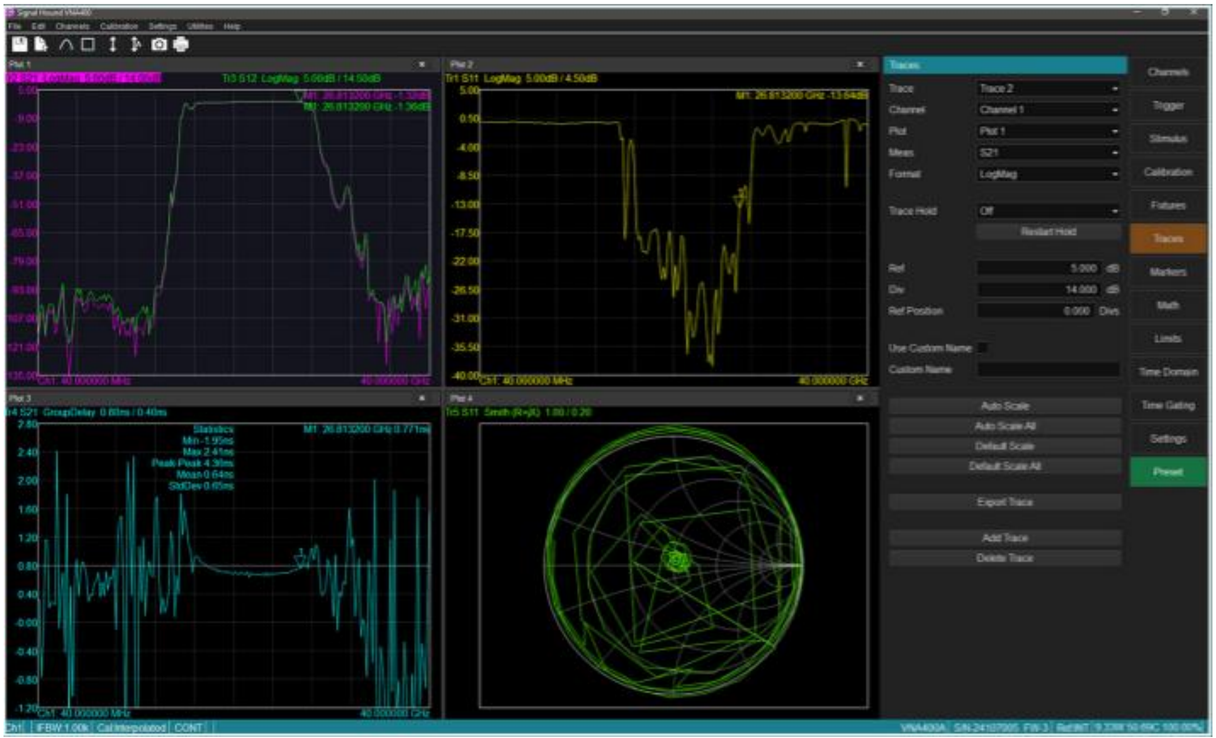
Signal Hound®



Vector Network Analyzer
Software Manual

Signal Hound VNA400 Software Manual

Published 11/7/2025

1502 SE Commerce Ave, Suite 101
Battle Ground, WA
Phone 360-313-7997

Contents

1 Getting Started

1.1 System Requirements

Supported operating systems

- Windows 10/11, 64-bit recommended
- Ubuntu 18.04/20.04, 64-bit

Minimum System Requirements

- Processor
  - X86-64 architecture
  - ARM not supported
- RAM
  - The software will typically requires less than 1GB of memory.
- Storage
  - ~100MB for the VNA application and drivers.
  - All files generated by the software such as calibration data, s2p files, or exported traces are stored on the PC and will require additional space.
    - Typically < 100MB
- Peripheral Support
  - Thunderbolt 3 recommended
  - Or USB-C with 15W power delivery and USB 2.0 data rates.

Signal Hound does not recommend or test using our products in a virtual machine.

1.2 Software Installation

1.2.1 Windows

The latest version of the software can always be found on our website at www.signalhound.com/vnasoftware.

Once downloaded, run the **VNA Installer.msi** and follow the on-screen instructions. You must have administrative privileges to install the software. The installer will also install the necessary USB driver.

It is recommended to install the application into the default location.

1.2.2 Linux

Download the latest Linux application version from the website at www.signalhound.com/vnasoftware and follow the instructions found in the README file included in the download. The Linux application is distributed in a compressed .zip folder and needs to be extracted prior to installation.

1.3 Driver Installation

1.3.1 Windows

The drivers for the VNA are installed during the software installation process and are placed in the application folder. The /drivers/x86/ folder is for 32-bit systems and the /drivers/x64/ is for 64-bit systems.

If for some reason the drivers did not install correctly during the installation process (often due to permissions issues), you can manually install them by navigating to the application directory (where you installed the VNA software) and right clicking on the “Drivers64bit.exe” (if on a 32-bit system, the Drivers32bit.exe file) and clicking “Run as Administrator”. The console output will tell you if the installation was successful.

If the drivers continue to fail to install, contact Signal Hound.

1.3.2 Linux

The Signal Hound USB devices utilize the libusb driver on Linux. Any USB driver specific instructions are found in the README in the Linux software download.

1.4 Connecting the VNA for the First Time

Once the software has been downloaded and installed, you are ready to connect your instrument.

The VNA comes with a supplied USB-C cable. The cable is responsible for all data transfers over the USB 2.0 protocol and for providing up to 15W power to the instrument. For many PCs, a Thunderbolt port will be adequate to power and run the VNA. Most standard USB-C ports will only source 10W, and not be sufficient to power the VNA. In rare cases, a USB-C port will be able to source the required 15W.

Once connected, verify the LEDs on the front panel of the instrument are solid green and that the device shows up in the device manager on Windows, or is listed in lsusb on Linux.

You are now ready to launch the software.

1.5 Running the Software for the First Time

When the software is launched, if only one device is detected, it will attempt to automatically connect to it. If it fails to connect, you will be presented with a warning, and the ability to troubleshoot and reconnect using the File→Connect menu.

The first time a new device is connected and run on a PC, there is additional initialization time required. This initialization time is for reading and caching the factory calibration file.

When you are finished with the device, you may close the software at any time, which will put the VNA into an idle state.

1.6 Software Updates

The latest version of the software is always available on our website www.signalhound.com. The software will also alert you when a newer version is available. This alert will appear in the status bar as well as in the Help → About menu, which also displays the current version information.

2 Basic Concepts

2.1 User Interface

The VNA user interface contains the following key regions

- The file menu. Located along the top of the interface, several drop down menus exist. Many of the file menu options are duplicate in the control panels, offering several ways to accomplish tasks in the software depending on preference.
- The control panels. The primary controls for the instrument and software. See the section on control panels for more information.
- The toolbar. Located above the plot, the toolbar contains several common operations for quick access.
- The plot region. A user configurable region of the interface for organizing and customing plots. See the section on plots for more information.

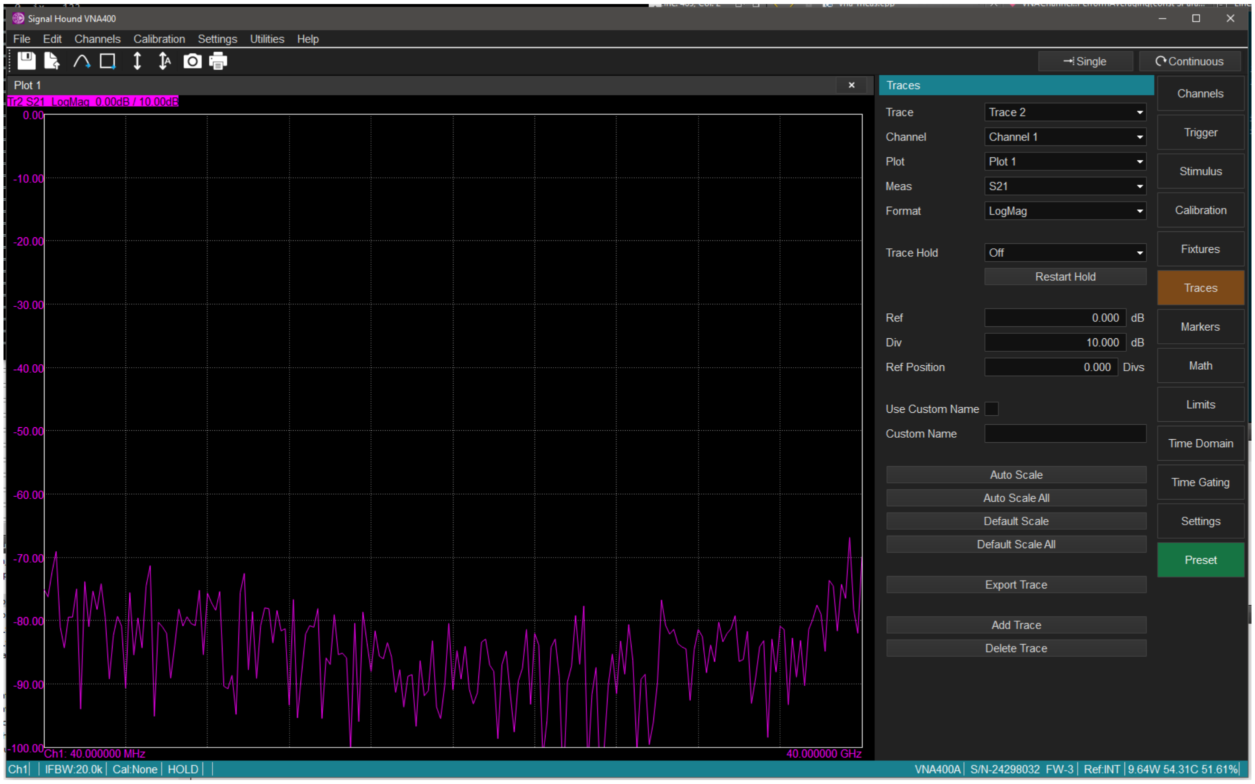


Figure 1: The VNA software user interface.

2.2 Control Panel

The control panel holds all measurement settings and controls. Submenus within the control panel are accessible through the buttons. Only one submenu can be visible at a time. Pressing the button for the active submenu will hide the control panel creating more space for the plots.

The control panel can also be moved to the left hand side of the application by changing the settings in the preferences menu.

