CALIBRATION PROCEDURE

NI 5132/5133

This document contains instructions for calibrating National Instruments USB-5132/5133 digitizers. This calibration procedure is intended for metrology labs. This document describes specific programming steps for writing an external calibration procedure for the NI 5132/5133.

Contents

Conventions

The following conventions are used in this manual:

» — 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.

This icon denotes a caution, which advises you of precautions to take to avoid injury, data loss, or a system crash. When this symbol is marked on a product, refer to the *Read Me First: Safety and Electromagnetic Interference* document for information about precautions to take.

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.

italic — Italic text denotes variables, emphasis, a cross-reference, or an introduction to a key concept. This font 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.

`monospace italic` — Italic text in this font denotes text that is a placeholder for a word or value that you must supply.

Platform — Text in this font denotes a specific platform and indicates that the text following it applies only to that platform.

Software Requirements

Calibrating the NI 5132/5133 requires installing NI-SCOPE version 3.4 or later on the calibration system.You can download NI-SCOPE from the Instrument Driver Network at `ni.com/idnet`. NI-SCOPE supports programming the *Self-Calibration* and *Verification* sections in a number of programming languages. However, only LabVIEW and C are supported for the *Adjustment* section.

NI-SCOPE includes all the functions and attributes necessary for calibrating the NI 5132/5133. LabVIEW support is installed in `niScopeCal.llb`, and all calibration functions appear in the function palette. For LabWindows™/CVI™, the NI-SCOPE function panel `niScopeCal.fp` provides further help on the functions available in CVI. Refer to Table 1 for installed file locations.

Calibration functions are LabVIEW VIs or C function calls in the NI-SCOPE driver. The C function calls are valid for any compiler capable of calling a 32-bit DLL. Many of the functions use constants defined in the `niScopeCal.h` file. To use these constants in C, you must include `niScopeCal.h` in your code when you write the calibration procedure.

For more information on the calibration VIs and functions, refer to the *NI-SCOPE LabVIEW Reference Help* or the *NI-SCOPE Function Reference Help*. These references can be found in the *NI High-Speed Digitizers Help*. Refer to the *NI-SCOPE Readme* for the installed locations of these documents.

Table 1. Calibration File Locations after Installing NI-SCOPE

File Name and Location	Description
`IVI\Bin\niscope_32.dll`	NI-SCOPE driver containing the entire NI-SCOPE API, including calibration functions
`IVI\Lib\msc\niscope.lib`	NI-SCOPE library for Microsoft C containing the entire NI-SCOPE API, including calibration functions
`LabVIEW (`*`version`*`)\examples\instr\niScope`	Directory of LabVIEW NI-SCOPE example VIs, including self-calibration
`LabVIEW (`*`version`*`)\instr.lib\niScope Calibrate\niScopeCal.llb`	LabVIEW VI library containing VIs for calling the NI-SCOPE calibration API; access calibration functions from the NI-SCOPE calibration section of the LabVIEW function palette

Table 1. Calibration File Locations after Installing NI-SCOPE (Continued)

File Name and Location	Description
`IVI\Drivers\niScope\niScopeCal.fp`	CVI function panel file that includes external calibration function prototypes and help on using NI-SCOPE in the CVI environment
`IVI\Include\niScopeCal.h`	Calibration header file, which you must include in any C program accessing calibration functions. This file automatically includes `niScope.h`, which defines the rest of the NI-SCOPE interface
`IVI\Drivers\niScope\Examples\`	Directory of NI-SCOPE examples for CVI, C, Visual C++, and Visual Basic

Documentation Requirements

You may find the following documentation helpful as you write your calibration procedure:

- *NI High-Speed Digitizers Getting Started Guide*
- *NI High-Speed Digitizers Help*
- *NI USB-5132/5133 Specifications*
- *NI-SCOPE Function Reference Help* or *NI-SCOPE LabVIEW Reference Help*

These documents are installed with NI-SCOPE. You can also download the latest versions from the NI Web site at `ni.com/manuals`.

Password

The password required to open an external calibration session. If the password has not been changed since manufacturing, the password is “`NI`”.

Calibration Interval

External Calibration

The measurement accuracy requirements of your application determine how often you should externally calibrate the NI 5132/5133. NI recommends that you perform a complete external calibration at least once every two years. You can shorten this interval based on the accuracy demands of your application. Refer to the *Appendix A: Calibration Options* section for more information.

Self-Calibration

Self-calibration can be performed as necessary to compensate for environmental changes.

Caution Although you can use self-calibration repeatedly, self-calibrating the NI 5132/5133 more than a few times a day may cause excessive wear on the relays over time.

Test Equipment

Table 2 lists the equipment required for externally calibrating the NI 5132/5133. If you do not have the recommended instruments, use these specifications to select a substitute calibration standard.

Table 2. Required Equipment Specifications for NI 5132/5133 External Calibration

Required Equipment	Recommended Equipment	Parameter Measured	Specification
Signal Generator	Fluke 9500B oscilloscope calibrator or Wavetek 9500 with high-stability reference option	DC Accuracy, Adjustment	Output Range: 20 V to –20 V into 1 MΩ DC Voltage Accuracy: DC ±0.3% of output into 1 MΩ
		Bandwidth	Output Range: 0.032 Vpp to 5.5 Vpp into 50 Ω AC Voltage Accuracy: ±2% output amplitude flatness for leveled sine wave up to 50.1 MHz relative to 50 kHz into 50 Ω
		Adjustment	1 kHz square wave Output Range: 1.65 Vpp to 8.5 Vpp into 1 MΩ Rise Time: 20 ns to 150 ns
		Timing Accuracy	11 MHz sine wave Output Range: 1 Vpp into 50 Ω Frequency Accuracy: ±4 ppm
Digital Multimeter (DMM)	NI PXI-4071	AC Amplitude Accuracy	Input Range: 0.03 Vpp to 25.0 Vpp AC Voltage Accuracy: 0.3% of input at 50 kHz
BNC Tee	—	AC Amplitude Accuracy	50 Ω
Signal Generator	NI PXI-5421	AC Amplitude Accuracy	50 kHz sine wave Amplitude Range: 0.036 Vpp to 24 Vpp into 1 MΩ

Note The delay times indicated in this procedure apply specifically to the Fluke 9500B calibrator. If you use a different calibrator, you may need to adjust these delay times.

Test Conditions

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

- Always connect the calibrator test head directly to the input BNC of the digitizer, or use a short 50 Ω BNC coaxial cable if necessary. Long cables and wires act as antennae, picking up extra noise that can affect measurements.
- Keep relative humidity between 10 and 90% non-condensing, or consult the digitizer hardware documentation for the optimum relative humidity.
- Maintain an ambient temperature of 23 ±5 °C.
- Allow a warm-up time of at least 10 minutes after the NI-SCOPE driver is loaded. Unless manually disabled, the NI-SCOPE driver automatically loads with the operating system and enables the device. The warm-up time ensures that the measurement circuitry of the NI 5132/5133 is at a stable operating temperature.
- Install a heavy ground wire between the NI 5132/5133 ground lug and the signal source ground. Refer to the *Ground Loop Noise* topic in the *NI High-Speed Digitizers Help* for more information.

Calibration Procedures

The calibration process includes the following steps:

1. *Initial Setup*—Install the device and configure it in Measurement & Automation Explorer (MAX).

Note Allow a 10-minute warm-up time before beginning self-calibration.

2. *Self-Calibration*—Adjust the self-calibration constants of the device.
3. *Verification*—Verify the existing operation of the device. This step confirms whether the device is operating within its specified range prior to calibration.
4. *Adjustment*—Perform an external adjustment of the device that adjusts the calibration constants with respect to a known voltage source. The adjustment procedure automatically stores the calibration date on the EEPROM to allow traceability.
5. Reverification—Repeat the verification procedure to ensure that the device is operating within its specifications after adjustment.

