temperature, which can affect measurement accuracy as a device ages. Calibration restores your digitizer to its specified accuracy and ensures that it still meets NI standards.

How Often Should You Calibrate?

The measurement accuracy requirements of your application determine how often you should externally calibrate your NI 5102 digitizer. National Instruments recommends that you perform a complete calibration at least once every year. You can shorten this interval to 90 days or six months based on the demands of your application.

You can also use the verification procedure at a regular interval to determine if your digitizer needs adjustment.

Equipment and Other Test Requirements

This section describes the equipment, documentation, software, and test conditions required for calibrating your NI 5102.

Test Equipment

Table 1 contains a list of equipment required for calibrating your NI 5102. If you do not have the recommended instruments, use these specifications to select a substitute calibration standard.

Table 1. Required Equipment Specifications for NI 5102 Calibration

Required Equipment	Recommended Equipment	Parameter Measured	Necessary Specifications
Signal Generator/ Ohmmeter	Wavetek 9500 Scope Calibrator	Vertical Gain	DC ± 40 mV to ± 4.5 V, $\pm 0.1\%$ into $1\text{ M}\Omega$
		AC Coupling	Sine wave 9–13 Hz ± 100 ppm, $1.8\text{ V}_{pp} \pm 2\%$ into $1\text{ M}\Omega$
		Bandwidth	1.5% amplitude flatness for leveled sine wave 100 kHz–20 MHz ± 50 ppm, $1.5\text{ V}_{pp} \pm 2\%$ into $1\text{ M}\Omega$
		Input Impedance	2-wire resistance accuracy of 0.25% for $1\text{ M}\Omega$ measurement
BNC Cable	—	—	$50\ \Omega$
BNC Short-Circuit Cap	—	Vertical Offset	0 VDC ± 0.1 mV

Test Conditions

Follow these guidelines to optimize the connections and the environment during calibration:

- Keep connections to the NI 5102 short. Long cables and wires act as antennae, picking up extra noise that can affect measurements.
- Use a $50\ \Omega$ BNC coaxial cable for all connections to the digitizer.
- Keep relative humidity between 10 and 90%, noncondensing, or consult your digitizer hardware manual for the optimum relative humidity.

- Maintain the temperature between 5 and 40 °C, or consult your digitizer hardware manual for the optimum temperature range.
- Allow a warm-up time of at least 15 minutes to ensure that the measurement circuitry of the NI 5102 is at a stable operating temperature.

Documentation

This section describes the documentation you need to calibrate your NI 5102 digitizer. In addition to this calibration procedure, you may need to refer to the following documents:

- *NI-SCOPE Quick Reference Guide*
- *Where to Start with Your NI Digitizer*
- *NI 5102 User Manual*

You can download these documents from the NI Web site at ni.com/manuals.

Software

This section describes the software you need to calibrate your NI 5102 digitizer. Unless otherwise specified, calibration functions are C function calls in the NI-SCOPE driver, which you can download from the NI Web site at ni.com. These function calls are also valid for Visual Basic programs. While LabVIEW virtual instruments (VIs) are not discussed in this procedure because many LabVIEW VIs have the same names as the listed NI-SCOPE function calls. Many of the functions use constants defined in the `niScopeCal.h` file. To use these constants, you must include `niScopeCal.h` in your code when you write your calibration procedure.

To install the calibration software, complete the following steps:

1. Launch the NI-SCOPE installer from the NI-SCOPE CD.
2. Select **Programmatic and Interactive Support** as the installation type.
3. Click the **Components** button in lower left corner of the Application Development Environments screen.
4. Check the box labelled **External Calibration Support**.
5. Complete the installation of NI-SCOPE by following the instructions in your NI-SCOPE documentation.

Writing Your Calibration Procedure

The latest version of NI-SCOPE includes all the functions necessary for calibrating NI digitizers. Because calibration support is included in `niScope_32.dll`, you can access it through any compiler capable of calling into a DLL. If you use a C compiler, include the `niScopeCal.h` header file, which defines all calibration-specific functions and briefly explains the parameters. With Measurement Studio, the NI-SCOPE function panel `niScopeCal.fp` provides further help on these functions. LabVIEW support is installed in `niScopeCal.llb`, and all calibration functions appear in the function palette. See Table 2 for file locations.