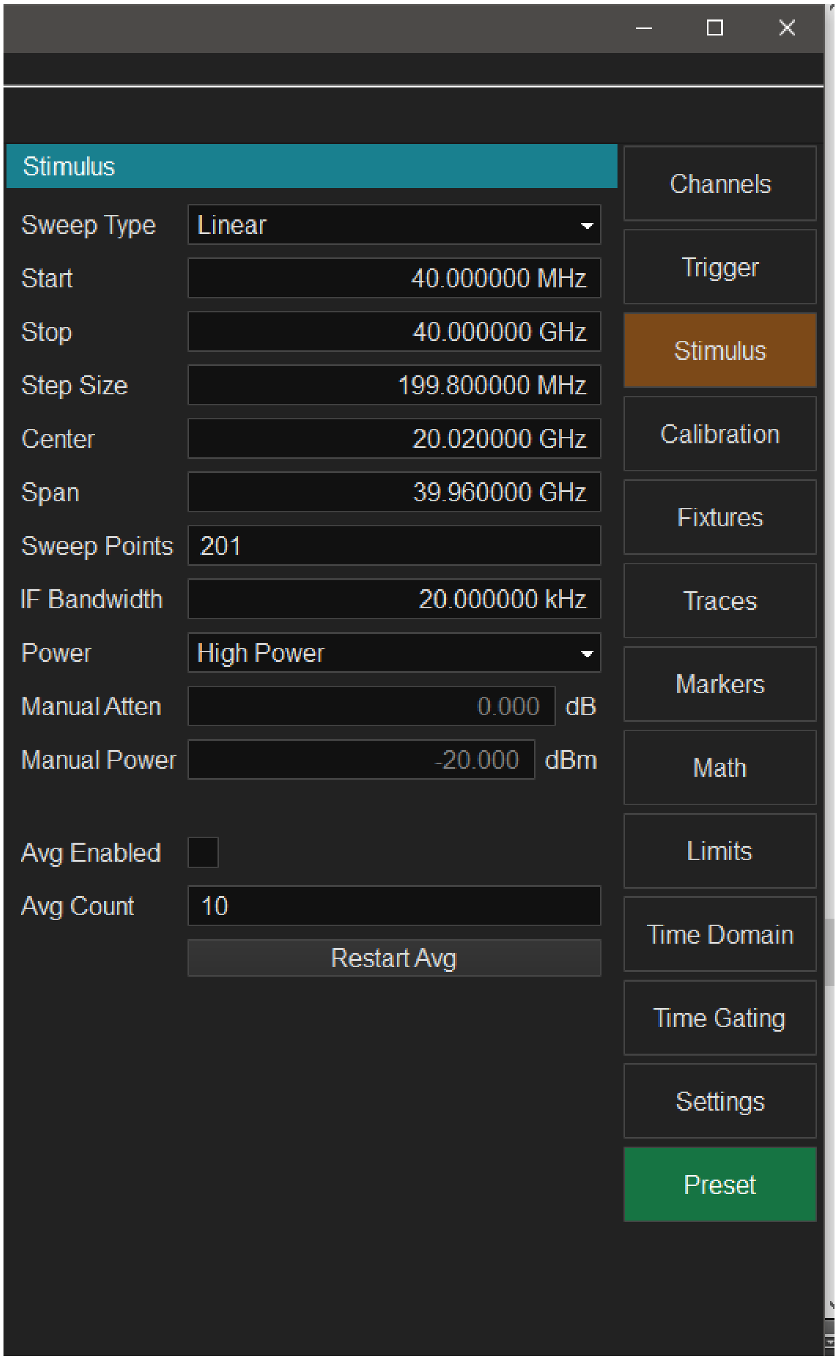


Figure 2 Control panel.

2.3 S-Parameters

The VNA software supports the measurement of ratioed S-parameters. S-parameters describe the behavior of linear networks. S-parameters are complex values. S-parameters can be plotted directly, but commonly are first converted into other formats such as amplitude or phase before being displayed. Formats are covered in the Traces section.

For a 2 port VNA, there are 4 S-parameters. The S-parameters can be defined as,

$$S_{11} = \frac{b_1}{a_1} \quad S_{21} = \frac{b_2}{a_1} \quad S_{12} = \frac{b_1}{a_2} \quad S_{22} = \frac{b_2}{a_2}$$

Where a_N and b_N represent the receivers for the incident and reflected power of port N. The receiver results for a/b are complex values.

2.4 Traces

A trace represents one sweep. The trace is the result of taking one of the ratioed S-parameters, converting it into a specific format, then displaying it in a plot. The s-parameter, format, and plot are all selectable by the user by using the traces control panel, or by using the right click context trace menu on the plot.

A trace is specific to a channel. Multiple traces may be created on each channel. Multiple traces may be displayed on the same plot.

Each trace has it's own set of markers, memory, and math operations. These are all described further in the Data Analysis section.

A trace is displayed on only one plot at a time. The plot in which the trace is displayed can be selected and moved using either the trace control panel or the right click trace context menu within the plot.

2.4.1 Selected Trace

At any given time in the application, a single trace is considered "selected". The channel and plot that that trace is associated with are also considered selected. Many of the control panels are context sensitive and display settings associated with either the selected trace, channel, or plot associated with that trace.

The selected trace is denoted in a couple locations

- In the trace control panel in the trace combo box.
- In the plot of the selected trace, the trace header will be highlighted. See picture below.

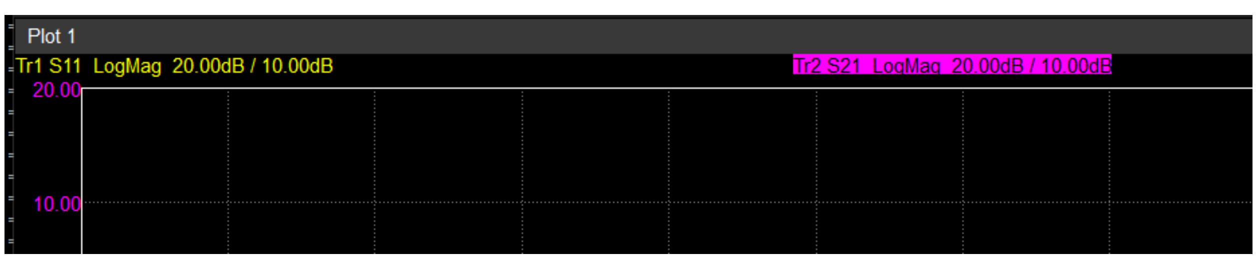


Figure 3: Shows trace 2 as the selected trace.

Similarly the selected trace can be changed by,

- Choosing a trace from the traces control panel.
- Clicking on one of the trace headers above a plot.
- Clicking on any plot that does not contain the selected trace will select one of the traces on that plot.

2.4.2 Formats

The supported set of display formats are:

- Linear Magnitude
- Log Magnitude
- Phase (radian)
- Phase (degrees)
- Unwrapped Phase (degrees)
- SWR
- Smith (R + jX)
- Smith (G + jB)
- Real
- Imag
- Group Delay

2.5 Channels

Channels can be thought of as a complete measurement setup. The VNA software supports several channels to be active at once. Each channel contains its own configuration, calibration, and traces. By default only 1 channel is active.

If multiple channels are active, they are swept in succession. Triggering can be configured separately for each channel.

Traces on different channels can share the same plots. This means that two channels with two different frequency ranges can be displayed on the same plot. It is important to observe the axis and which trace is selected to properly interpret the results in these scenarios.

2.6 Plots

Plots are used to view traces. Plots can hold several traces. Traces can be freely moved between plots. Plots can be manually rearranged in the user interface. Closing a plot containing traces will also delete those traces.

Plots can be create using the “Add Plot” toolbar button, or in the “Add Trace” dialog.

2.6.1 Plot Trace Menu

At the top of each plot is a list of all traces that belong in the plot. Clicking on any of the trace text will select that trace in the application. A full trace menu can be accessed by right-clicking on any of the trace text lines (see picture below).

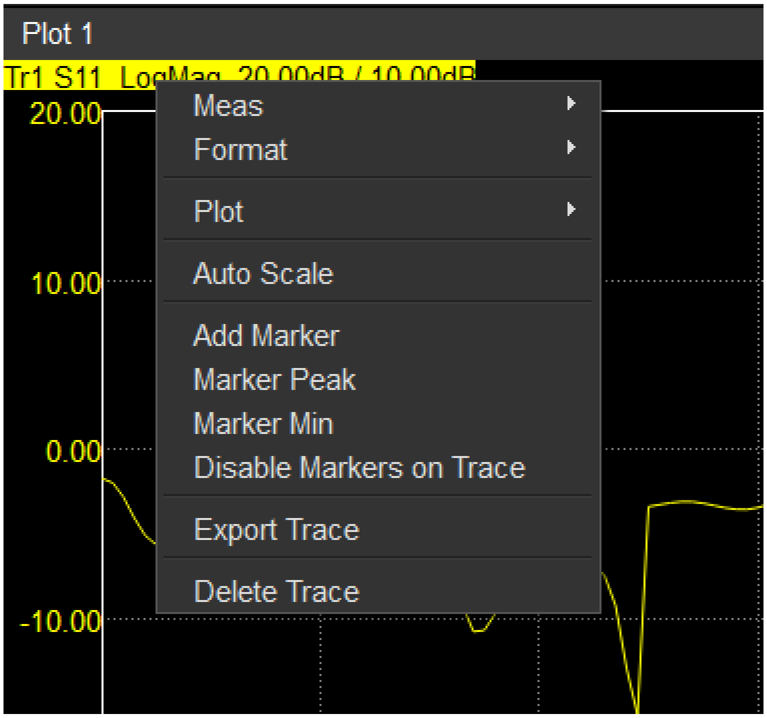


Figure 4: Right click context menu for traces within the plot.

2.6.2 Plot Titles

Titles can be added to plots by right clicking in the graticule and selecting “Set Title”.

2.6.3 Shifting the Y-Axis

The y-axis of a plot can be shifted by left clicking in the left margin of the plot, and dragging up or down to shift the scale. The scale is shifted in increments of the division size. The scale applies only to the the selected trace within the plot.

2.7 Saving and Loading

Saving and loading allows you to revisit measurement scenarios There are 2 ways to save and load measurement state in the VNA software.

2.7.1 Calibration

Once a calibration is complete you are prompted to save the calibration to a file. That file can be loaded in the software at a later date to return back to the calibrated state. The stimulus settings are also stored with the calibration and if the current stimulus settings don’t match what’s in the file being loaded, the user will be presented with the ability to load the calibration stimulus.

When a calibration is loaded, it is loaded into the currently selected channel. It does not modify any current plots or traces.

2.7.2 Full Instrument State

The full instrument state can be saved and loaded using the File -> Save/Recall Instrument State file menu options. When saving and loading the instrument state, all channels, calibrations, plots, traces, and stimulus are affected.

2.8 Presetting the VNA

The full instrument state can be reset using the "Preset" button on the control panel. All channel, trace, marker, calibration, and plot configuration will be reset to their default power on conditions.

2.9 Data Processing Map

The diagram below shows the data processing chain in the VNA software. The diagram also shows where data is retrieved in the processing chain when requested via SCPI.

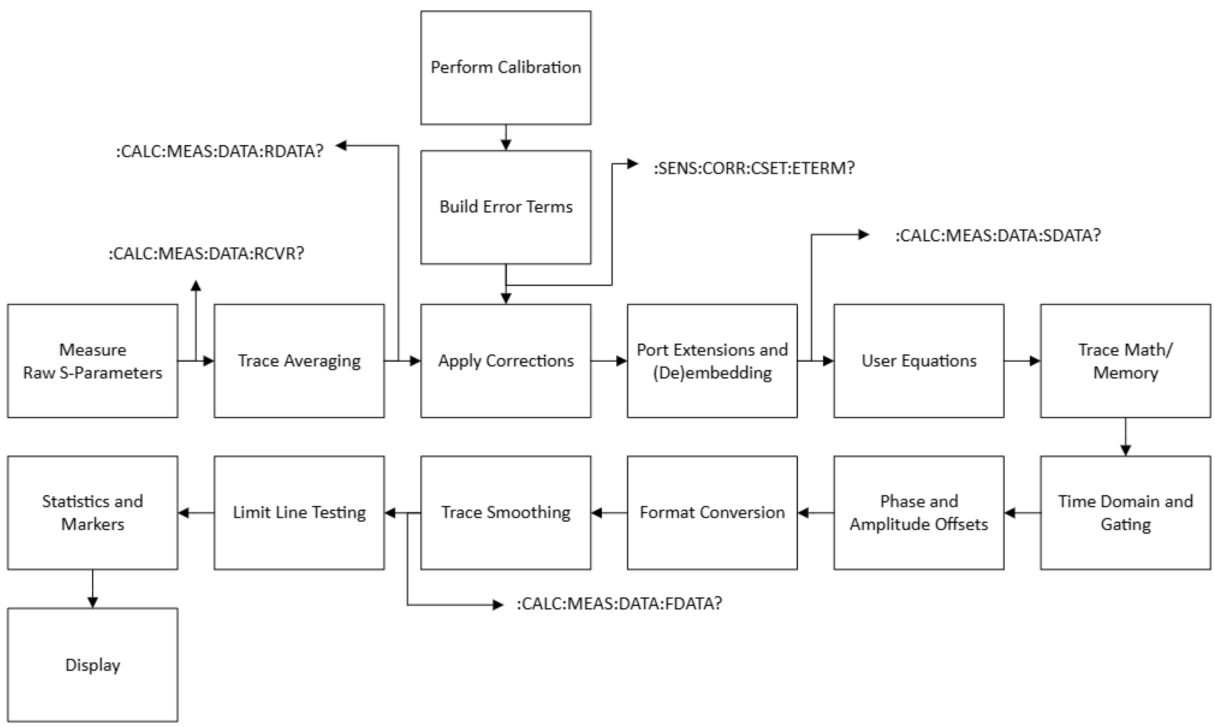


Figure 5 Data processing chain

3 Measurement Setup

3.1 Stimulus

Stimulus refers to all settings that control the acquisition of the S-parameters. This includes the frequency range, amplitude control, averaging, bandwidth, and any other hardware settings.