These steps are described in more detail in the following sections.

Note In some cases, the complete calibration procedure may not be required. Refer to *Appendix A: Calibration Options* for more information.

Initial Setup

Refer to the *NI High-Speed Digitizers Getting Started Guide* for information about how to install the software and hardware, and how to configure the device in MAX.

Self-Calibration

Self-calibration of the NI 5132/5133 corrects for DC offset errors for all ranges and filter paths.

Note Allow a 10-minute warm-up time before beginning self-calibration.

Note Self-calibrate the digitizer before you perform verification. NI-SCOPE includes self-calibration example programs for LabVIEW, CVI, and Microsoft Visual C.

You can initiate self-calibration using the following methods:

- MAX
- NI-SCOPE Soft Front Panel (SFP)
- NI-SCOPE

MAX

To initiate self-calibration from MAX, complete the following steps:

1. Disconnect or disable any AC inputs to the digitizer.
2. Launch MAX.
3. Select **My System»Devices and Interfaces»NI-DAQmx Devices**.
4. Select the device that you want to calibrate.
5. Initiate self-calibration using one of the following methods:
   - Click **Self-Calibrate** in the upper right corner of MAX.
   - Right-click the name of the device in the MAX configuration tree and select **Self-Calibrate** from the drop-down menu.

NI-SCOPE SFP

To initiate self-calibration from the NI-SCOPE SFP, complete the following steps:

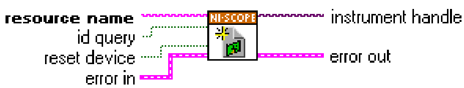
1. Disconnect or disable any AC inputs to the digitizer.
2. Launch the NI-SCOPE SFP, which is available at **Start» All Programs»National Instruments»NI-SCOPE» SCOPE Soft Front Panel**.
3. Select the device you want to calibrate using the Device Configuration dialog box by selecting **Edit»Device Configuration**.
4. Launch the Calibration dialog box by selecting **Utility»Self Calibration**.
5. Click **OK** to begin self-calibration.

NI-SCOPE

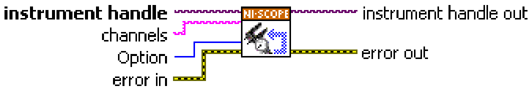
To self-calibrate the NI 5132/5133 programmatically using NI-SCOPE, complete the following steps:

1. Disconnect or disable any AC inputs to the digitizer.
2. Open a session and obtain a session handle using the niScope Initialize VI.

Note Throughout the procedure, refer to the C/C++ function call parameters for the LabVIEW input values.

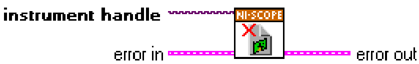
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE` **vi**: The returned session handle that you use to identify the instrument in all subsequent NI-SCOPE driver function calls

3. Use the niScope Cal Self Calibrate VI to self-calibrate the digitizer.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalSelfCalibrate` with the following parameters: **sessionHandle**: The instrument handle from `niScope_init` **channelList**: `VI_NULL` **option**: `VI_NULL`

Note Because the session is a standard session rather than an external calibration session, the new calibration constants are immediately stored in the EEPROM. Therefore, you can include this procedure in any application that uses the digitizer.

4. End the session using the niScope Close VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

Verification

Note After the 10-minute warm-up time, always self-calibrate the digitizer before beginning a verification procedure.

This section describes the program you must write to verify either the calibration test limits or the published specifications for the digitizer. Refer to *Appendix A: Calibration Options* to determine which limits to use in these procedures.

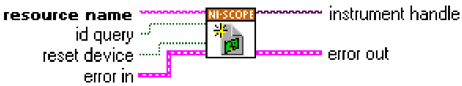
Note If any of these tests fail immediately after you perform an external adjustment, make sure that you have met the requirements listed in the *Test Equipment* and *Test Conditions* sections before you return the digitizer to National Instruments for repair.

Vertical Offset Accuracy

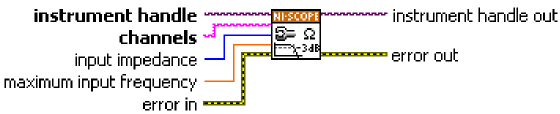
Complete the following steps to verify vertical offset accuracy of the NI 5132/5133. You must verify both channels with each iteration in Table 3.

1. Open a session and obtain a session handle using the niScope Initialize VI.

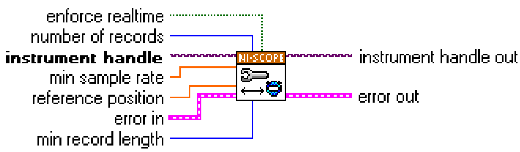
Note Throughout the procedure, refer to the C/C++ function call parameters for the LabVIEW input values.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE` **vi**: The returned session handle that you use to identify the instrument in all subsequent NI-SCOPE driver function calls

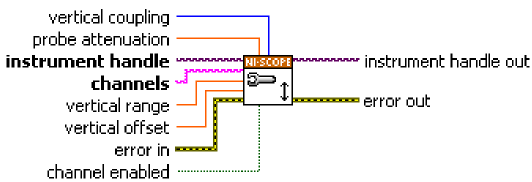
2. Configure the input impedance and the maximum input frequency using the niScope Configure Chan Characteristics VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureChan Characteristics` with the following parameters: **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **inputImpedance**: `NISCOPE_VAL_1_MEG_OHM` **maxInputFrequency**: `50,000,000`

3. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Configure HorizontalTiming` with the following parameters: **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **vi**: The instrument handle from `niScope_init` **minSampleRate**: `10,000,000` **refPosition**: `50.0` **minNumPts**: `100,000`

4. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureVertical` with the following parameters: **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **range**: The *Range* value from Table 3 for the current iteration **offset**: `0.0` **enabled**: `NISCOPE_VAL_TRUE`

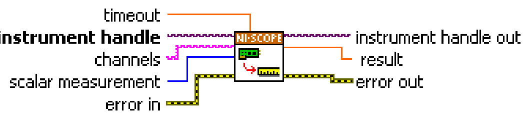
5. Commit all the parameter settings to hardware using the niScope Commit VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

6. Short-circuit the channel 0 input of the digitizer by connecting the calibrator test head directly to the digitizer and grounding the output of the calibrator.
7. Wait 2,500 ms for the calibrator to ground its output.
8. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

9. Fetch a waveform from the digitizer and perform a voltage average measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI.

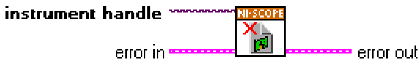
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **timeout**: `1.0` **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_AVERAGE`

Compare the resulting average voltage to the value listed in the *Published Specifications* column in Table 3 that corresponds to the vertical range used.

If the result is within the selected test limit, the device has passed this portion of the verification.

10. Repeat steps 4 through 9 for each iteration listed in Table 3.
11. Move the calibrator test head to the digitizer input channel 1 and repeat steps 4 through 10 for each iteration listed in Table 3, replacing `"0"` with `"1"` for the **channelList** parameter.

12. End the session using the niScope Close VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

You have finished verifying the vertical offset accuracy of the NI 5132/5133.

Table 3. NI 5132/5133 Vertical Offset Accuracy

Iteration	Range (Vpp)	Published Specifications (V)
1	0.04	±0.0007
2	0.1	±0.0013
3	0.2	±0.0023
4	0.4	±0.0043
5	1	±0.0103
6	2	±0.0203
7	4	±0.0403
8	10	±0.1003
9	20	±0.2003
10	40	±0.4003

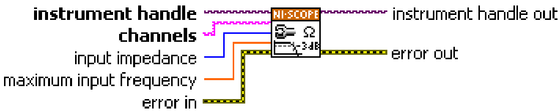
Vertical Gain Accuracy

Complete the following steps to verify the vertical gain accuracy of the digitizer. You must verify both channels with each iteration in Table 4.