Table 2. Calibration File Location after Installing NI-SCOPE 1.6 or Later

File Name and Location	Description
<code>vxipnp\winxx\ niScope_32.dll</code>	NI-SCOPE driver containing the entire NI-SCOPE API, including calibration functions
<code>vxipnp\winxx\ lib\msc\niScope_32.lib</code>	NI-SCOPE library containing the entire NI-SCOPE API, including calibration functions
<code>LabView\examples\instr\ niScopeExamples</code>	LabVIEW VI library containing NI-SCOPE examples, including self calibration; you can access calibration examples from the function palette in LabVIEW
<code>LabView\instr.lib\niscope\ calibrateniScopeCal.llb</code>	LabVIEW VI library containing VIs for calling the NI-SCOPE calibration API; you can access calibration functions from the function palette in LabVIEW
<code>vxipnp\winxx\include\ niScopeCal.h</code>	Calibration header file, which must be included in any C program accessing calibration functions; this file automatically includes <code>niScope.h</code> , which defines the rest of the NI-SCOPE interface
<code>vxipnp\winxx\niScope\ niScope.fp</code>	CVI function panel file, which includes function prototypes and help using NI-SCOPE in the CVI environment
<code>vxipnp\winxx\niScope\ niScopeCal.fp</code>	CVI function panel file, which includes function prototypes and help using NI-SCOPE in the CVI environment for external calibration
<code>vxipnp\winxx\niScope\ examples\</code>	Directory of NI-SCOPE examples for CVI, MSVC, Visual C++, and Visual Basic, including the self-calibration examples in CVI and MSVC

Self-Calibration Procedure

Self-calibrate your digitizer before you begin externally calibrating it. NI-SCOPE for LabVIEW, CVI, and MSVC includes self-calibration example programs. Table 2 shows the filenames and their locations.



Note Self-calibrating the NI 5102 adjusts *only* the timing for random interleaved sampling (RIS) acquisitions.

Self-Calibrating Your NI 5102

To self-calibrate your digitizer, complete the following steps:

1. Call `niScope_init` to obtain an instrument session handle.
2. Call `niScope_calSelfCalibrate` with **option** set to `VI_NULL`. The new calibration constants are immediately stored in the EEPROM, so you can include this procedure in any application that uses the digitizer.
3. Call `niScope_close` to close the session handle.

External Calibration Procedures

The external calibration procedure consists of three steps: verifying the performance of your digitizer, adjusting the calibration constants, and verifying again after the adjustments. All procedures start by calling `niScope_init` with **resetDevice** set to `NISCOPE_VAL_TRUE` and end by calling `niScope_close`. Table 1 lists the accuracies of input stimuli for specific devices.

Verifying the Performance of Your NI 5102

To determine if your NI 5102 requires adjustments, verify each vertical offset and vertical gain specification listed in Table 3. If any of these tests fail, externally calibrate your digitizer. If any tests fail immediately after you perform an external calibration, return your digitizer to NI for repair.

Verifying Vertical Offset

Complete the following steps for each vertical offset entry in Table 3 and each channel:

1. Short-circuit the input of the digitizer with the BNC short-circuiting cap.
2. Call `niScope_ConfigureAcquisition` with **acquisitionType** set to `NISCOPE_VAL_NORMAL`.
3. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = 20,000,000 S/s
 - **minRecordLength** = 30,000
 - **referencePosition** = 50.0
4. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = The digitizer parameter value for the first vertical offset entry in Table 3
 - **offset** = 0.0
 - **coupling** = `NISCOPE_VAL_DC`
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`
5. Wait 10 ms for the input stage to settle.
6. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_VOLTAGE_AVERAGE`.
7. Compare the resulting average voltage to the success condition listed in Table 3. If it is outside the range of the success condition, your NI 5102 has failed this portion of the verification. You must externally calibrate your digitizer to guarantee its accuracy. If your digitizer fails after an external calibration, return it to NI for repair.
8. Repeat steps 2 through 7 for each channel and vertical offset entry in Table 3.

You have now finished verifying the vertical offset.

Verifying Vertical Gain

Complete the following steps for each vertical gain entry in Table 3 and each channel:

1. Connect the source to the high-impedance input of the digitizer channel.
2. Call `niScope_ConfigureAcquisition` with **acquisitionType** set to `NISCOPE_VAL_NORMAL`.
3. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = 20,000,000 S/s
 - **minRecordLength** = 30,000
 - **referencePosition** = 50.0
4. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = The digitizer parameter value for the first vertical gain entry in Table 3
 - **offset** = 0.0
 - **coupling** = `NISCOPE_VAL_DC`
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`Wait 10 ms for the input stage to settle.
5. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance** = `NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth** = `NISCOPE_VAL_FULL_BANDWIDTH` (or 0.0 in LabVIEW)
6. Apply the positive DC stimulus voltage for the first vertical gain entry listed in Table 3.
7. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_VOLTAGE_AVERAGE`.
8. Apply the negative DC stimulus voltage for the first vertical gain entry listed in Table 3.
9. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_VOLTAGE_AVERAGE`.

10. Compute the error in the vertical gain using the formula:

$$error = (a - b) - (c - d)$$

where a is the measured positive voltage, b is the measured negative voltage, c is the applied positive voltage, and d is the applied negative voltage.