3.1.1 Frequency Range

The frequency range specifies the exact frequencies at which the sweep is performed. All frequency controls can be found in the stimulus control panel.

3.1.1.1 Frequency Band Crossing

The VNA400 has 2 frequency bands

Band	Start Frequency	Stop Frequency
1	40MHz	10GHz
2	10GHz	40GHz

If your sweep spans both bands it is important to note that when interpolating, either calibration or math/memory, the accuracy at the band crossing cannot be predicted.

3.1.2 Power Level

The transmitter power level can be controlled with the Power, Manual Atten, and Manual Power controls. These controls are found in the stimulus control panel.

The output power range on the VNA400 is roughly 0 to -30dBm. Please refer to the VNA400 product manual for typical output power curves.

The VNA400 has 30dB of attenuation control and is adjustable in 1dB steps.

By default the VNA is configured to output maximum power, which uses 0dB attenuation. If set to low power, the full 30dB of attenuation is used.

Use the manual attenuation control to directly control the attenuation.

Use the manual power control to control the output power level directly. When manual output power is used, the attenuator is adjusted across frequency in attempt to reach the selected output power.

3.1.2.1 RF Output Enabled

The RF power can be turned on or off with the RF Out control on the Stimulus control panel. When disabled, the VNA will continue to sweep but the RF power will be disabled and no power will be delivered to the test port. This setting is global and will apply to all channels.

There are some scenarios where the RF can become enabled again. The RF output state is saved and loaded in user presets, and the RF output state will be re-enabled when performing a system Preset.

Note: There can still be low lying energy present (typical -80dBm) at the port.

3.1.3 Sweep Points

Specifies the number of points in the sweep. Increasing the number of points will decrease the overall sweep time.

3.1.4 IF Bandwidth

IF bandwidth controls the IF bandpass filter width at which each sweep point is measured. Bandwidth is controlled via digital filtering on the VNA400. Decreasing IF bandwidth reduces the noise of the measurement, in effect improving dynamic range and reducing trace noise, at the expense of sweep time.

3.1.5 Averaging

When averaging is enabled, successive sweeps are averaged together in attempt to reduce noise. Sweeps are averaged point by point. Full averaging is not in effect until N sweeps have been collected, where N is the number of averages to be performed. An average counter appears in the status bar when enabled. Any change in stimulus settings will reset the average counter.

If averaging is enabled, averaging will also be used during the calibration procedure. This will increase the time it takes to perform a calibration, but will reduce the noise in all measurements made using this calibration. Averaging does not have to be enabled for the calibration to use averaging after a calibration is performed, and vice-versa.

Averaging is performed using the following formula,

$$UpdatedAverageValue = (NewValue \,/\, n) + (LastAverageValue * \frac{n-1}{n})$$

Where n is the running number of averages performed. This equation ensures that measurements are valid even before the full number of averages are performed.

3.1.6 Reference

The reference setting controls whether the device should use the internal 10MHz reference clock or one provided at the 10MHz input port.

3.2 Triggering

The trigger state controls the acquisition of new sweeps. Each channel has its own trigger state. Trigger state can be changed in the trigger control panel or using the Single and Continuous buttons on the toolbar.

When it is time to start a new sweep the current triggering state determines if a new sweep will take place. If the trigger state is set to continuous or single, a sweep will occur. If set to hold, no sweep will occur. If a state of single triggers a sweep, the state will be set to hold once the sweep completes.

3.2.1 Trigger Scope

Trigger scope affects how a change in trigger state affects other channels. If trigger scope is set to “All Channels” any change in trigger state in the current channel is applied to all other channels. In addition, changing the trigger scope to “All Channels” from “Active Channel” applies the current trigger state to all other channels.

A trigger scope of “Active Channel” means any change to trigger state affects only the currently selected channel. Use “Active Channel” to control each channel independently.

3.2.2 Saving and Loading Trigger State

The trigger state for each channel is stored with the instrument state. This means that if a channels trigger state is set to hold when saving the instrument state, loading that instrument state at a later time will set that channel to hold and it will not be swept until the trigger state is modified.

4 Calibration

Calibration is an important part of VNA measurements. A user calibration using a defined calibration kit is recommended prior to making any VNA measurements. The VNA software supports 1 port short-open-load (SOL), 2 port short-open-load-thru (SOLT), and 2 port short-open-load-reciprocal (SOLR) calibrations using a number of predefined and custom calibration kits.

All calibrations are saved to a file and can be reloaded in the software later.

Each channel has its own calibration, and only one calibration can be active per channel.

4.1 Starting a New Calibration

A new calibration can be started at any time using the calibration control panel or the calibration file menu. When starting a calibration, it will be performed with the active/selected channels settings. When selected, the “Create New Calibration” dialog window will appear.

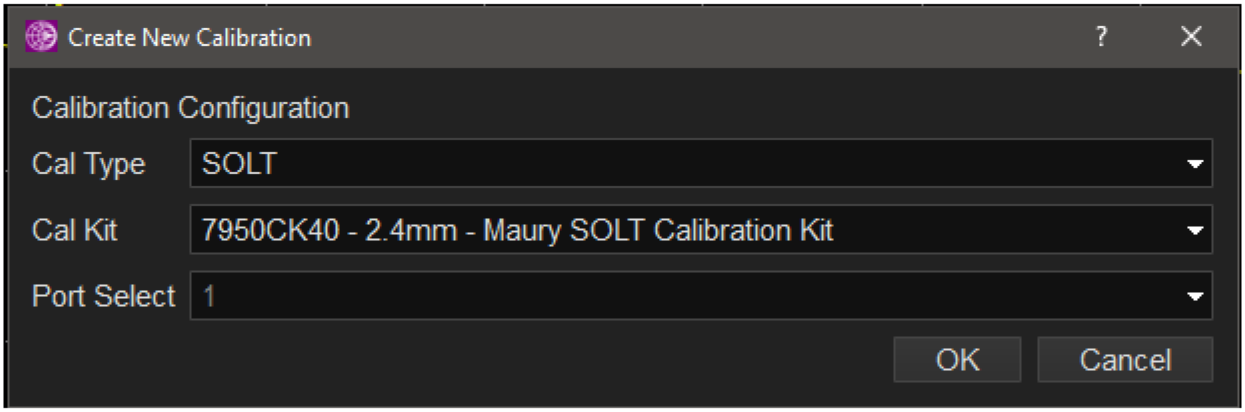


Figure 6: Setup for a new calibration.

In this dialog you will select the calibration type and the calibration kit to be used. A calibration kit must contain all the necessary standards for the given calibration type. For example, a calibration kit without a designated thru will not be able to finish a SOLT kit.

Once you have started the calibration, a new dialog will appear for measuring the standards. Each standard will need to be measured. Selecting a standard will display another dialog that will let you select from any standard in that cal kit that matches that class. Once selected the standard will be measured and will report complete. The standard should be fully connected prior to starting the measurement, and should remain connected until reporting complete.

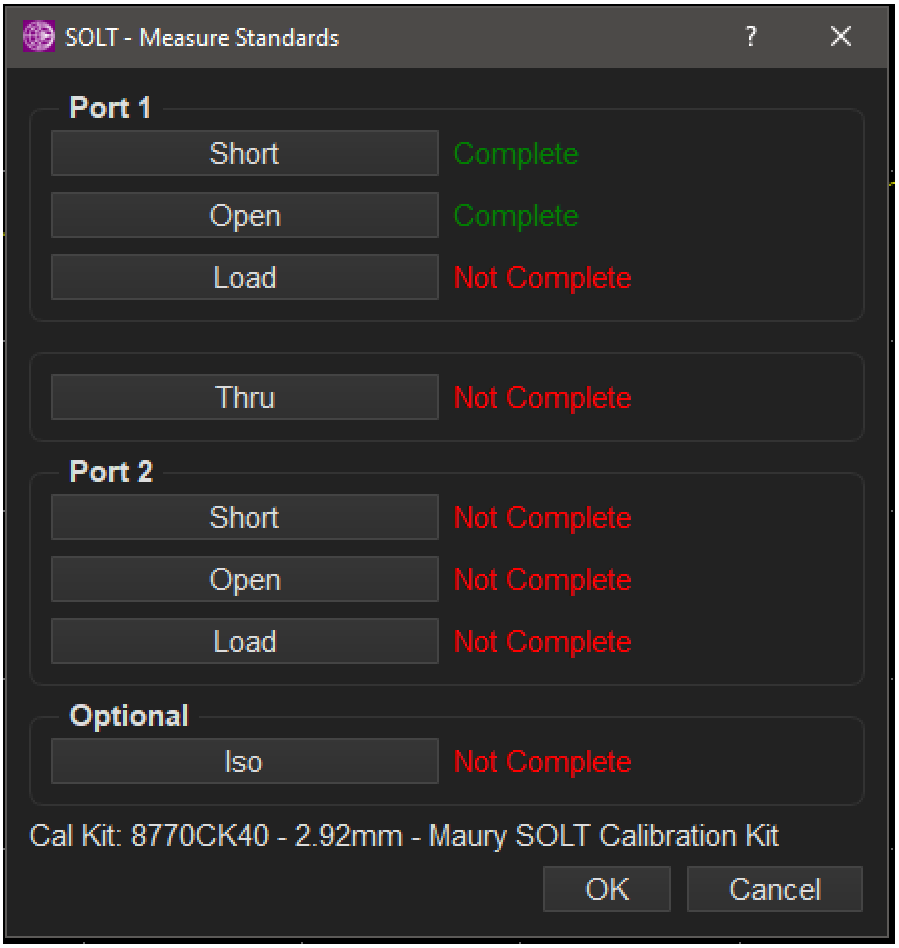


Figure 7: Measure standards dialog.

When finished measuring all standards, press OK. A new dialog will appear for selecting a file name for the calibration. The calibration will be saved to a file and will immediately be loaded into the active channel.

4.2 Calibration Types

4.2.1 SOL

Short/Open/Load (SOL) calibrations are 1 port calibrations used for return loss measurements. During calibration setup, which port to calibrate should be selected. A minimum of 3 measurements are needed, for each the short, open, and load classes.

If using a SOL calibration, any measurements made on the non-calibrated port are invalid.

Either S11 or S22 (depending on which port was calibrated) can be measured with an SOL calibration.

4.2.2 SOLT

Short/Open/Load/Thru (SOLT) calibrations are 2 port bi-directional calibrations. A minimum of 7 measurements are required during the calibration process. This includes 2, 1 port SOL calibrations plus a “thru” measurement. An optional isolation measurement can be performed for measurements that require high dynamic range.

All 4-Sparameters (S11/S21/S12/S22) can be measured with a SOLT calibration.

4.2.3 SOLR

Short/Open/Load/Reciprocal (SOLR) calibrations are 2 port bi-directional calibrations that are similar to the SOLT cal except that the “thru” calibration standard does not have to be specified.

The SOLR calibration is recommended for making measurements on a non-insertable devices and when your calibration kit does not have an adequate characterized thru.

If any phase measurements will be performed on the DUT, ensure the stimulus step size is sufficient to prevent phase aliasing. The software estimates the DUT length during the calibration procedure based on the phase response of the unknown thru. See DUT Length and Aliasing for more information.