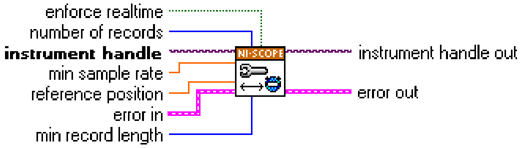
1. Open a session and obtain a session handle using the niScope Initialize VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE` **vi**: The returned session handle that you use to identify the instrument in all subsequent NI-SCOPE driver function calls

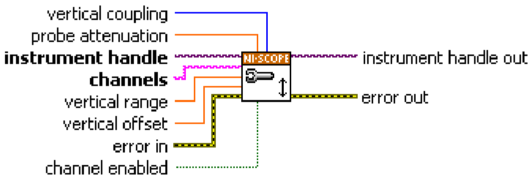
2. Configure the input impedance and the maximum input frequency using the niScope Configure Chan Characteristics VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureChan Characteristics` with the following parameters: **channelList**: `"0"` **inputImpedance**: `NISCOPE_VAL_1_MEG_OHM` **maxInputFrequency**: `50,000,000`

3. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureHorizontalTiming` with the following parameters: **vi**: The instrument handle from `niScope_init` **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **vi**: The instrument handle from `niScope_init` **minSampleRate**: `10,000,000` **refPosition**: `50.0` **minNumPts**: `100,000`

4. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureVertical` with the following parameters: **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **range**: The *Range* value from Table 4 for the current iteration **offset**: `0.0` **enabled**: `NISCOPE_VAL_TRUE`

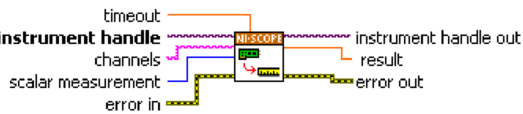
5. Commit all the parameter settings to hardware using the niScope Commit VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

6. Connect the calibrator test head directly to the channel 0 input of the digitizer and output the *Positive Input* voltage from Table 4 for the current iteration. Be sure to configure the load impedance of the calibrator to match the input impedance of the digitizer.
7. Wait 2,500 ms for the impedance matching of the calibrator to settle.
8. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

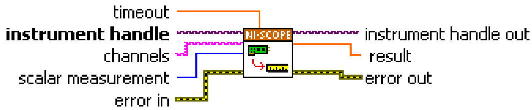
9. Fetch a waveform from the digitizer and perform a voltage average measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. This value is the *Measured Positive Voltag*e used in step 14.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **vi**: The instrument handle from `niScope_init` **timeout**: `1.0` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_AVERAGE`

10. Using the calibrator, output the *Negative Input* voltage from Table 4 for the current iteration.
11. Wait 2500 ms for the output of the calibrator to settle.
12. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

13. Fetch a waveform from the digitizer and perform a voltage average measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. This value is the *Measured Negative Voltage* used in step 14.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **timeout**: `1.0` **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_AVERAGE`

14. Calculate the error in the vertical gain as a percentage of input using the following formula:

$$error = \left(\left(\frac{a-b}{c-d}\right) - 1\right) \times 100$$

where

a = the *Measured Positive Voltage*

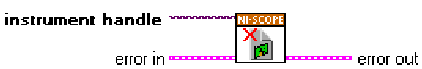
b = the *Measured Negative Voltage*

c = the applied *Positive Voltage*

d = the applied *Negative Voltage*

Compare the resulting percentage error to the *Published Specifications* listed in Table 4. If the result is within the selected test limit, the device has passed this portion of the verification.

15. Repeat steps 4 through 14 for each iteration listed in Table 4.
16. Move the calibrator test head to the digitizer input channel 1 and repeat steps 4 through 15 for each iteration listed in Table 4, replacing `"0"` with `"1"` for the **channelList** parameter.
17. End the session using the niScope Close VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

You have finished verifying the vertical gain accuracy of the NI 5132/5133.

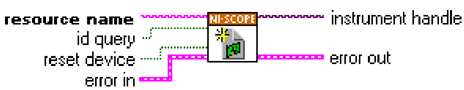
Table 4. NI 5132/5133 Vertical Gain Accuracy

Iteration	Range (Vpp)	Positive Input (V)	Negative Input (V)	Published Specifications (V)
1	0.04	0.018	–0.018	±2.0%
2	0.1	0.045	–0.045	±2.0%
3	0.2	0.09	–0.09	±2.0%
4	0.4	0.18	–0.18	±2.0%
5	1	0.45	–0.45	±2.0%
6	2	0.9	–0.9	±2.0%
7	4	1.8	–1.8	±2.0%
8	10	4.5	–4.5	±2.0%
9	20	9	–9	±2.0%
10	40	18	–18	±2.0%

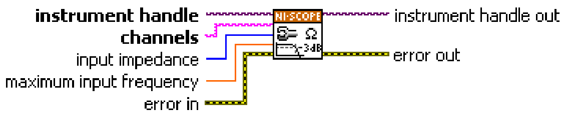
Programmable Vertical Offset Accuracy

Complete the following steps to verify the programmable vertical offset accuracy for each digitizer channel.

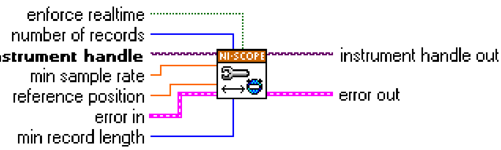
1. Open a session and obtain a session handle using the niScope Initialize VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE` **vi**: The returned session handle that you use to identify the instrument in all subsequent NI-SCOPE driver function calls

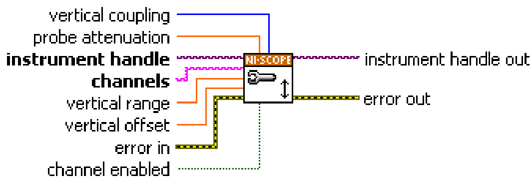
2. Configure the input impedance and the maximum input frequency using the niScope Configure Chan Characteristics VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureChan Characteristics` with the following parameters: **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **inputImpedance**: `NISCOPE_VAL_1_MEG_OHM` **maxInputFrequency**: `50,000,000`

3. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Configure HorizontalTiming` with the following parameters: **vi**: The instrument handle from `niScope_init` **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **minSampleRate**: `10,000,000` **refPosition**: `50.0` **minNumPts**: `100,000`

4. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureVertical` with the following parameters: **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **range**: The *Range* value from Table 5 for the current iteration **offset**: The *Positive Input* value from Table 5 for the current iteration **enabled**: `NISCOPE_VAL_TRUE`

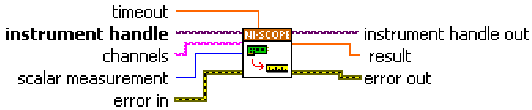
5. Commit all the parameter settings to hardware using the niScope Commit VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

6. Connect the calibrator test head directly to the channel 0 input of the digitizer and output the *Positive Input* voltage value from Table 5 for the current iteration with a 1 MΩ load impedance.
7. Wait 2,500 ms for the impedance matching of the calibrator to settle.
8. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

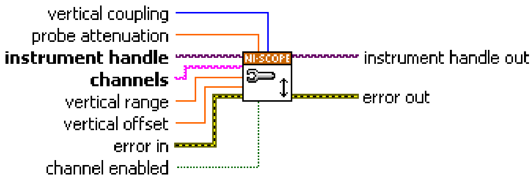
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