11. Compare the error to the success condition in the table. If the error is less than the success condition value, your NI 5102 has passed this portion of the vertical gain test. Otherwise, externally calibrate your digitizer. If your digitizer fails after an external calibration, return it to NI for repair.

12. Repeat steps 2 through 11 for each channel and vertical gain entry in Table 3.

You have now finished verifying the vertical gain.

Table 3. NI 5102 Vertical Offset and Vertical Gain Specifications

Name	Digitizer Parameters	Stimulus Parameters	Success Condition
Vertical Offset	range = 10 V	Short-Circuit Input	$ x < 0.05800 \text{ V}$
Vertical Offset	range = 2 V	Short-Circuit Input	$ x < 0.01100 \text{ V}$
Vertical Offset	range = 0.5 V	Short-Circuit Input	$ x < 0.00290 \text{ V}$
Vertical Offset	range = 0.1 V	Short-Circuit Input	$ x < 0.00058 \text{ V}$
Vertical Gain	range = 10 V	$\pm 4.5 \text{ VDC}$	$ x < 0.1 \text{ V}$
Vertical Gain	range = 10 V	$\pm 1.5 \text{ VDC}$	$ x < 0.1 \text{ V}$
Vertical Gain	range = 2 V	$\pm 0.9 \text{ VDC}$	$ x < 0.02 \text{ V}$
Vertical Gain	range = 2 V	$\pm 0.5 \text{ VDC}$	$ x < 0.02 \text{ V}$
Vertical Gain	range = 0.5 V	$\pm 0.22 \text{ VDC}$	$ x < 0.005 \text{ V}$
Vertical Gain	range = 0.5 V	$\pm 0.1 \text{ VDC}$	$ x < 0.005 \text{ V}$
Vertical Gain	range = 0.1 V	$\pm 0.045 \text{ VDC}$	$ x < 0.001 \text{ V}$
Vertical Gain	range = 0.1 V	$\pm 0.007 \text{ VDC}$	$ x < 0.001 \text{ V}$

Verifying Large Signal Bandwidth

To verify large signal bandwidth, complete the following steps for the bandwidth entries in Table 4, and for each channel and coupling mode:

1. Connect the BNC T connector to your digitizer, and connect the signal generator and 50 Ω terminator to the two inputs on the T connector.
2. Set the signal generator to the frequency and amplitude listed in Table 4 for the reference bandwidth entry.
3. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = 2 V
 - **offset** = 0
 - **coupling** = `NISCOPE_VAL_DC` or `NISCOPE_VAL_AC` (you need to test both)
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`

Wait 300 ms for the input stage to settle.

4. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance** = `NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth** = `NISCOPE_VAL_FULL_BANDWIDTH`
5. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = The digitizer parameter value for the Reference Bandwidth entry in Table 4
 - **minRecordLength** = 30,000
 - **referencePosition** = 50.0
6. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AC_ESTIMATE`.
Record this value; it is the reference AC estimate used in step 10.
7. Apply the signal specified in the second bandwidth entry in Table 4.
8. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = The digitizer parameter value for the second bandwidth entry in Table 4
 - **minRecordLength** = 30,000
 - **referencePosition** = 50.0

9. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AC_ESTIMATE`.
Record this value; it is the AC estimate used in step 10.
10. Compute the response in decibels using the formula:

$$response = (20\log_{10})\left[\frac{AC\ estimate}{reference\ AC\ estimate}\right]$$

11. Compare *response* to the success condition range in Table 4.
If *response* is outside the given range, your NI 5102 has failed this test.
Return your digitizer to NI for repair.
12. Repeat steps 7 through 11 for the last bandwidth entry in Table 4.

You have now finished verifying the large signal bandwidth.

Table 4. NI 5102 Large Signal Bandwidth Specifications

Name	Digitizer Parameters	Stimulus Parameters	Success Condition
Reference Bandwidth (Large Signal)	minSampleRate = 20,000,000 S/s bandwidth = NISCOPE_VAL_FULL_BANDWIDTH (0.0 in LabVIEW)	100 kHz, 1.9 Vpp	—
Bandwidth (Large Signal)	minSampleRate = 1,000,000 S/s	999,500 Hz, 1.9 Vpp (measurement is intentionally aliased)	$ x < 3\text{ dB}$
Bandwidth (Large Signal)	minSampleRate = 10,000,000 S/s	9.9 MHz, 1.9 Vpp (measurement is intentionally aliased)	$ x < 3\text{ dB}$

Verifying Small Signal Bandwidth

To verify small signal bandwidth, complete the following steps for each reference bandwidth entry in Table 5, and for each channel and coupling mode:

1. Connect the BNC T connector to your digitizer, and connect the signal generator and 50 Ω terminator to the two inputs on the T connector.
2. Set the signal generator to the frequency and amplitude listed in Table 5 for the Reference Bandwidth entry.
3. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = 2 V
 - **offset** = 0
 - **coupling** = `NISCOPE_VAL_DC` or `NISCOPE_VAL_AC` (you need to test both)
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`

Wait 300 ms for the input stage to settle.

4. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance** = `NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth** = `NISCOPE_VAL_FULL_BANDWIDTH`
5. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = The scope parameter value for the reference bandwidth entry in Table 5
 - **minRecordLength** = 30,000
 - **referencePosition** = 50.0
6. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AC_ESTIMATE`.
Record this value; it is the reference AC estimate used in step 10.
7. Apply the signal specified in the Stimulus Parameter column in Table 5.

8. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = The digitizer parameter value for the second bandwidth entry in Table 5
 - **minRecordLength** = 30,000
 - **referencePosition** = 50.0
9. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AC_ESTIMATE`.
Record this value; it is the AC estimate used in step 10.
10. Compute the response in decibels using the formula: Compare *response* to the success condition range in Table 5. If *response* is outside the given range, your NI 5102 has failed this test. Return your digitizer to NI for repair.

$$response = (20\log_{10})\left[\frac{AC\ estimate}{reference\ AC\ estimate}\right]$$

11. Repeat steps 7 through 10 for the last bandwidth entry in Table 5.

You have now finished verifying the small signal bandwidth.

Table 5. NI 5102 Small Signal Bandwidth Specifications

Name	Digitizer Parameters	Stimulus Parameters	Success Condition
Reference Bandwidth (Small Signal)	minSampleRate = 20,000,000 S/s bandwidth = <code>NISCOPE_VAL_FULL_BANDWIDTH</code> (0.0 in LabVIEW)	100 kHz, 0.2 Vpp	—
Bandwidth (Small Signal)	minSampleRate = 1,000,000 S/s	999,500 Hz, 0.2 Vpp (measurement is intentionally aliased)	$ x < 3\text{ dB}$
Bandwidth (Small Signal)	minSampleRate = 5,000,000 S/s	14.9 MHz, 0.2 Vpp (measurement is intentionally aliased)	$ x < 3\text{ dB}$

Verifying Input Impedance

To verify the input impedance specifications of your digitizer, complete the following steps for each channel:

1. Connect the digitizer input to the ohmmeter with a coaxial cable.
2. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = The range values for the Input Impedance entry Table 6; you must test all four ranges
 - **offset** = 0.0
 - **coupling** = `NISCOPE_VAL_DC`
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`

Wait 10 ms for the input stage to settle.

3. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance** = `NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth** = `NISCOPE_VAL_FULL_BANDWIDTH` (or 0.0 in LabVIEW)
4. Call `niScope_ReadWaveform` to ensure the hardware is programmed.
5. Measure the impedance (x) on the ohmmeter and compare it to the success condition in Table 6. If x is outside the given range, your NI 5102 has failed this test. Return your digitizer to NI for repair.

You have now finished verifying the input impedance.

Verifying AC Coupling Cutoff Frequency

To verify the AC coupling cutoff frequency of your digitizer, complete the following steps for each AC coupling entry in Table 6 and each channel:

1. Connect BNC from the signal generator to the high-impedance digitizer input.
2. Set signal generator to the frequency and amplitude listed in Table 6.
3. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = 2.0
 - **offset** = 0.0
 - **coupling** = `NISCOPE_VAL_DC`
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`
4. Wait 10 ms for the input stage to settle.

5. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance** = `NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth** = `NISCOPE_VAL_FULL_BANDWIDTH` (or 0.0 in LabVIEW)
6. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate** = 10,000 S/s
 - **minRecordLength** = 10,000
 - **referencePosition** = 50.0
7. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AC_ESTIMATE`.
Record this value. It is the AC estimate with DC coupling used in step 10.
8. Call `niScope_ConfigureVertical` with the following parameters:
 - **range** = 2.0
 - **offset** = 0.0
 - **coupling** = `NISCOPE_VAL_AC`
 - **probeAttenuation** = 1.0
 - **enabled** = `NISCOPE_VAL_TRUE`

Wait 300 ms for the input stage to settle.
9. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AC_ESTIMATE`.
Record this value. It is the AC estimate with AC coupling used in step 10.
10. Compute the response in decibels using the formula:

$$response = 20\log_{10}\left[\frac{\text{AC estimate with AC coupling}}{\text{AC estimate with DC coupling}}\right]$$

11. Compare *response* to the success condition in Table 6. If *response* is outside the listed range, this test has failed. Return your digitizer to NI for repair.

You have now finished verifying the AC coupling cutoff frequency.

Verifying Timing

Complete the following steps to verify the timing entry in Table 6:

1. Connect the BNC cable from the signal generator to the high-impedance digitizer input.
2. Set the signal generator to 10 kHz, 1.8 Vpp sine wave.
3. Call `niScope_ConfigureVertical` with the following parameters:
 - **range**—2.0
 - **offset**—0.0
 - **coupling**—`NISCOPE_VAL_DC`
 - **probeAttenuation**—1.0
 - **enabled**—`NISCOPE_VAL_TRUE`

Wait 10 ms for the input stage to settle.

4. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance**—`NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth**—`NISCOPE_VAL_FULL_BANDWIDTH` (or 0.0 in LabVIEW)
5. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate**—1,000,000
 - **minRecordLength**—100,000
6. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AVERAGE_FREQUENCY`.
7. The returned frequency value must be between 9999 and 10,001 Hz, or a hardware error exists. If this step fails, terminate the verification procedure and return your digitizer to NI for repair.
8. Set the signal generator to a 1.8 Vpp, 10 MHz sine wave. This wave is intentionally undersampled, where the sampling rate is an even multiple of the sine wave frequency.
9. Call `niScope_ReadWaveformMeasurement` with **measFunction** set to `NISCOPE_VAL_AVERAGE_PERIOD`.

Record the period measurement to use in step 11.

10. If the returned status is `NISCOPE_ERROR_UNABLE_TO_PERFORM_MEASUREMENT`, call `niScope_errorHandler` with **errorCode** set to the returned error value. If the timing is perfectly aliased, the waveform is a DC level and the period measurement fails. Therefore, if the error description indicates the measurement failed due to not enough crosspoints, the device passed this portion of the test.
11. If the return status is anything other than `NISCOPE_ERROR_UNABLE_TO_PERFORM_MEASUREMENT`, compute the actual sample rate (x), assuming a perfect source, with the following formula:

$$x = \frac{\text{specified sample rate} \times \text{source frequency} \times \text{period}}{\text{source frequency} \times \text{period} - 1}$$

which is:

$$x = \frac{10^{13} \times \text{period}}{10^7 \times \text{period} - 1}$$

12. Compare the actual sample rate (x) to the success condition in Table 6. If x is outside the range of the success condition, return your digitizer to NI for repair.

You have finished verifying the timing.

Verifying Trigger Sensitivity

Complete the following steps for all analog trigger channels to test the smallest signal on which the digitizer should trigger by trying all possible trigger levels:

1. Connect the BNC cable from the signal generator to the high-impedance digitizer input.
2. Apply a 1 MHz sine wave with zero vertical offset, and peak-to-peak voltage as listed in Table 6.
3. Call `niScope_ConfigureVertical` with the following parameters:
 - **range**—10.0
 - **offset**—0.0
 - **coupling**—`NISCOPE_VAL_DC`
 - **probeAttenuation**—1.0
 - **enabled**—`NISCOPE_VAL_TRUE`

4. Wait 10 ms for the input stage to settle.
5. Call `niScope_ConfigureChanCharacteristics` with the following parameters:
 - **inputImpedance**—`NISCOPE_VAL_1_MEG_OHM`
 - **bandwidth**—`NISCOPE_VAL_FULL_BANDWIDTH` (or `0.0` in LabVIEW)
6. Call `niScope_ConfigureHorizontalRate` with the following parameters:
 - **minSampleRate**—`20,000,000`
 - **minRecordLength**—`128`
7. Call `niScope_ConfigureTriggerSource` with the following parameters:
 - **triggerSource**—The channel you are verifying, such as `0` or `NISCOPE_VAL_EXTERNAL`
 - **triggerType**—`NISCOPE_VAL_EDGE`
 - **triggerDelay**—`0`
 - **holdoff**—`0`
8. Call `niScope_ConfigureEdgeTrigger` with the following parameters:
 - **level**—The trigger level as discussed in step 10
 - **triggerCoupling**—`NISCOPE_VAL_AC`
 - **slope**—`NISCOPE_VAL_POSITIVE`
9. Call `niScope_ReadWaveform` to read a waveform with **maxTime** set to 100 ms or greater and **waveform size** set to 128. If this function returns a “maximum time exceeded” error, the digitizer did not trigger. If the `niScope_ReadWaveform` succeeds, the digitizer passed this test.
10. Call `niScope_Abort` to stop the test.
11. Repeat steps 7 through 10 starting with the low-trigger level listed in Table 6. Each time an error is returned, increment the **level** by the trigger level delta value in Table 6 until **level** is greater than the high trigger level listed in the table, or until the digitizer triggers. If all trigger levels have failed, a hardware problem exists with the trigger sensitivity. Return your digitizer to NI for repair.

You have now finished verifying the trigger sensitivity.