Also see Unknown Thru for more information.

4.3 Calibration Kits

Calibration kits are collections of defined standards used to perform VNA calibrations. Commonly used calibrations kits are predefined in the VNA software, with the ability to add and customize additional ones.

View and manage all calibration kits by pressing the “Manage Calibration Kits” button on the “Calibration” control panel.

4.3.1 Editing a Calibration Kit

Only “user” calibration kits can be edited. Creating a user calibration kit can be performed by either copying one of the system calibration kits, or adding a new calibration kit. Editing a user calibration kit will bring up the dialog below.

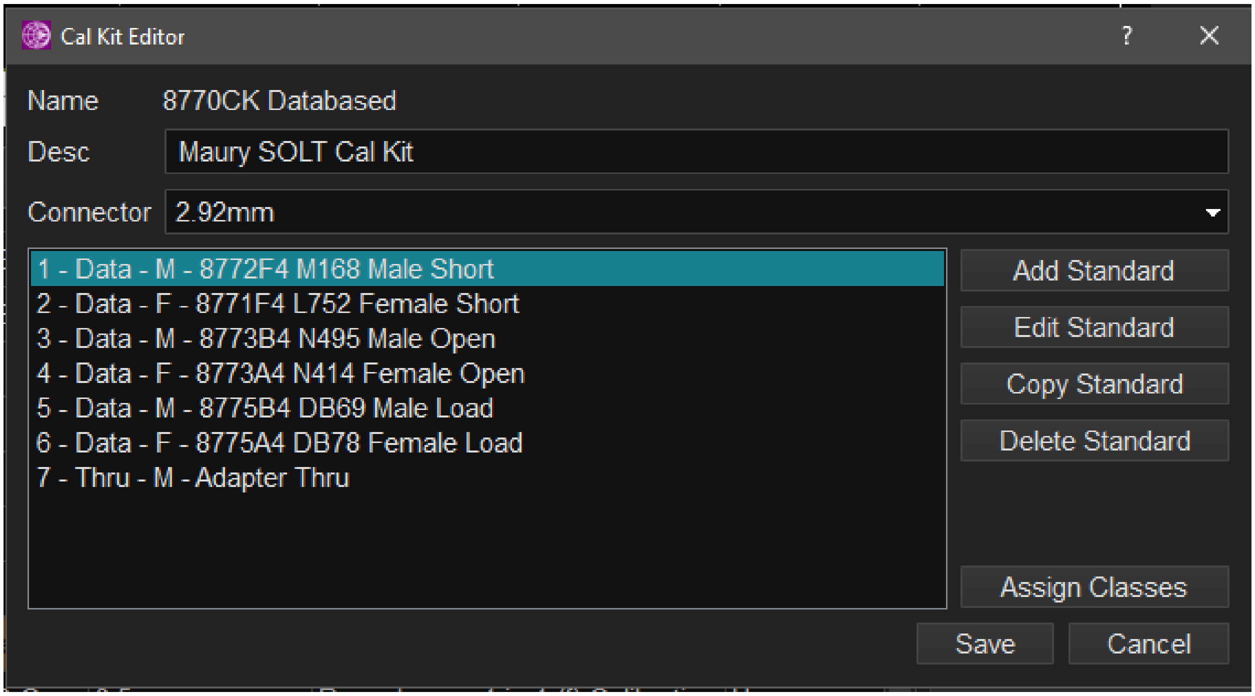


Figure 8: Calibration kit editor dialog.

Within the cal kit editor dialog, you will add and edit all of your standards and assign standard classes. Each standard has a different editor dialog showing only relevant coefficients for that standard type. An example standard editor dialog is below. You would enter the calibration coefficients provided by the calibration kit manufacturer.

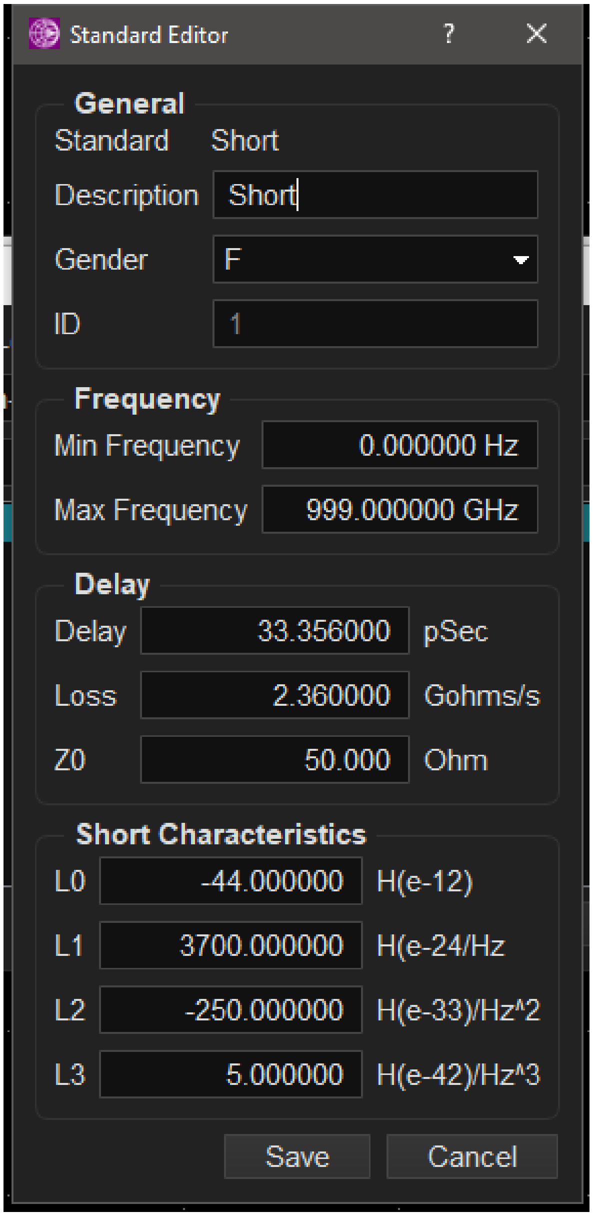


Figure 9: Standard editor for short.

4.3.2 Classes

Each standard in a calibration kit should be assigned a class. The list of possible classes are

- Short
- Open
- Load
- Thru

To use a standard during a calibration, it must be assigned a class. If a standard is not assigned a class, it will not appear as a selection during the calibration process.

When a standard is added, there is an option to automatically assign it the corresponding class. Classes can be assigned to existing standards in the cal kit editor dialog by pressing “Assign Classes”, which brings up a separate dialog. For each class you want to assign, move all relevant standards into the selected column.

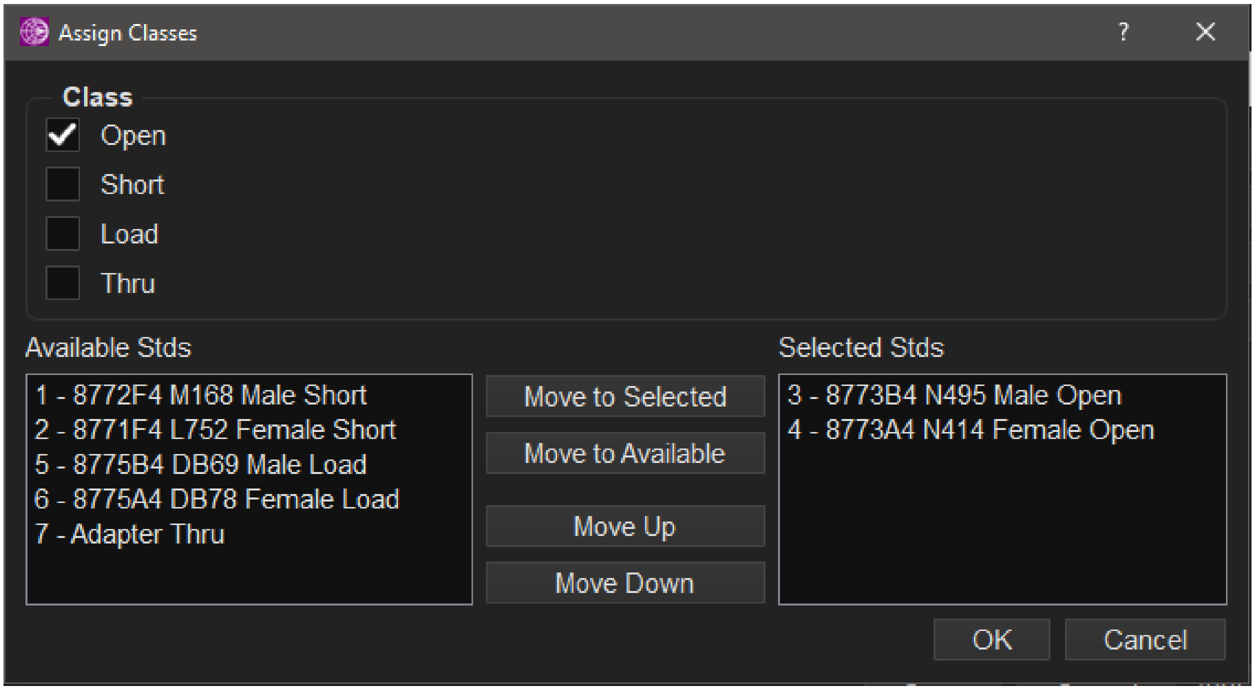


Figure 10: Calibration kit class editor.

4.3.3 Databased Standards

Databased standards are ones in which the standard is defined by an SnP file instead of a set of coefficients. When available, this can make for more accurate calibrations and measurements. In the standard editor, a new control for loading an SnP file is present. The dialog is shown below. Databased standards must still be assigned to a class.

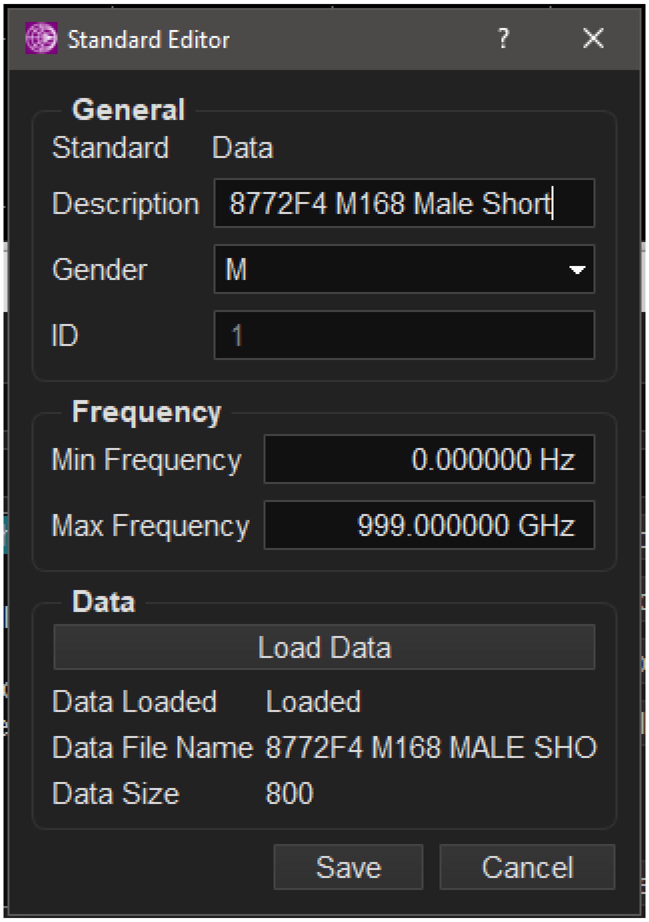


Figure 11: Editing a databased standard.

4.3.4 Frequency Range

Each standard is defined across a frequency range. The frequency range of all standards combined for any given class must cover the frequency range of the stimulus at the time of calibration.