9. Fetch a waveform from the digitizer and perform a voltage average measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. This value is the *Measured Positive Voltage* used in step 16.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **vi**: The instrument handle from `niScope_init` **timeout**: `1.0` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_AVERAGE`

10. Output the *Negative Input* voltage value from Table 5 for the current iteration with the calibrator.

11. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureVertical` with the following parameters: **vi**: The instrument handle from `niScope_init` **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **channelList**: `"0"` **range**: The *Range* value from Table 5 for the current iteration **offset**: The *Negative Input* value from Table 5 for the current iteration **enabled**: `NISCOPE_VAL_TRUE`

12. Commit all the parameter settings to hardware using the niScope Commit VI.

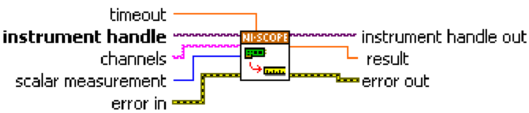
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

13. Wait 2,500 ms for the output of the calibrator to settle.

14. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

15. Fetch a waveform from the digitizer and perform a voltage average measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. This value is the *Measured Negative Voltage* used in step 16.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **vi**: The instrument handle from `niScope_init` **timeout**: `1.0` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_AVERAGE`

16. Calculate the error in the programmable vertical offset as a percentage of input using the following formula:

$$error = \left(\left(\frac{a-b}{c-d}\right) - 1\right) \times 100$$

where

a = the *Measured Positive Voltage*

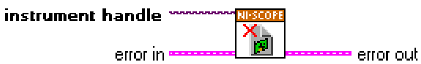
b = the *Measured Negative Voltage*

c = the applied *Positive Voltage*

d = the applied *Negative Voltage*

Compare the resulting percent to the *Calibration Test Limits* or the *Published Specifications* listed in Table 5. If the result is within the selected test limit, the device has passed this portion of the verification.

17. Repeat steps 4 through 16 for each iteration listed in Table 5.
18. Move the calibrator test head to the digitizer input channel 1 and repeat steps 4 through 17, replacing `"0"` with `"1"` for the **channelList** parameter.
19. End the session using the niScope Close VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

You have finished verifying the programmable vertical offset accuracy of the NI 5132/5133.

Table 5. NI 5132/5133 Programmable Vertical Offset

Iteration	Range (Vpp)	Positive Input (V)	Negative Input (V)	Calibration Test Limits	Published Specifications
1	0.04	0.4	–0.4	±4.6%	±5.0%
2	0.1	0.4	–0.4	±2.1%	±2.5%
3	0.2	0.4	–0.4	±2.1%	±2.5%
4	0.4	0.4	–0.4	±2.1%	±2.5%
5	1	4	–4	±2.1%	±2.5%

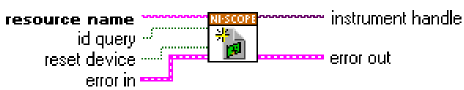
Table 5. NI 5132/5133 Programmable Vertical Offset (Continued)

Iteration	Range (Vpp)	Positive Input (V)	Negative Input (V)	Calibration Test Limits	Published Specifications
6	2	4	–4	±2.1%	±2.5%
7	4	4	–4	±2.1%	±2.5%
8	10	20	–20	±2.1%	±2.5%
9	20	15	–15	±2.1%	±2.5%
10	40	5	–5	±2.1%	±2.5%

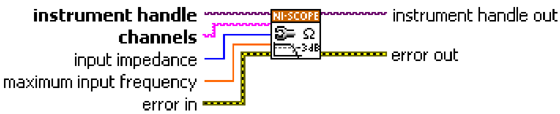
AC Amplitude Accuracy

Complete the following steps to verify the AC amplitude accuracy for the NI 5132/5133:

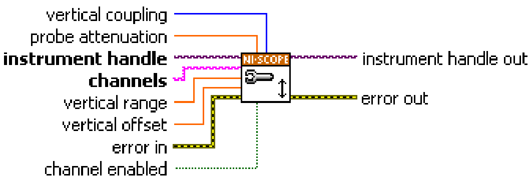
1. Make the following connections:
    a. Connect the signal generator to one port of the BNC tee.
    b. Connect the DMM to another port of the BNC tee.
    c. Connect the channel 0 input of the digitizer to the remaining port of the BNC tee.
2. Open a session and obtain a session handle using the niScope Initialize VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE` **vi**: The returned session handle that you use to identify the instrument in all subsequent NI-SCOPE driver function calls

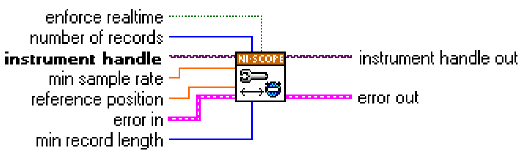
3. Configure the input impedance and the maximum input frequency using the niScope Configure Chan Characteristics VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureChan Characteristics` with the following parameters: **channelList**: `"0"` **inputImpedance**: `NISCOPE_VAL_1_MEG_OHM` **maxInputFrequency**: `50,000,000`

4. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureVertical` with the following parameters: **vi**: The instrument handle from `niScope_init` **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **channelList**: `"0"` **range**: The *Range* value from Table 6 for the current iteration. **offset**: `0.0` **enabled**: `NISCOPE_VAL_TRUE`

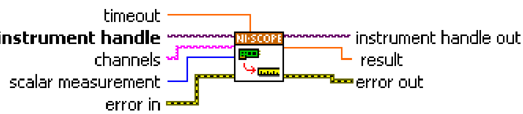
5. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Configure HorizontalTiming` with the following parameters: **vi**: The instrument handle from `niScope_init` **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **minSampleRate**: `10,000,000` **refPosition**: `50.0` **minNumPts**: `100,000`

6. Commit all the parameter settings to hardware using the niScope Commit VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

7. Configure the signal generator to output a 50 kHz, *Input Voltage* sine wave into a 1 MΩ load impedance for the current iteration in Table 6.
8. Wait 2,500 ms for the output of the signal generator to settle.
9. Fetch a waveform from the digitizer and perform a voltage RMS measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. This value is the *Measured RMS Voltage of 50 kHz Sine Wave* used in step 11.

LabVIEW VI	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **timeout**: `1.0` **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_RMS`

10. Measure the amplitude (in V_{rms}) of the sine wave using the DMM. This value is the *DMM Measured RMS Voltage of 50 kHz Sine Wave* used in step 11.
11. Calculate the amplitude difference using the following formula:

$$error = \frac{(a-b)}{b} \times 100$$

where

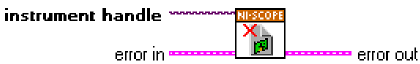
a = *Measured RMS Voltage of 50 kHz Sine Wave*

b = *DMM Measured RMS Voltage of 50 kHz Sine Wave*

If the result is within the *Published Specifications* listed in Table 6, the device has passed this portion of the verification.

12. Repeat steps 4 through 11 for each iteration listed in Table 6.

13. Make the following connections:
    a. Connect the signal generator to one port of the BNC tee.
    b. Connect the DMM to another port of the BNC tee.
    c. Connect the channel 1 input of the digitizer to the remaining port of the BNC tee.
14. Repeat steps 4 through 12 for each iteration listed in Table 6, replacing `"0"` with `"1"` for the **channelList** parameter.
15. End the session using the niScope Close VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

You have finished verifying the ac amplitude accuracy of the NI 5132/5133.