Table 6. NI 5102 AC Coupling Input Impedance, Timing, and Trigger Sensitivity Specifications

Name	Digitizer Parameters	Stimulus Parameters	Success Condition
AC Coupling	—	12.1 Hz, 1.8 Vpp	$ x < 3$ dB
AC Coupling	—	9.9 Hz, 1.8 Vpp	$ x > 3$ dB
Input Impedance	range = 10, 2, 0.5, and 0.1 Impedance = NISCOPE_VAL_1_MEG_OHM	—	$990,000 < x < 1,010,000 \Omega$
Timing	—	—	$999,950 < x < 1,000,050$ Hz
Trigger Sensitivity	low trigger level = -0.5 high trigger level = 0.5 trigger level delta = 0.01	300 mVpp	digitizer triggers with any valid trigger level

Adjusting Your NI 5102

If the verification procedure determines that your digitizer is out of specification, you need to adjust the calibration constants.

The NI 5102 has no internal reference voltage, so external connections are required for vertical range and offset adjustments. The external adjustment is completely independent for each channel and vertical range setting combination. You can perform the verification and adjustment for only the vertical ranges in question, or for all the ranges.

This procedure adjusts the calibration constants stored in the external calibration area of the EEPROM and is equivalent to a factory calibration. External calibration can provide NIST-traceability if the source is traceable.

This procedure is password-protected so that users cannot accidentally modify calibration constants. You can change the password by calling the function `niScope_CalChangePassword`.

To adjust the calibration constants of your NI 5102, complete the following steps:

1. Call `niScope_CalStart` to receive a calibration-specific device handle. You must provide the external calibration password, which defaults to 0 or the empty string, "". Set the following parameters:
 - **resourceName**—A string assigned by MAX, such as `DAQ::1`
 - **password**—Default password is 0 or ""
 - **newInstrumentHandle**—The returned session handle
2. Short-circuit the input channel and call `niScope_CalAdjustOffset` to perform a preliminary offset calibration for the 0.1 V range on channel 0. Set the following parameters:
 - **sessionHandle**—The value returned by `niScope_CalStart`
 - **channelName**—The string name of the channel, such as 0 or 1
 - **range**—The voltage range you want to calibrate
3. Connect the oscilloscope calibrator to the input channel with a 50 Ω coaxial cable. Apply the positive stimulus as specified in the stimulus parameters column of Table 7.
4. Call `niScope_CalAdjustRange` to adjust the vertical range of a specified channel. Set the following parameters:
 - **sessionHandle**—The value returned by `niScope_CalStart`
 - **channelName**—The string name of the channel, such as 0 or 1
 - **range**—The voltage range you want to calibrate
 - **stimulus**—The positive DC voltage you apply; refer to Table 7 for the proper stimulus for each vertical range entry
5. Apply a negative stimulus as specified in the stimulus parameters column of Table 7.
6. Call `niScope_CalAdjustRange` to adjust the vertical range of a specified channel. Set the following parameters:
 - **sessionHandle**—The value returned by `niScope_CalStart`
 - **channelName**—The string name of the channel, such as 0 or 1
 - **range**—The voltage range you want to calibrate
 - **stimulus**—The negative DC voltage you apply; refer to Table 7 for the proper stimulus for each vertical range entry

7. Short-circuit the input channel and call `niScope_CalAdjustOffset` to perform a precise offset calibration using the calibrated vertical range from the previous step. Set the following parameters:
 - **sessionHandle**—The value returned by `niScope_CalStart`
 - **channelName**—The string name of the channel, such as 0 or 1
 - **range**—The voltage range you want to calibrate; refer to Table 7 to obtain the value for each vertical range
8. Repeat steps 2 through 7 for each channel and vertical range you want to adjust.
9. Call `niScope_CalSelfCalibrate` to self-calibrate. Set the following parameters:
 - **sessionHandle**—The session handle returned by `niScope_CalStart` or `niScope_init`
 - **channelName**—Ignore this parameter; use `VI_Null`
 - **option**—Use `VI_NULL` for a normal self-calibration operation; only `NISCOPE_VAL_CAL_RESTORE_EXTERNAL_CALIBRATION` is supported
10. Call `niScope_CalEnd` to release the device handle. Set the following parameters:
 - **sessionHandle**—The session handle returned by `niScope_CalStart`
 - **action**—`NISCOPE_VAL_CAL_ACTION_STORE` (as defined in `niScopeCal.h`) to store the calibration constants in the EEPROM; this option stores the external calibration date and the incremented external calibration count in the EEPROM

Table 7. NI 5102 Calibration Stimuli

Calibration Step	Digitizer Parameters	Stimulus Parameters
Vertical Range	range = 10.0 V	±4.25 VDC
Vertical Range	range = 2.0 V	±0.85 VDC
Vertical Range	range = 0.5 V	±0.21 VDC
Vertical Range	range = 0.1 V	±0.04 VDC

The vertical range and offset constants of your digitizer are now adjusted with respect to the external source.

Verifying the New Calibration Constants

NI recommends that you verify your NI 5102 operating specifications again by repeating the verification procedures in the [Verifying the Performance of Your NI 5102](#) section of this document. If your digitizer fails any test immediately after you perform an external calibration, return your digitizer to NI for repair.