When using multiple standards to cover the frequency range of the stimulus, the frequency range of the most recently measured standard will have priority over previously measured standards. For example, if the measurement stimulus is 40MHz to 40GHz, and you have 2 standards that cover the following frequency ranges,

- Standard A: 40MHz to 2GHz
- Standard B: 1GHz to 40GHz

If the standards are measured in the following order, A then B, the calibration will use standard A from 40MHz to 1GHz and standard B from 1GHz to 40GHz. If the measured order was reversed, B then A, then the calibration will use standard A from 40MHz to 2GHz and standard B from 2GHz to 40GHz.

4.4 Performing a Calibration

4.4.1 Different Types of Thrus

How you calibrate your thru will depend on whether your DUT is insertable, and what type of thru is available to you.

4.4.1.1 Defined Thru

A defined thru is one that is characterized either using the delay and loss terms in the cal kit standard editor, or it can be data-based with an .s2p file.

A defined thru can be used to calibrate both an insertable and non-insertable device.

Ideally a defined thru has the same connectors and gender as the DUT and is inserted during the thru step. It is then removed and replaced with the DUT when the calibration is completed.

A defined thru can be less accurate than alternative thru options, due to the characterization of the thru becoming less accurate over time through regular use. For this reason, it might be necessary to re-characterize the thru periodically. If characterizing the thru is not possible, consider using an unknown thru.

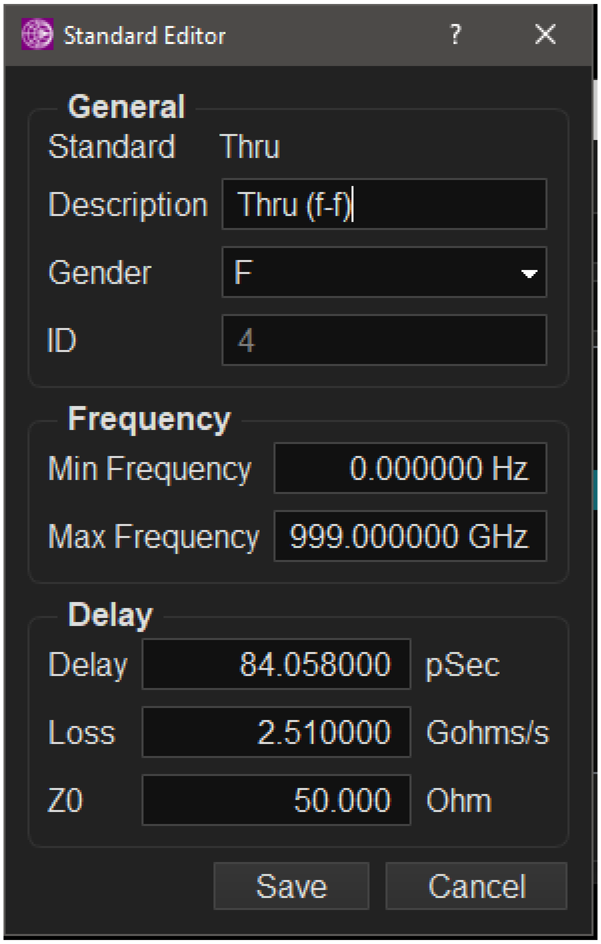


Figure 12 An example of a defined thru using the Delay and Loss values.

4.4.1.1.1 Test Procedure

This is the test procedure when using a defined thru.

- Perform 1 port calibration at the end of port 1.
- Perform 1 port calibration at the end of port 2.
- Insert the defined thru between port 1 and 2.
- Measure the thru using the defined thru standard.
- Complete calibration.
- Remove defined thru.
- Insert the DUT between port 1 and 2.

4.4.1.2 Zero-length/Flush Thru

When measuring insertable devices, using the zero-length thru method is ideal. The thru calibration is performed by connecting the test ports directly together during the thru measurement. Once the calibration is complete, the test ports are then directly connected to the DUT.

This method is in general more accurate than using a defined thru, and does not require an actual physical thru standard.

To configure a flush thru, leave the delay and loss characteristics set to zero when defining the thru in the calibration kit.

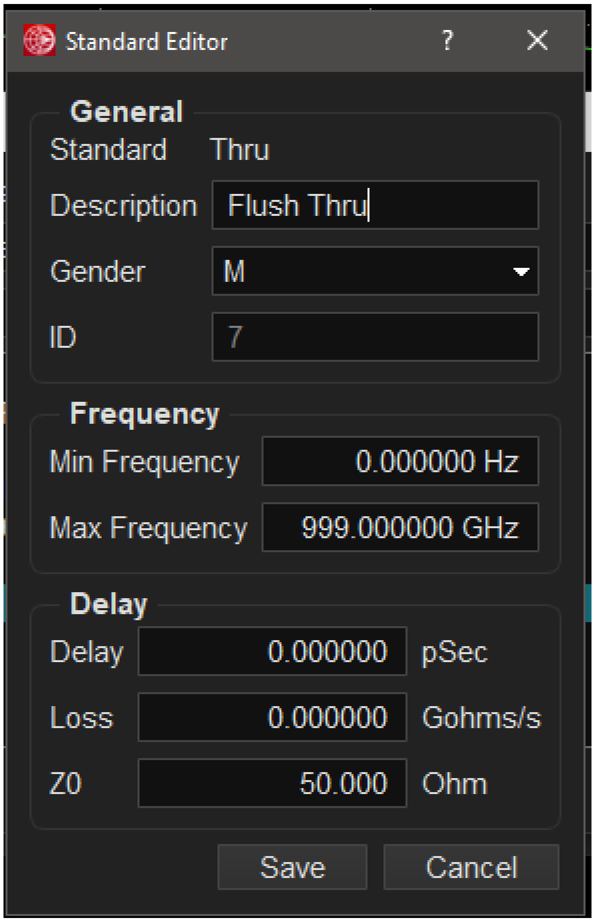


Figure 13: Example of a flush thru cal kit standard.

4.4.1.2.1 Test Procedure

This is the test procedure for the zero-length thru method.

- Perform a 1 port calibration at the end of port 1.
- Perform a 1 port calibration at the end of port 2.
- Connect port 1 to port 2.
- Measure the thru using a thru standard that has zero delay/loss.
- Complete calibration.
- Disconnect port 1 from port 2.
- Insert the DUT between port 1 and port 2.

4.4.1.3 Unknown Thru

Then unknown thru method is ideal for measuring non-insertable devices. It is a method for characterizing and removing the contribution of a non-defined/unknown thru. A requirement to using this method is that the unknown thru used must be reciprocal (S21 = S12).

4.4.1.3.1 Test Procedure

This the the test procedure for the unknown thru method.

- Perform a 1 port calibration at the end of port 1.
- Perform a 1 port calibration at the end of port 2.
- Connect a reciprocal thru between port 1 and port 2.
- Measure the thru using an unknown thru standard.
- Complete calibration.
- Remove reciprocal thru between port 1 and port 2.
- Insert the DUT between port 1 and port 2.

4.4.2 Isolation

When performing an SOLT calibration, an optional isolation step can be performed. During this step, each port should be terminated with a 50Ohm load. No standard or class needs to be associated with this measurement.

4.4.3 Trace Updates During Calibration

During a calibration and after measuring each standard, the traces associated with the channel being calibrated are updated. This enabled visuals confirmation of the measurement. Only the traces visible will be updated. For instance, if only an S11 trace is active during a SOLT calibration, only the S11 will be shown after each measurement. To see all S-parameters during a SOLT calibration, activate all S-parameters prior to starting the calibration.

4.5 Temperature

For best measurement accuracy, is it important to let the internal device temperature settle. Signal Hound recommends a 45 minute warm up period after powering on the device. The device may sit idle or run in the application during this warm up period.

The VNA400 has 2 fans that provide temperature control. These fans assist in maintaining a stable temperature over ~8C temperature range. When no calibration is loaded, the device uses a default 50% fan speed. This allows the unit to settle into the middle of the temperature range that the fan can control.

The temperature of the VNA is captured at the end of a calibration. When that calibration is loaded, the fans are set to target this stored temperature. This ensures the device is as close to the same temperature conditions as when calibrated. This will minimize errors associated with temperature drift.

When multiple channels are present, each with separate calibrations, the average of all calibration temperatures is used to for the temperature set point.

Given this, it is important to ensure the VNA temperature has settled before starting the calibration. Any temperature drift that occurs over the calibration procedure will contribute to errors in the calibrated measurement.

If the device is not given enough time to warm up, the captured temperature may be too low for the fans to achieve, contributing to measurement error.

The fan speed and temperature setpoint can be overridden using the Fan Control dialog in the Settings file menu.

4.5.1 Temperature Warning

The software will display a temperature warning in certain conditions. The warning will appear as orange text in the status bar.

Figure 14: Temperature warning, temp readout highlighted in orange.

The conditions are as follows:

- If the unit has not yet reached 40C and the unit has been powered on for less than 20 minutes. In the event that the unit is operating in cold ambient temperatures and the unit will not reach 40C, this warning can be ignored if the user is certain the device temperature has settled.

- If the temperature difference is greater than 2 degress Celsius from the current set point. This condition will occur when loading a calibration which updates the setpoint to the temperature captured during the time of calibration, and that temperature differs from the current device temperature. If possible, to achieve best measurement result, it is best to wait until the device temperature reaches the target set point. If the calibration occurred in a different environment (significant difference in ambient temperature), the set point may not be reachable, and a new calibration should be performed.

4.6 Interpolation

If the current stimulus does not match the stimulus used during the calibration, the calibration data/error terms must be interpolated. This will introduce undefined error in the measurements.

Interpolation error can be minimized by ensuring the number of measurement points is dense enough to not introduce large phase rotations between adjacent points.

Additional consideration must be taken when interpolating across hardware frequency bands. The VNA400 has 2 frequency bands, split at 10GHz. Any points interpolated across this band break will see measurement error.

It is recommended that interpolation is avoided when possible.

The status bar will indicate the current state of the calibration and if any interpolation is being performed. These are the following indications,

- **Cal:Factory**: Only the factory calibration is being used. No user calibration is active.
- **Cal:Disabled**: A calibration is loaded, but disabled.
- **Cal:Full N-Port**: A calibration is loaded and no interpolation is being performed. This has the highest accuracy.
- **Cal:Interpolated**: A calibration is loaded and interpolation is being performed.
- **Cal:Changed**: A calibration is loaded and active, but the stimulus has changed in a way that interpolation is not possible.

4.7 Port Extension

Port extensions allow you to extend and move the reference plane. This is needed if you perform a calibration at one reference plane, and then extend the port with a test fixture such as a cable/adapter/etc. If you can fully characterize the test fixture (S-parameters), it may be more appropriate to use the deembedding feature.

Port extensions work by accounting for additional phase delay and amplitude loss in the extended port. See the data processing map to see where this correction is applied. Port extensions work on each port independently, I.E. can be applied to either port, or both simultaneously.

Port extensions are applied per channel.

Port extensions settings can be manually entered or measured automatically. Measuring them automatically will generally result in a better measurement. If you know your loss and delay values (from a previous measurement for instance, you can manually enter them.

4.7.1 Automatic Port Extension

To measure the port extension automatically, you will connect the test fixture to your test port after performing and applying a calibration. Then you will need to connect and measure a short and open at the end of your test fixture using the Port Extension dialog “Measure Open” and “Measure Short” buttons. This will automatically populate the loss and delay values. This has the effect of moving your reference plane to the short/open. This procedure can be done for each port independently.

4.7.2 Loss

The loss is characterized by a curve using the freq/loss pairs defined in the port extension dialog. For this reason, port extensions will work better for ideal transmission lines. The curve is defined with the following equations,

When one loss pair is defined and enabled,

$$Loss\ at\ Freq\ F = Loss_{DC} + Loss_1 * \sqrt{\frac{F}{F_1}}$$

When both loss pairs are enabled,

$$Loss\ at\ Freq\ F = Loss_{DC} + Loss_1 * \left(\frac{F}{F_1}\right)^N$$

$$N = \frac{Log_{10}\left|\frac{Loss_1}{Loss_2}\right|}{Log_{10}\left(\frac{F_1}{F_2}\right)}$$

4.7.3 Procedure

This is the procedure for extending a port using automatic port extensions.