Table 6. NI 5132/5133 AC Amplitude Accuracy

Iteration	Range (Vpp)	Sine Wave Amplitude (Vpp)	Published Specifications
1	0.04	0.036	±2.0%
2	0.1	0.09	±2.0%
3	0.2	0.18	±2.0%
4	0.4	0.36	±2.0%
5	1	0.9	±2.0%
6	2	1.8	±2.0%
7	4	3.6	±2.0%
8	10	9	±2.0%
9	20	18	±2.0%
10	40	24	±2.0%

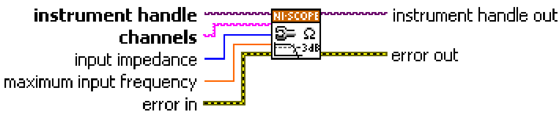
Timing Accuracy

Complete the following steps to verify the timing accuracy for the NI 5132/5133:

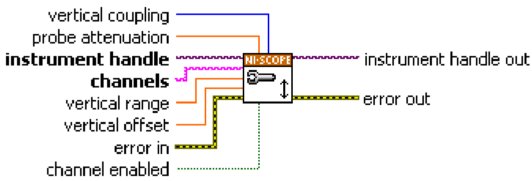
1. Open a session and obtain a session handle using the niScope Initialize VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE` **vi**: The returned session handle that you use to identify the instrument in all subsequent NI-SCOPE driver function calls

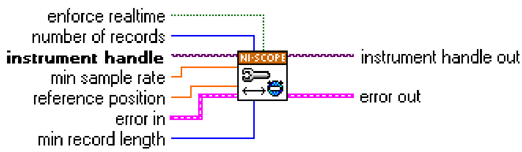
2. Configure the input impedance and the maximum input frequency using the niScope Configure Chan Characteristics VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureChanCharacteristics` with the following parameters: **channelList**: `"0"` **inputImpedance**: `NISCOPE_VAL_1_MEG_OHM` **maxInputFrequency**: `20,000,000`

3. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	C/C++ Function Call
 	Call `niScope_ConfigureVertical` with the following parameters: **vi**: The instrument handle from `niScope_init` **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **channelList**: `"0"` **range**: `2.0` **offset**: `0.0` **enabled**: `NISCOPE_VAL_TRUE`

4. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
 	Call `niScope_Configure HorizontalTiming` with the following parameters: **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **vi**: The instrument handle from `niScope_init` **minSampleRate**: **(NI 5132)** `50,000,000`, **(NI 5133)** `100,000,000` **refPosition**: `50.0` **minNumPts**: `1,000,000`

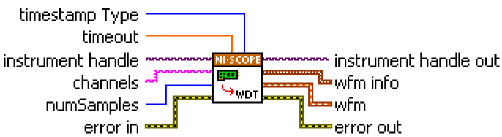
5. Commit all the parameter settings to hardware using the niScope Commit VI.

LabVIEW Block Diagram	C/C++ Function Call
 	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

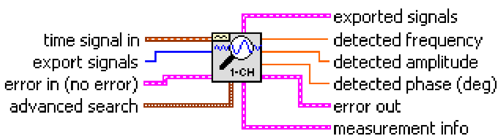
6. Connect the scope calibrator test head directly to the channel 0 input of the digitizer. Configure the calibrator to output an exact 11 MHz sine wave with 1 $V_{pk\text{-}pk}$ amplitude and 1 MΩ load impedance.
7. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.
8. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

9. Retrieve a waveform using the niScope Fetch (poly) VI. Select the WDT instance of the VI. Use the default value for the **timestamp Type** parameter.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Fetch` with the following parameters: **vi**: The instrument handle from `niScope_init` **timeout**: `5.0` **channelList**: `"0"` **numSamples**: `-1`

10. Measure the exact frequency of the peak around 11 MHz using the Extract Single Tone Information VI with the following inputs.
    - **advanced search»approx freq.**: `-1`
    - **advanced search»search**: `5`
    - **export signals**: `0` (none)

LabVIEW Block Diagram	C/C++ Function Call
	Perform an FFT on the array of data from step 9.

11. Calculate the error in timing as parts per million (ppm) using the following formula:

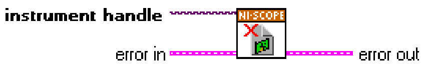
$$error = (a - 11{,}000{,}000) / 11$$

where a is the measured frequency.

If the result of this calculation is within ±50 ppm, the device has passed this portion of the verification.

Note The same time source is used for both channel 0 and channel 1, so you only need to verify the timing accuracy on one channel.

12. End the session using the niScope Close VI.

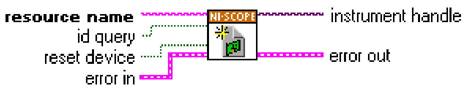
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

You have finished verifying the timing accuracy of the NI 5132/5133.

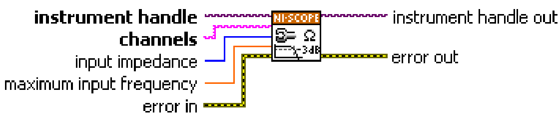
Bandwidth

Complete the following steps to verify the bandwidth of the NI 5132/5133. You must verify both channels with each iteration listed in Table 7.

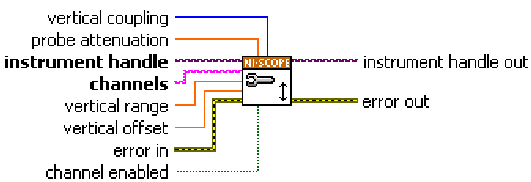
1. Open a session and obtain a session handle using the niScope Initialize VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_init` with the following parameters: **resourceName**: The device name assigned by MAX **idQuery**: `VI_FALSE` **resetDevice**: `VI_TRUE`

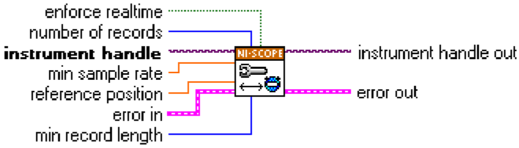
2. Configure the input impedance and the maximum input frequency using the niScope Configure Chan Characteristics VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureChan Characteristics` with the following parameters: **vi**: The instrument handle from `niScope_init` **channelList**: `"0"` **inputImpedance**: `NISCOPE_VAL_I_MEG_OHM` **maxInputFrequency**: The *Maximum Input Frequency* value from Table 7 for the current iteration

3. Configure the common vertical properties using the niScope Configure Vertical VI.

LabVIEW Block Diagram	**C/C++ Function Call**
	Call `niScope_ConfigureVertical` with the following parameters: **vi**: The instrument handle from `niScope_init` **coupling**: `NISCOPE_VAL_DC` **probeAttenuation**: `1.0` **channelList**: `"0"` **range**: The *Range* value from Table 7 for the current iteration **offset**: `0.0` **enabled**: `NISCOPE_VAL_TRUE`

4. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureHorizontalTiming` with the following parameters: **vi**: The instrument handle from `niScope_init` **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **minSampleRate**: `10,000,000` **refPosition**: `50.0` **minNumPts**: `100,000`

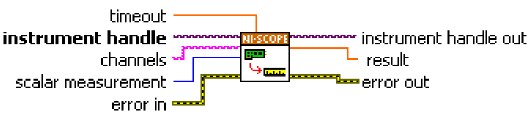
5. Commit all the parameter settings to hardware using the niScope Commit VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_Commit` with the following parameter: **vi**: The instrument handle from `niScope_init`

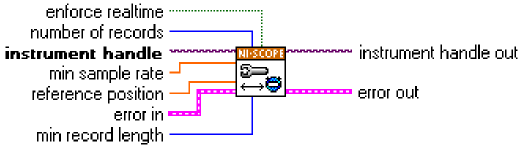
6. Connect the calibrator test head directly to the channel 0 input of the digitizer. Configure the calibrator to output a 50 kHz sine wave of *Input Voltage,* configuring the load impedance of the calibrator to match the input impedance of the digitizer.
7. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.
8. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