Calibration Function Reference

This section lists functions specific to NI-SCOPE calibration. See the [Software](#) section of this document for instructions on installing these functions.

niScope_CalStart

Format

```
ViStatus _VI_FUNC niScope_CalStart  
(  
    ViRsrc resourceName,  
    ViConstString password,  
    ViSession *newSessionHandle  
);
```

Purpose

niScope_CalStart opens an external calibration session.

Using This Function

password is compared to the password stored in the EEPROM for additional security. By default, the password is set to NULL or an empty string. The password is stored in the EEPROM as an array of four bytes. Non-printable characters are allowed, but the array is padded with NULLs after the first NULL is found. This padding allows strings of less than four characters to be legal passwords.

All calibration functions require a session handle, such as **newSessionHandle**, that is returned by this function. Only the external calibration functions require a calibration session handle to allow password protection. All other functions, such as verification and read functions, work with both a calibration session and a session handle obtained from niScope_init. Acceptable session handles are documented for each function in this section.

Only one session handle can be obtained at a time, and every session must be closed by calling niScope_CalEnd. If you fail to close the session, you must unload the niScope_32.dll by closing your application or application development environment (ADE) before you can open another session.

If an error occurs during calibration, call niScope_errorHandler to get the error message text and niScope_CalEnd with **action** set to NISCOPE_VAL_CAL_ACTION_ABORT to close the session.

Parameters

Name	Description
resourceName	assigned by Measurement & Automation Explorer (MAX); this is a string such as "DAQ: :1"
password	compared to password in EEPROM
newInstrumentHandle	returned session handle

niScope_CalEnd

Format

```
ViStatus _VI_FUNC niScope_CalEnd  
(  
    ViSession sessionHandle,  
    ViInt32 action  
);
```

Purpose

niScope_CalEnd closes an external calibration session.

Using This Function

If **action** is NISCOPE_VAL_CAL_ACTION_ABORT, the session is closed, and the calibration constants are lost. The abort close is necessary when an error occurs during calibration. Some devices write to the EEPROM during calibration, in which case the **abort** parameter restores the EEPROM to its original state. It is, therefore, very important to call niScope_CalEnd each time niScope_CalStart is called, even if an error occurs during calibration.

If **action** is NISCOPE_VAL_CAL_ACTION_STORE, the calibration constants are stored in the EEPROM. If you call niScope_CalStoreMiscInfo during the calibration session, the miscellaneous information is stored. Otherwise, the miscellaneous information is set to zero or the empty string in the EEPROM. The current system date and an incremented external calibration count are automatically stored in the EEPROM.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart
action	values defined in niScopeCal.h: NISCOPE_VAL_CAL_ACTION_STORE, NISCOPE_VAL_CAL_ACTION_ABORT

niScope_CalChangePassword

Format

```
ViStatus _VI_FUNC niScope_CalChangePassword  
(  
    ViSession sessionHandle,  
    ViConstString oldPassword,  
    ViConstString newPassword  
);
```

Purpose

niScope_CalChangePassword checks your old password with the one stored in the EEPROM. If they match, the new password is stored in the EEPROM.

Using This Function

The password can be four characters long, but shorter strings are acceptable. Non-printable values are acceptable, but zero is treated as an end-of-string character. If a zero (or end-of-string marker) is detected, zeros are added to the end to make the string four characters long.

By default, the password in the EEPROM is an array of nulls, or the empty string.

Do not forget your password. If you do, call NI.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart or niScope_init
oldPassword	value currently stored in EEPROM (factory default is "")
newPassword	new value to store in EEPROM

niScope_CalFetchCount

Format

```
ViStatus _VI_FUNC niScope_CalFetchCount  
(  
    ViSession sessionHandle  
    ViInt32 whichOne,  
    ViInt32 *calibrationCount  
);
```

Purpose

niScope_CalFetchCount returns the calibration count, which is the number of times the digitizer has been calibrated.

Using This Function

whichOne determines whether it is the self or external calibration count. Possible values are defined in niScopeCal.h.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart or niScope_init
whichOne	values defined in niScopeCal.h: NISCOPE_VAL_CAL_SELF, NISCOPE_VAL_CAL_EXTERNAL
calibrationCount	returned number of times device has been calibrated

niScope_CalFetchDate

Format

```
ViStatus _VI_FUNC niScope_CalFetchDate  
(  
    ViSession sessionHandle  
    ViInt32 whichOne,  
    ViInt32 *year,  
    ViInt32 *month,  
    ViInt32 *day  
);
```

Purpose

niScope_CalFetchDate returns the self-calibration, external calibration, or manufacture date.