1. Perform a 1 or 2 port calibration.
2. Attach your fixture on the calibrated port.
3. In the fixtures control panel, press the “Config Port Extensions” button.
4. Connect an open to the fixture (or leave it disconnected)
5. Press the “Measure Open” button on the Port Extensions control panel.
6. Connect a short to the fixture
7. Press the “Measure Short” button on the Port Extensions control panel.
8. Verify the estimated loss and delay values are correct.
9. Repeat steps 5,6,7,8 on the other port if desired.
10. Press “OK” on the Port Extensions control panel.
11. Enable Port Extensions in the fixture control panel.
12. Insert your DUT and perform your measurements as normal.

4.8 De/embedding and Fixture Removal

Deembedding is the process of removing the response from the measurement, of a test fixture attached to the test port and effectively moving the reference plane. The test fixture needs to be characterized using a touchstone file.

Embedding is the process of adding the response of a test fixture to the measurement, rather than removing it. Both Deembedding and embedding can be enabled independently, and can be configured for both ports.

Use the fixture control panel to load touchstone files for the test fixtures to be de/embedded. These files should be full 2 port characterizations.

See the data processing map to see where de/embedding is performed.

4.9 Factory Calibration

All Signal Hound VNAs come with a factory calibration. When a user calibration is not active, the factory calibration is used. The factory calibration is a full 2 port SOLT calibration. The factory cal can be used for basic measurements but for best performance it is recommended the customer perform their own calibrations prior to their measurements.

The factory calibration is performed with a large sweep size with the goal of having better interpolation performance.

5 Data Analysis

5.1 Markers

Markers provide a way to make measurements on a trace. A user can activate up to 15 markers and 1 reference marker per trace.

Markers display measurement data in the same format of the trace.

Markers can be placed on a trace by clicking anywhere on the graticule of a plot. A marker is placed on the selected trace. Markers can also be activated with the various marker search capabilities or through marker coupling.

Marker readouts are automatically updated when the trace is updated, unless “fixed” is activated.

5.1.1 Searching

The peak search capabilities make it quick to move a marker to points of interest. Peak search other than min/max peak operate only on peaks that meet the peak criteria. To be a valid peak, the peak must exceed the marker threshold setting, and the peak must be higher than surrounding trace points by the value set in the marker excursion setting. The following peak search features are supported.

- Peak Search – Move the marker to the largest value in the trace.
- Min Peak – Move the marker to the smallest value in the trace.
- Peak Left – Move the marker to the next peak to the left of the current marker position that meets the peak criteria. If no peak is found, no action occurs.
- Peak Right – Move the marker the next peak to the right of the current marker position that meets the peak criteria. If no peak is found, no action occurs.

- Next Peak – Moves the marker to the next highest peak that meets the peak criteria. If no peak is found, no action occurs.

5.1.2 Coupling

Coupling is useful when a user wants to synchronize multiple markers across several traces simultaneously. The 3 types of coupling are defined,

- Off – No marker coupling is performed. Moving any marker only affects the currently selected marker.
- Channel – When moving a marker on a trace, for all other traces in that channel, move the same numbered marker on each trace to the same x-position.
- All – When moving a marker on a trace, for all traces active in the software, move the same numbered marker on each trace to the same x-position.

5.1.3 Tracking

Tracking provides a way to automate search functions. For example, setting tracking to "Max" is equivalent to executing a peak search after every measurement.

If coupling is enabled, non-tracking markers will be coupled to the tracking marker in the highest numbered trace.

5.1.4 Reference Marker

For each trace 1 reference marker can be activated. The reference marker functions as a normal marker. When the reference marker is activated, the difference between all non-reference markers can be measured. This difference can be measured by enabling "Delta" on any non reference marker.

If the reference marker is not active when a marker is set to delta, the reference marker is activated and set to the current marker position.

The reference marker can be moved to the current marker position at any time by pressing the "Mkr -> Ref Mkr" button.

5.1.5 Bandwidth Markers

Each trace has one bandwidth marker. A bandwidth marker is useful for measuring a bandpass filter response.

Bandwidth markers do not override the normal markers.

The algorithm for the bandwidth marker is,

- Find the peak value on the trace.
- Search both left and right of the peak for a total drop in signal level of N-dB, where N is selectable in the BW level setting.

If a valid bandpass signal is measured, 3 visual markers are placed on the trace, one at the peak, and one at both N-dB points.

By default the bandwidth marker does not update on each trace update, but this can be performed by enabling the BW Tracking setting.

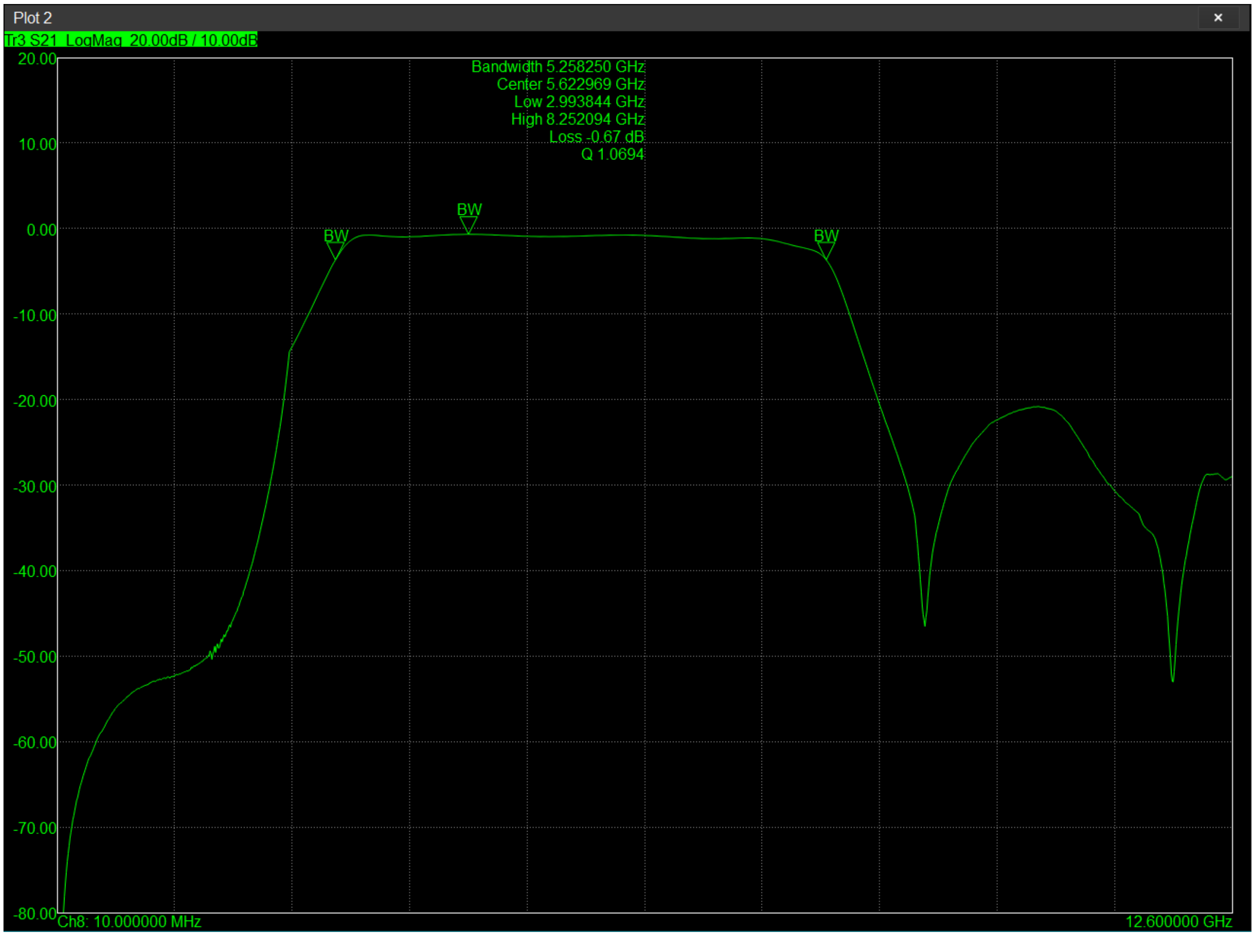


Figure 15 Bandwidth markers measuring a bandpass filter

5.2 Traces

Traces can be added using either the “Add Trace” toolbar button, or the “Add Trace” button on the trace control panel. Adding a new trace will bring up the New Trace dialog. This dialog allows full configuration of the trace prior to adding it to the software.

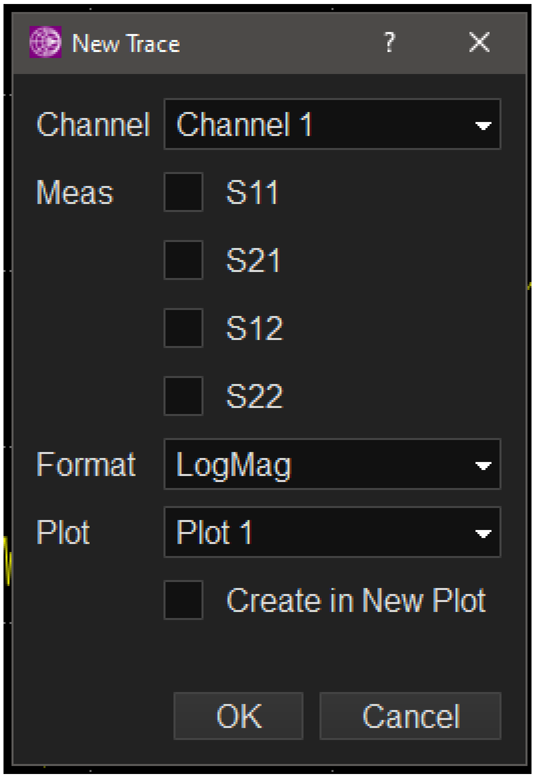


Figure 16 New Trace Dialog

The currently selected trace can be configured using the trace control panel.

5.2.1 Manual Scale

Each trace has it's own plot scaling for each possible format that can be assigned to the trace.

The plot scaling can be manually configured using the Ref, Div, and Ref Position control in the trace control panel. Additionally, it is possible to click and drag on the y-axis on the plot to change the reference.

5.2.2 Auto Scale

Scaling can be performed automatically for any given trace by selecting the "Auto Scale" button on the control panel. Auto scale will adjust the reference and div based on the current data stored in the trace.

Auto scaling can also be applied to all traces in the software simultaneously by pressing the "Auto Scale All" button.

5.2.3 Default Scale

Traces can be reset to their default scale by pressing the "Default Scale" button on the control panel. The scale for all formats will be reset to their default values, not just the currenty selected trace format.

5.2.4 Scale Coupling

Scale coupling affects how other traces are scaled when changing the scale of the selected trace. Only traces that have the same format will couple. The ref, div, and ref position settings couple. Scale coupling can be changed in the trace control panel and is set system wide.

Scale coupling method

- **Off:** No scale coupling.
- **Plot:** All traces that share the same plot as the selected trace will be updated to match the scale of the selected trace.
- **All:** All traces are updated to match the scale of the selected trace.

5.3 Trace Math

Trace math encompasses several operations that can be configured on any given trace. See the Data Processing Map for more information on when math is performed during the measurement.

5.3.1 Math Memory Operations

The math memory operation allows a user to store the current s-parameters into memory, and then compare future sweeps to the stored data. When the "Data→Memory" button is pressed on the Math control panel, the s-parameters on the current channel are saved into a memory buffer. At that point the trace can be configured to operated on either the memory, or a math operation involving the memory. All arithmetic operations +,-,*,/ are available as well as plotting the memory directly. These operations are performed on the complex s-parameter data.