9. Fetch a waveform from the digitizer and perform a voltage RMS measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. The resulting value is the *Measured RMS Voltage of 50 kHz Sine Wave* used in step 15.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **vi**: The instrument handle from `niScope_init` **timeout**: `1.0` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_RMS`

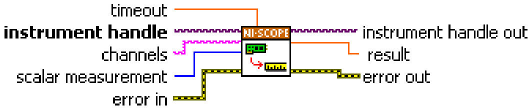
10. Configure the horizontal properties using the niScope Configure Horizontal Timing VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_ConfigureHorizontalTiming` with the following parameters: **vi**: The instrument handle from `niScope_init` **enforceRealtime**: `NISCOPE_VAL_TRUE` **numRecords**: `1` **minSampleRate**: **(NI 5132)** `50,000,000`, **(NI 5133)** `100,000,000` **refPosition**: `50.0` **minNumPts**: `300,000`

11. Configure the calibrator to output the *Input Frequency* for the current iteration in Table 7.
12. Wait 2,500 ms for the output and frequency of the calibrator to settle.
13. Initiate a waveform acquisition using the niScope Initiate Acquisition VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_InitiateAcquisition` with the following parameter: **vi**: The instrument handle from `niScope_init`

14. Fetch a waveform from the digitizer and perform a voltage RMS measurement using the niScope Fetch Measurement (poly) VI. Select the Measurement Scalar DBL instance of the VI. The resulting value is the *Measured RMS Voltage of Generated Sine Wave* used in step 15.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_FetchMeasurement` with the following parameters: **vi**: The instrument handle from `niScope_init` **timeout**: `1.0` **channelList**: `"0"` **scalarMeasFunction**: `NISCOPE_VAL_VOLTAGE_RMS`

15. Calculate the magnitude difference using the following formula:

$$magnitude = (20\log_{10} a) - (20\log_{10} b)$$

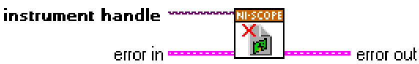
where

a = *Measured RMS Voltage of Generated Sine Wave*

b = *Measured RMS Voltage of 50 kHz Sine Wave*

If the result is within the test limits from Table 7, the device has passed this portion of the verification.

16. Repeat steps 2 through 15 for each iteration listed in Table 7.
17. Move the calibrator test head to the digitizer input channel 1 and repeat steps 2 through 16, replacing `"0"` with `"1"` for the **channelList** parameter.
18. End the session using the niScope Close VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_close` with the following parameter: **vi**: The instrument handle from `niScope_init`

You have finished verifying the bandwidth of the NI 5132/5133.

Table 7. NI 5132/5133 Bandwidth

Iteration	Max Input Frequency (MHz)	Range (Vpp)	Input Voltage (Vpp)	Input Frequency (Hz)	Calibration Test Limits		Published Specifications	
					Max Level (dB)	Min Level (dB)	Max Level (dB)	Min Level (dB)
1	50	0.04	0.032	35,100,000	N/A	–2.5	N/A	–3.0
2	50	0.1	0.08	50,100,000	N/A	–2.5	N/A	–3.0
3	50	0.2	0.16	50,100,000	N/A	–2.5	N/A	–3.0
4	50	0.4	0.32	50,100,000	N/A	–2.5	N/A	–3.0
5	50	1	0.8	50,100,000	N/A	–2.5	N/A	–3.0
6	50	2	1.6	50,100,000	N/A	–2.5	N/A	–3.0
7	50	4	3.2	50,100,000	N/A	–2.5	N/A	–3.0
8	50	10	5.5	50,100,000	N/A	–2.5	N/A	–3.0
9	50	20	5.5	50,100,000	N/A	–2.5	N/A	–3.0
10	50	40	5.5	50,100,000	N/A	–2.5	N/A	–3.0
11	20	0.04	0.032	15,100,000	2.9*	–2.9*	3.0*	–3.0*
12	20	0.04	0.032	25,100,000	–3.0*	N/A	3.0*	N/A

* Published specification value listed for validation of noise filter only. The specification is not included in the device specifications document.

Adjustment

If the NI 5132/5133 successfully passed each of the verification procedures within the calibration test limits, then an adjustment is recommended but not required to warrant the published specifications for the next two years. If the digitizer was not within the calibration test limits for each of the verification procedures, you can perform the adjustment procedure to improve the accuracy of the digitizer. Refer to *Appendix A: Calibration Options* to determine which procedures to perform.

An adjustment is required only once every two years. Following the adjustment procedure automatically updates the calibration date and temperature in the EEPROM of the digitizer.

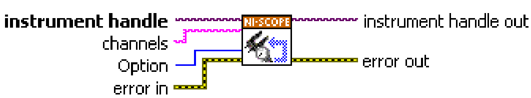
Note If the digitizer passed the entire verification procedure within the calibration test limits and you do not want to perform an adjustment, you can update the calibration date and onboard calibration temperature without making any adjustments by calling *only* niScope Cal Start and niScope Cal End.

Complete the following steps to externally adjust the NI 5132/5133.

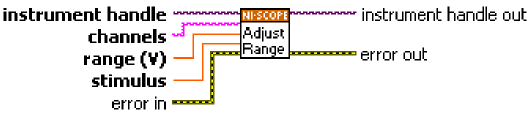
1. Disconnect or disable all inputs to the digitizer.
2. Obtain a calibration session handle using the niScope Cal Start VI.

LabVIEW VI	C/C++ Function Call
	Call `niScope_CalStart` with the following parameters: **resourceName**: The device name assigned by MAX **password**: The password required to open an external calibration session. If this password has not been changed since manufacturing, the password is "`NI`"

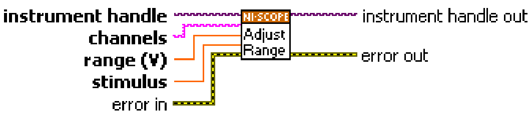
3. Use the niScope Cal Self Calibrate VI to self-calibrate the digitizer.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalSelfCalibrate` with the following parameters: **sessionHandle**: The instrument handle from `niScope_init` **channelList**: `VI_NULL` **option**: `VI_NULL`

4. Connect the scope calibrator test head directly to the channel 0 input of the digitizer.
5. Configure the scope calibrator to output the *Positive Input* voltage for the current iteration listed in Table 8.
6. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.
7. Adjust the vertical range using the niScope Cal Adjust Range VI.

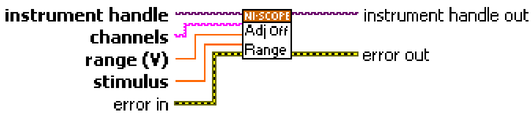
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalAdjustRange` with the following parameters: **sessionHandle**: The session handle from `niScope_CalStart` **channelName**: `"0"` **range**: The *Range* value from Table 8 for the current iteration **stimulus**: The *Positive Input* voltage value from Table 8 for the current iteration.

8. Configure the scope calibrator to output the *Negative Input* voltage from Table 8 for the current iteration.
9. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.
10. Adjust the vertical range using the niScope Cal Adjust Range VI.

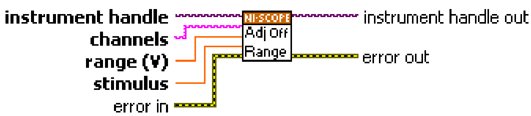
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalAdjustRange` with the following parameters: **sessionHandle**: The session handle from `niScope_CalStart` **channelName**: `"0"` **range**: The *Range* value from Table 8 for the current iteration **stimulus**: The *Negative Input* voltage value from Table 8 for the current iteration.

11. Repeat steps 5 through 10 for each iteration listed in Table 8.
12. Configure the scope calibrator to output the *Negative Input* voltage from Table 9 for the current iteration.
13. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.