Using This Function

If you are upgrading from NI-Scope version 1.5 or earlier, the initial calibration dates will be incorrect because the older versions of NI-SCOPE calibration do not support the date feature.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart or niScope_init
whichOne	values defined in niScopeCal.h: NISCOPE_VAL_CAL_SELF, NISCOPE_VAL_CAL_EXTERNAL, or NISCOPE_VAL_CAL_MANUFACTURE
year	returned year of last calibration (for example, 2000)
month	returned month of last calibration (1–12)
day	returned day of last calibration (1–31)

niScope_CalFetchMiscInfo

Format

```
ViStatus _VI_FUNC niScope_CalFetchMiscInfo  
(  
    ViSession sessionHandle,  
    ViChar *info  
);
```

Purpose

niScope_CalFetchMiscInfo returns the miscellaneous information you may store during an external calibration using niScope_StoreMiscInfo.

Using This Function

info must be a character array of length five or greater. The fifth byte is always set to NULL to terminate the string.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart or niScope_init
info	array of 5 bytes (4 bytes of information plus 1 NULL byte) stored in EEPROM during last external calibration

niScope_CalStoreMiscInfo

Format

```
ViStatus _VI_FUNC niScope_CalStoreMiscInfo
(
    ViSession sessionHandle,
    ViConstString info
);
```

Purpose

niScope_CalStoreMiscInfo allows you to store miscellaneous information in the EEPROM during external calibration. For example, you can store an operator ID for the person or company performing the calibration.

Using This Function

If this function is not called during an external calibration, the miscellaneous information is set to zero in the EEPROM. This setting ensures a consistent calibration date, count, and miscellaneous information values in the EEPROM.

Four bytes are stored in the EEPROM, and non-printable characters are valid. However, NULL is treated as an end of string marker, and all bytes following the first NULL are set to NULL.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart
info	array of 4 bytes of info stored in EEPROM during last external calibration

niScope_CalSelfCalibrate

Format

```
ViStatus _VI_FUNC niScope_CalSelfCalibrate  
(  
    ViSession sessionHandle,  
    ViConstString channelName,  
    ViInt32 option  
);
```

Purpose

niScope_CalSelfCalibrate performs a self-calibration.

Using This Function

If the self-calibration is successful, the calibration constants are immediately stored in the self-calibration area of the EEPROM, along with the self-calibration date and incremented count. The outdated niScope_Calibrate with **calibrationOperation** set to NISCOPE_VAL_SELF_CALIBRATION calls this function. The only valid value for **option** is NISCOPE_VAL_CAL_RESTORE_EXTERNAL_CALIBRATION. This is equivalent to calling the outdated niScope_Calibrate with **calibrationOperation** set to NISCOPE_VAL_RESTORE_FACTORY_CALIBRATION. You should only use this function if the self-calibration routine fails and you must use the digitizer rather than return it for repair. This function restores the last external calibration constants. Notice that using the external calibration constants does not correct for environmental conditions, so the digitizer will not be in an optimal calibration state.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart or niScope_init
channelName	this parameter is ignored; use VI_NULL
option	only NISCOPE_VAL_CAL_RESTORE_EXTERNAL_CALIBRATION is supported; use VI_NULL for a normal self-calibration operation

niScope_CalAdjustRange

Format

```
ViStatus _VI_FUNC niScope_CalAdjustRange  
(  
    ViSession sessionHandle,  
    ViConstString channelName,  
    ViReal64 range,  
    ViReal64 stimulus  
);
```

Purpose

niScope_CalAdjustRange performs an external calibration of the vertical range for a specified channel and vertical range setting of the NI 5102. You should call niScope_CalAdjustOffset prior to this function, with the same **channelName** and **range** parameters, to perform a preliminary offset calibration using a default vertical range. See the [External Calibration Procedures](#) section earlier in this document for details on writing a calibration procedure for the NI 5102.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart
channelName	string name of channel; for example, “0” or “1”
range	the voltage range to calibrate; see the External Calibration Procedures section in this document for the legal values
stimulus	the DC voltage of the applied signal; see the External Calibration Procedures section in this document for the proper stimulus to apply

niScope_CalAdjustOffset

Format

```
ViStatus _VI_FUNC niScope_CalAdjustOffset  
(  
    ViSession sessionHandle,  
    ViConstString channelName,  
    ViReal64 range  
);
```

Purpose

niScope_CalAdjustOffset externally calibrates a specified channel and vertical range of the NI 5102. Call this function before niScope_CalAdjustRange to perform a preliminary offset calibration. The first time niScope_CalAdjustOffset is called, a default vertical range parameter is used in NI-SCOPE to calibrate the vertical offset. When the function is called again after niScope_CalAdjustRange, it performs a precise offset calibration using the calibrated vertical range. See the [External Calibration Procedures](#) section earlier in this document for details on writing a calibration procedure for the NI 5102.

Parameters

Name	Description
sessionHandle	session handle returned by niScope_CalStart
channelName	string name of channel; for example, 0 or 1
range	the voltage range to calibrate; see External Calibration Procedures section in this document for legal values