A common use case for math memory is to store a baseline, insert a new DUT, and use the division math operation to plot the difference between the baseline and system with DUT, effectively plotting just the response of the DUT.

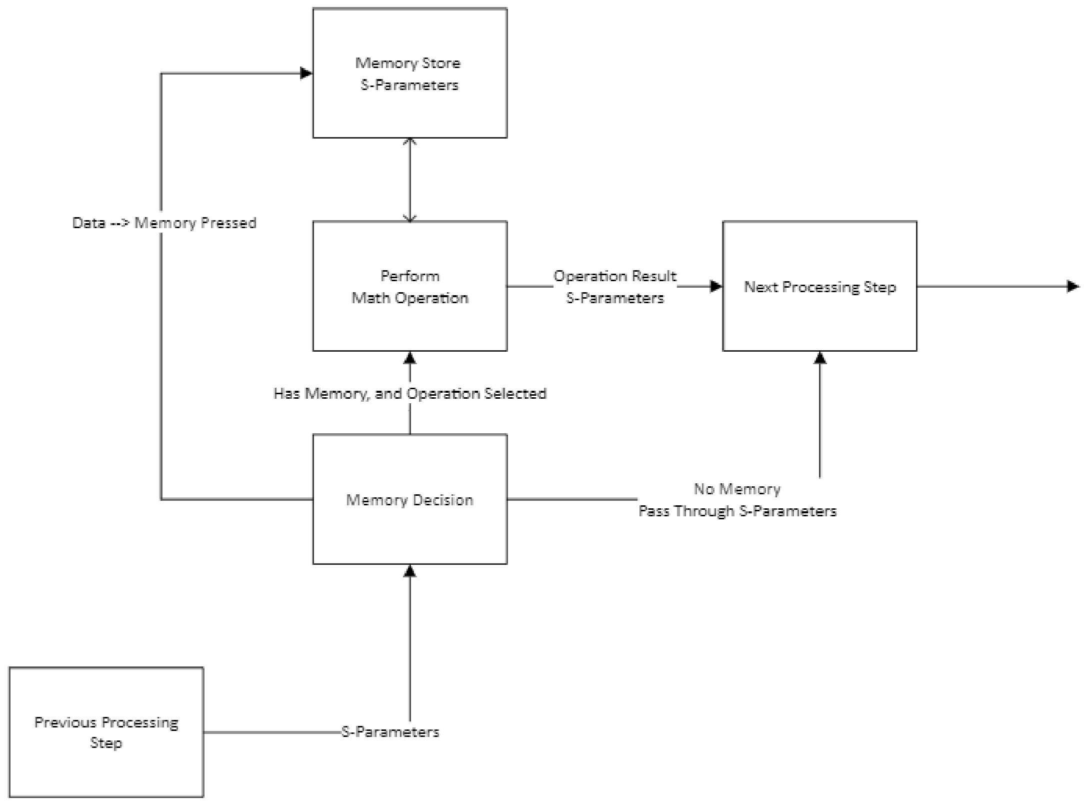


Figure 17 Math Memory Processing

5.3.1.1 Interpolation

If the stimulus for the stored math memory does not match the current stimulus, interpolation of the memory can be performed in attempt to continue performing the math memory operation. This will work as long as the new stimulus frequency range does not exceed the frequency range of the memory stimulus. In that event, the math memory operation will be disabled.

It is recommended to avoid interpolation when possible.

5.3.2 Smoothing

Trace smoothing is a moving average window that averages several adjacent points in attempt to reduce noise of the measurement. The averaging occurs on the display units, that is after the format conversion. The number of points is a function of the smoothing aperture size, which is a percentage of the overall number of points. The window size is also rounded up to the next odd number.

For example, a 1001 point smoothed trace with aperture size of 2%, would result in a moving average window size of

$$
\begin{aligned}
WindowSize &= oddCeil(floor(1001 * 0.02)) \\
&= oddCeil(floor(20.02)) \\
&= oddCeil(20) \\
&= \ 21
\end{aligned}
$$

5.3.3 Equation Editor

The equation editor allows a user to create a trace using an equation that is an algebraic combination of other traces or s-parameters. The equation can be edited using the equation editor dialog. Equations can be stored and recalled later.

Traces are processed in numerical order, so a trace that is a result of an equation using other traces should have a higher trace number to ensure the traces it relies upon are updated prior to executing the equation.

Any traces that are a part of the equation should have the same number of points as the resulting trace.The equation editor does not check if the stimulus match, only the size of the trace.

If the equation is invalid or cannot be executed, an error message will show in the equation editor indicating such, and the trace will default to the equation being disabled.

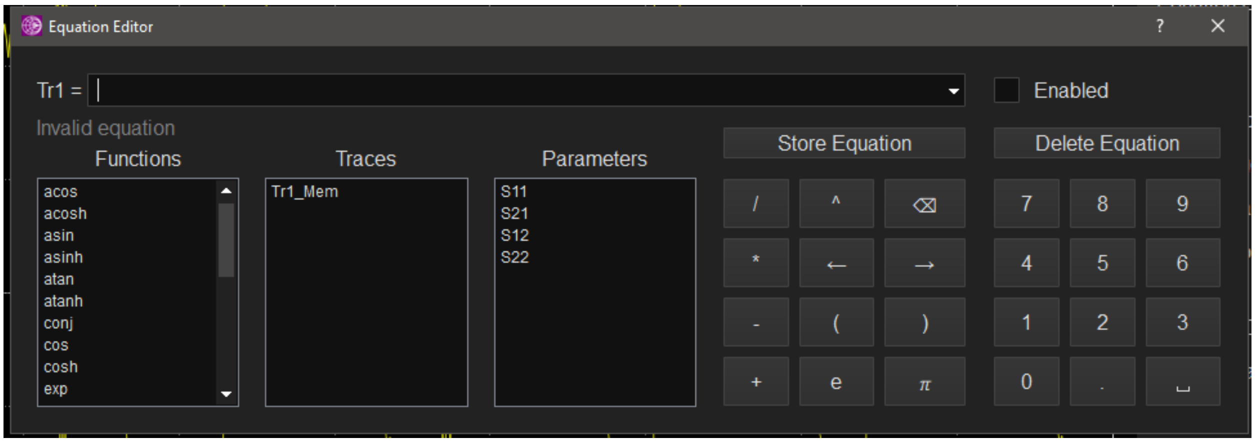


Figure 18 Equation editor dialog.

5.3.4 Trace Statistics

Statistics are made over a specified interval on a trace. By default statistics are measured over the entire frequency range of the trace. This the the behavior when “Auto Range” is selected. When auto range is unselected, the user may manually enter the frequency range for the statistics to be measured over.

Statistics are measured after the format conversion and reported in the same units as the trace.

Figure 19 Example statistics measuring flatness of a passband filter.

5.3.5 Trace Delays and Offsets

Several delays and amplitude/phase offsets can be added to a trace. These can be used to simulate or offset additional path loss or cable length from an added thru, attenuator, or similar.

These offsets are applied prior to format conversion while still complex s-parameters.

These settings are found in the Math control panel.

5.4 Limit Testing

The VNA software offers several automatic limit testing. All limit testing can be configured in the Limits control panel.

5.4.1 Limit Lines

Limit lines are fixed thresholds against which a trace is tested. Limits are defined as lines which a trace can not exceed. A pass/fail message indicates whether a trace exceeds the limit line. Each limit line is tested over a defined frequency range. Multiple ranges can be defined for each trace.

Limit lines are defined and tested in the units defined by the trace format. Changing the trace format does not change the loaded limit values, but does interpret the values in the new format.

Each trace has it's own set of limit line ranges. To test multiple traces against the same limit lines, use the Save/Load Table functionality in the limit line editor dialog.

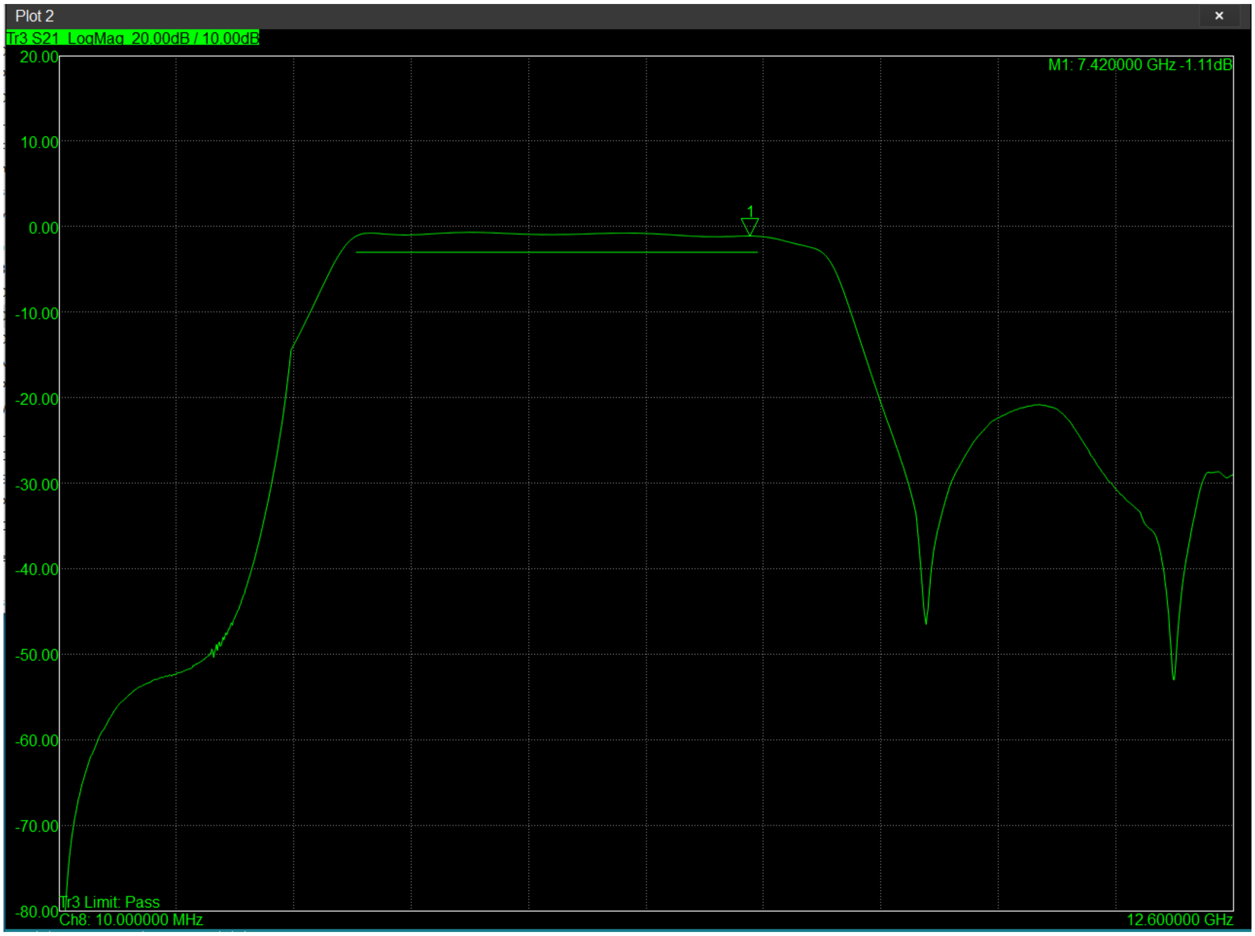


Figure 20 A limit line is used to test the maximum insertion loss of a pass band filter.

5.4.2 Ripple Tests

Ripple is defined as the delta between the largest and smallest values in a trace over a defined frequency range. Ripple tests allow you to visualize the ripple and report a pass/fail based on a defined limit on the amount of allowed ripple. Ripple tests are defined and performed in the units of the trace format.