14. Adjust the vertical offset using the niScope Cal Adjust Offset Range VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalAdjustOffsetRange` with the following parameters: **sessionHandle**: The session handle from `niScope_CalStart` **channelName**: `"0"` **range**: The *Range* value from Table 9 for the current iteration **stimulus**: The *Negative Input* voltage value from Table 9 for the current iteration.

15. Configure the scope calibrator to output the *Positive Input* voltage from Table 9 for the current iteration.
16. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.
17. Adjust the vertical offset range using the niScope Cal Adjust Offset Range VI.

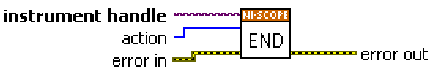
LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalAdjustOffsetRange` with the following parameters: **sessionHandle**: The session handle from `niScope_CalStart` **channelName**: `"0"` **range**: The *Range* value from Table 9 for the current iteration **stimulus**: The *Positive Input* voltage value from Table 9 for the current iteration.

18. Repeat steps 12 through 17 for each iteration listed in Table 9.
19. Configure the scope calibrator to output a 1 kHz, *Input Amplitude* square wave into a 1 MΩ load impedance for the current iteration from Table 10.
20. Wait 2,500 ms for the impedance matching and frequency of the calibrator to settle.

21. Adjust the compensation attenuator using the niScope Cal Adjust Compensation Attenuator VI.

LabVIEW Block Diagram	C/C++ Function Call
	Call `niScope_CalAdjustCompensationAttenuator` with the following parameters: **sessionHandle**: The session handle from `niScope_CalStart` **channelName**: `"0"` **range**: The *Range* value from Table 10 for the current iteration

22. Repeat steps 19 through 21 for each iteration listed in Table 10.
23. Move the calibrator test head to the digitizer input channel 1 and repeat steps 5 through 22, replacing `"0"` with `"1"` for the **channelList** parameter.
24. End the session using niScope Cal End VI.

LabVIEW VI	C/C++ Function Call
	Call `niScope_CalEnd` with the following parameters: **sessionHandle**: The instrument handle from `niScope_CalStart` **action**: `NISCOPE_VAL_ACTION_STORE` to save the results of the calibration

You have finished adjusting the NI 5132/5133. You should repeat the entire *Verification* procedure to verify a successful adjustment.

Table 8. NI 5132/5133 Input Parameters for Vertical Range Adjustment

Iteration	Range (Vpp)	Positive Input (V)	Negative Input (V)
1	0.04	0.018	–0.018
2	0.1	0.045	–0.045
3	0.2	0.09	–0.09
4	0.4	0.18	–0.18
5	1	0.45	–0.45
6	2	0.9	–0.9

Table 8. NI 5132/5133 Input Parameters for Vertical Range Adjustment (Continued)

Iteration	Range (Vpp)	Positive Input (V)	Negative Input (V)
7	4	1.8	–1.8
8	10	4.5	–4.5
9	20	9	–9
10	40	18	–18

Table 9. NI 5132/5133 Input Parameters for Vertical Offset Range Adjustment

Iteration	Range (Vpp)	Positive Input (V)	Negative Input (V)
1	0.1	0.4	–0.4
2	1	4.0	–4.0
3	10	12	–12

Table 10. NI 5132/5133 Input Parameters for Compensation Attenuator Adjustment

Iteration	Range (Vpp)	Input Amplitude (Vpp)
1	2	1.65
2	10	8.5

Appendix A: Calibration Options

External calibration involves verification and if necessary, adjustment and reverification. Adjustment is the process of measuring and compensating for device performance to improve the measurement accuracy. Performing an adjustment updates the calibration date, effectively resetting the calibration interval. The device is warranted to meet or exceed its published specifications for the duration of the calibration interval. Verification is the process of testing the device to ensure that the measurement accuracy is within certain specifications. Verification can be used to ensure that the adjustment process was successful or to determine if the adjustment process needs to be performed at all.

This document provides two sets of test limits for most verification stages—the *calibration test limits* and the *published specifications*. The calibration test limits are more restrictive than the published specifications. If all of the measurement errors determined during verification fall within the calibration test limits, the device is warranted to meet or exceed its published specifications for a full calibration interval (two years). For this reason, you must verify against the calibration test limits when performing verification after adjustment. If all of the measurement errors determined during verification fall within the published specifications, but not within the calibration test limits, the device is meeting its published specifications. However, the device will not necessarily remain within these specifications for an additional two years. The device will meet published specifications for the remainder of the current calibration interval. In this case, you can perform an adjustment if you want to further improve the measurement accuracy or reset the calibration interval. If some measurement errors determined during verification do not fall within the published specifications, you must perform an adjustment to restore the device operation to its published specifications.

The *Complete Calibration* section describes the recommended calibration procedure. The *Optional Calibration* section describes alternative procedures that allow you to skip adjustment if the device already meets its calibration test limits or published specifications.

Complete Calibration

Performing a complete calibration is the recommended way to warrant that the NI 5132/5133 will meet or exceed its published specifications for a two-year calibration interval. At the end of the complete calibration procedure, you verify that the measurement error falls within the calibration test limits. Figure 1 shows the programming flow for complete calibration.

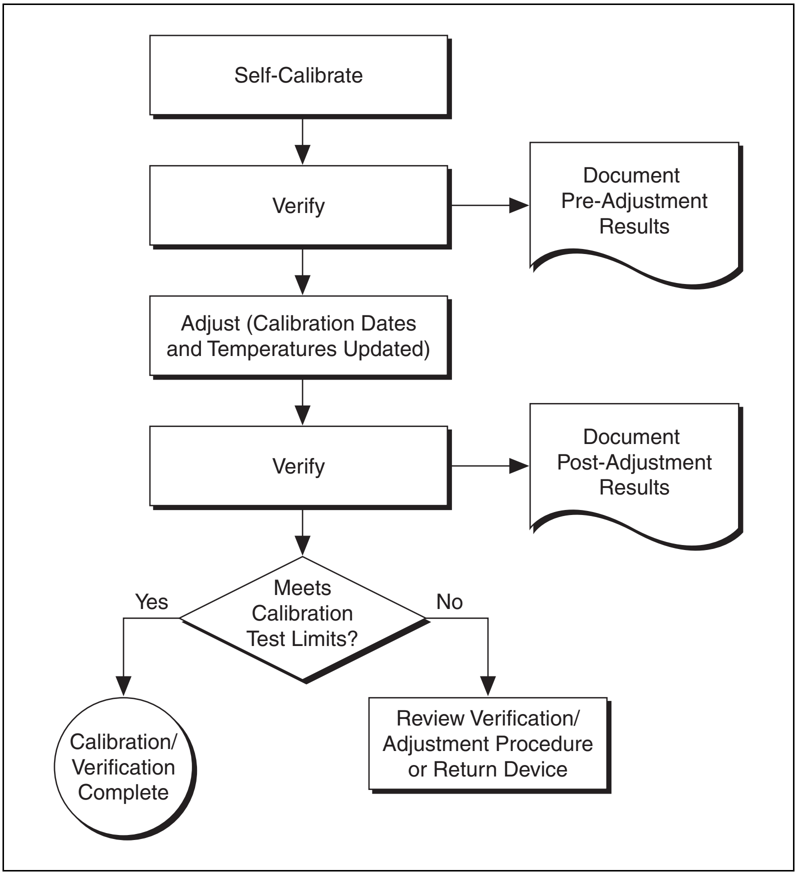


Figure 1. Complete Calibration Programming Flow

Optional Calibration

You can choose to skip the adjustment steps of the calibration procedure if the measurement error is within the calibration test limits or the published specifications during the first verification. If all of the measurement errors determined during the first verification fall within the calibration test limits, the device is warranted to meet or exceed its published specifications for a full calibration interval. In this case, you can update the calibration date, effectively resetting the calibration interval, without actually performing an adjustment. Refer to the *Adjustment* section for more information.

If all of the measurement errors determined during the first verification fall within the published specifications, but not within the calibration test limits, adjustment is also optional. However, you cannot update the calibration date, because the device will not necessarily operate within the published specifications for an additional two years.

Note Regardless of the results of the first verification, if you choose to perform an adjustment, you must verify that the measurement error falls within the calibration test limits at the end of the calibration procedure.

Figure 2 shows the programming flow for the optional calibration.

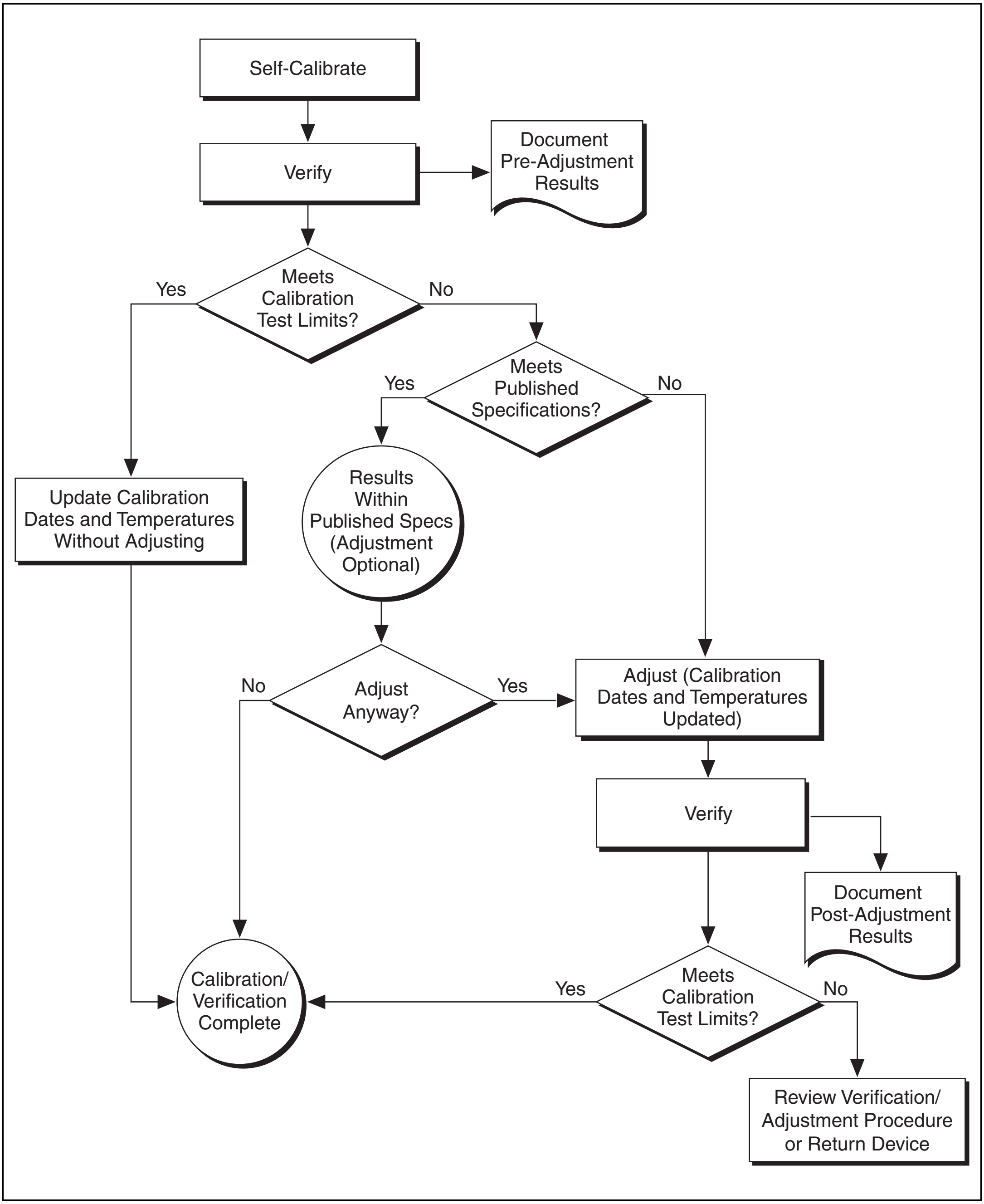


Figure 2. Optional Calibration Programming Flow

Appendix B: Calibration Utilities

NI-SCOPE supports several calibration utilities that allow you to retrieve information about adjustments performed on the NI 5132/5133, change the external calibration password, and store small amounts of information in the onboard EEPROM. You can retrieve some data using MAX, however, you can retrieve all the data programmatically using NI-SCOPE functions.

MAX

To retrieve data using MAX, complete the following steps:

1. Select the device from which you want to retrieve information from **My System»Devices and Interfaces»NI-DAQmx Devices**.
2. Select the **Calibration** tab in the lower right corner.

You should see information about the last date and temperature for both external and self-calibration.

NI-SCOPE

NI-SCOPE provides a full complement of calibration utility functions and VIs. Refer to the *NI High-Speed Digitizers Help* for the complete function reference and VI reference. The utility functions include:

- `niScope_CalChangePassword` (niScope Cal Change Password VI)
- `niScope_CalFetchCount` (niScope Cal Fetch Count VI)
- `niScope_CalFetchDate` (niScope Cal Fetch Date VI)
- `niScope_CalFetchMiscInfo` (niScope Cal Fetch Misc Info VI)
- `niScope_CalFetchTemperature` (niScope Cal Fetch Temperature VI)
- `niScope_CalStoreMiscInfo` (niScope Cal Store Misc Info VI)

Where to Go for Support

The National Instruments Web site is your complete resource for technical support. At `ni.com/support` you have access to everything from troubleshooting and application development self-help resources to email and phone assistance from NI Application Engineers.

A Declaration of Conformity (DoC) is our claim of compliance with the Council of the European Communities using the manufacturer's declaration of conformity. This system affords the user protection for electronic compatibility (EMC) and product safety. You can obtain the DoC for your product by visiting `ni.com/certification`. If your product supports calibration, you can obtain the calibration certificate for your product at `ni.com/calibration`.

National Instruments corporate headquarters is located at 11500 North Mopac Expressway, Austin, Texas, 78759-3504. National Instruments also has offices located around the world to help address your support needs. For telephone support in the United States, create your service request at `ni.com/support` and follow the calling instructions or dial 512 795 8248. For telephone support outside the United States, contact your local branch office:

Australia 1800 300 800, Austria 43 662 457990-0,
Belgium 32 (0) 2 757 0020, Brazil 55 11 3262 3599,
Canada 800 433 3488, China 86 21 5050 9800,
Czech Republic 420 224 235 774, Denmark 45 45 76 26 00,
Finland 358 (0) 9 725 72511, France 01 57 66 24 24,
Germany 49 89 7413130, India 91 80 41190000, Israel 972 3 6393737,
Italy 39 02 41309277, Japan 0120-527196, 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 22 328 90 10, Portugal 351 210 311 210,
Russia 7 495 783 6851, Singapore 1800 226 5886,
Slovenia 386 3 425 42 00, South Africa 27 0 11 805 8197,
Spain 34 91 640 0085, Sweden 46 (0) 8 587 895 00,
Switzerland 41 56 2005151, Taiwan 886 02 2377 2222,
Thailand 662 278 6777, Turkey 90 212 279 3031,
United Kingdom 44 (0) 1635 523545

372511A-01 Apr09