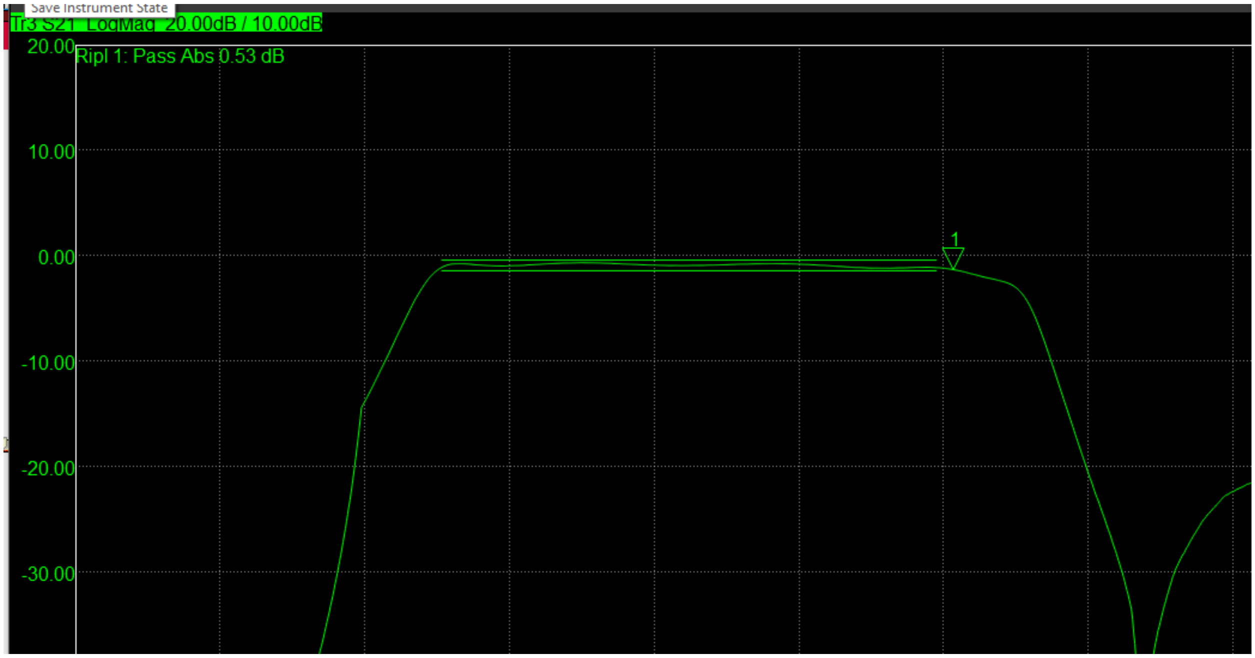


Figure 21 Ripple test measuring the ripple of the passband in a passband filter.

5.5 Time Domain

Time domain analysis allows the user to view the time domain response of their system under test. This is possible by performing an inverse FFT on the measured frequency response.

All time domain controls are found in the “Time Domain” control panel.

5.5.1 Time Resolution

The time resolution on the time domain response is determined by the span of the measurement. This controls the step size, or time per point on the time domain plot. The number of points in the measurement will then adjust the maximum time that can be viewed on the time domain plot.

5.5.2 Units

By default, the x-axis scale is time in seconds on the time domain plot. This can be changed to distance but will require a reasonable estimate of the velocity factor. The velocity factor can be manually entered. You can also define various mediums with corresponding velocity factors and recall them at a later time.

5.5.3 Scaling

By default the x-axis scale on the time domain plot displays all points, including negative times. This may not be desirable and can be manually changed. When the scale is changed

5.6 Time Gating

Time gating is a feature which allows filtering out time domain responses with the goal of isolating components in the system under test. It can also be used to remove the response of undesired components in the system digitally. Time gating controls are independent of time domain controls. Gating can be enabled while still plotting the frequency response.

Both bandpass or notch filtering can be selected as well as an adjustable window/gate.The window provides some adjustment between sidelobe rejection and sharpness on the filtering used. For example, "Maximum" gate shape optimizes for sidelobe level rejection at the expense of reduced time resolution.

All time gating controls are found in the "Time Gating" control panel.

5.6.1 Coupling

When coupling is enabled all time gate settings are copied to all other traces in the same channel when changes are made. Coupling can only be performed at the channel level. All traces in the channel will always be coupled.

5.7 DUT Length and Aliasing

If the phase shift is more than 180 degrees between measurement/frequency points, aliasing can occur. This affects several measurement,

- Phase plots will appear incorrectly
- Group delay plots will measure incorrectly.
- Interpolation of calibrations error terms will be incorrect.
- Port extensions will measure length incorrectly.
- SOLR calibrations will improperly

To ensure your frequency spacing is appropriate, your frequency spacing should be determined with the following formula.

$$f_{step} = \frac{c \; * VF}{2 * DUT_{length}}$$

Where f_step is the largest frequency step that will still not exhibit this type of aliasing. C is the speed of light in m/s. VF is the velocity factor of the DUT specified as a value between [0,1] (defaults to 1), and the DUT length is in meters. Because there is additional length inside the VNA, some additional length should be added as a margin on the DUT length. A reasonable margin to add is several inches of length.

Using the default VF = 1, this equation simplifies to

$$f_{step} = \frac{300e6}{2 * DUT_{length}(m)}$$

6 Other Topics

6.1 Connector Care

2.92 mm connectors can be worn or damaged if they are allowed to rotate during engagement or disengagement. Connectors should be clean and free of damage. Align them carefully, do not cross-thread, and rotate only the connector nut until finger tight. Then, holding the opposite connector stationary, tighten with a torque wrench, typically to 8 inch-pounds for stainless steel connectors.

6.2 Electrostatic Discharge (ESD)

ESD can damage the solid state switches and mixers inside the VNA400. This is especially important when using antennas, or other testing where the RF signal path is exposed. Observe ESD precautions when using the VNA400.

7 Programming/SCPI

This section covers the use of SCPI commands and programming to automate the VNA software. A brief description of SCPI is provided, followed by a list and description of all available SCPI commands.

7.1 SCPI Basics

SCPI (Standard Commands for Programmable Instruments) is a standard which covers the set of commands used to program various instruments. The standard covers the syntax, form, behavior, etc. of these commands to reduce development time for the user.

For the purposes of Signal Hound and the VNA software, a user can send SCPI commands to the VNA software to control and make measurements using the VNA software in an automatic manner. SCPI commands are sent to instruments over many interfaces, commonly GPIB, VXI, USB, Ethernet, etc. The VNA software accepts commands over a network socket. The VNA software will accept a single network connection in which it can receive SCPI commands and send responses.

For many users the VNA software will be running on the same PC as their SCPI program. In this scenario a network socket with a “localhost” IP address and port (which is specified in the VNA software preferences) can be used to communicate with the software.

7.1.1 Command Basics

A SCPI command is comprised of a series of keywords separated by colons. A command may be followed by a ‘?’ to represent a query, a series of parameters separated by spaces, or both.

`:SENSE:FREQUENCY:CENTER 1GHz` (Example command for setting the center frequency to 1GHz)

`:sense:frequency:center?` (Example command for querying the current center frequency)

Commands are case insensitive. Each keyword in a command can have a short and long form. Both can be used interchangeably.

`:SENSe:FREQuency:CENTer` is a command with three keywords. Each keyword has a short and long form. The short form is denoted by the uppercase characters and the long form is the full keyword including the upper and lower-case characters. For example, FREQ is the short form of FREQUENCY. When constructing a command, the short and long form can be interchanged. For example, you could construct the command as such, `:SENS:FREQUENCY:CENT` where SENSE and CENTER are sent as short form and FREQUENCY as longform.

Some commands are options and are denoted as such by the ‘[]’ characters.

`[:SENSe]:FREQuency:CENTer` is a command where the first keyword is optional. This command can be sent as `FREQ:CENT` and still be interpreted correctly.

Commands are terminated with a newline character. For example

```
:SENS:FREQ:CENT 1GHZ\n
```

The software will begin processing the commands once a newline is reached. Additionally, a newline will reset the current keyword path.

7.1.1.1 Keyword Parameters

Some command keywords may have a keyword parameter suffix. This will be designated with angle brackets “< >”. An example of this is below,

```
:SENSe<ch>:CORRection[:STATe] <bool>
```

In this example, <ch> is the keyword parameter and is an integer that represents which channel this command should be executed on. Often leaving off the keywork parameter will cause the command to fall back to a default behavior, in this case, the currently selected channel will be used. An example of using this command is below,

```
:SENSE<2>:CORRECTION ON
```

This will turn corrections on for channel 2. Note ":STATE" was left off since it is an optional keyword and does not have to be provided.

7.1.2 Multiple Commands

Multiple commands can be sent to the device at once using the semi colon character separating each command.

```
:SENS:FREQ:CENT 1GHz; :SENS:FREQ:SPAN 10MHz\n
```

This is an example of sending two commands at once. Additionally, when sending multiple commands, you don't need to repeat all keywords leading up to the final keyword for commands after the first.

```
:SENS:FREQ:CENT 1GHz; SPAN 10MHz\n
```

Here SPAN retains the `:SENS:FREQ:` keywords from the previous command. To prevent this from happening use the colon character leading the second command. For example

```
:SENS:FREQ:CENT 1GHz; :SPAN 10MHz\n
```

This is an invalid series of commands, since span is prefixed with a colon command which reset the previous keywords.

7.1.3 Parameters

There are several types of parameters that can be sent in commands. The '|' symbol represents "or", indicating only one of the parameters should be sent.

<bool>	ON | OFF | 0 | 1
Keyword (Example) MINimize | MAXimize	Character specific strings for a given command. These keywords can also have short and long form.

Numeric
<integer>
<double>
	Numeric parameters take either the form of integer or decimal values. Examples include
	`1`
	`1.23`
	`9`
	`3.14`

Frequency <freq>	These are numeric parameters with a frequency suffix. Possible frequency suffixes include

	HZ | KHZ | MHZ | GHZ The suffixes are case insensitive. If a suffix is not present, Hz is the default unit. Examples include 1kHz 20MHz 12GHz Any function that returns a frequency will return the frequency in Hz with no suffix present.
Amplitude <amplitude>	These are numeric parameters with an amplitude suffix. Possible amplitude suffixes include DBM | DBMV | DBUV | MV The suffixes are case insensitive. A suffix must be present unless indicated otherwise. Examples include -20DBM 60dbuv If a function returns an amplitude, it will return the amplitude in the current software units without a suffix.
Time <time>	Possible time suffices include S | MS | US | NS | PS The suffixes are case insensitive. Examples include 1ns 1 NS 3 0.1s If no suffix is provided, the software will interpret the value as seconds.
<filename>	File name parameters is a string of the absolute file path including the file name and recommended extension. The file name string should be enclosed in ascii quotations, which for many programming languages will require using an escape sequence to achieve. For example, in C++ the command for saving an image to a file name might look like, const char *saveFileCmd = “:SYSTEM:IMAGE:SAVE \”C:/Users/Me/Documents/My Pictures/Capture1.png\”” Note the escape sequenced quotation marks enclosing the file name.

7.1.4 Return Values

Values returned from the Spike software (as a result of sending a query command) are separated by a semi-colon if multiple query commands are sent in one string and are terminated by a newline. For example, sending

```
“CALC:MARK:MAX; X?; Y?\n”
```

results in a return string of

`"1000000;-20\n"`

The command sent performs a peak search and queries the X and Y positions of the marker. The return is the X and Y positions separated by a semicolon and terminated with a newline.

7.1.5 Special Characters

This section describes the numerous special characters that are present in the commands in this document.

Character	Description	Example
|	Vertical stroke between parameters indicates multiple choices	`FLATtop | GAUSsian` The choices are between FLATTOP or GAUSSIAN. Provide one or the other.

[ ]	Square brackets indicate an optional keyword	`:SYSTem:ERRor[:NEXT]?`
		
		Next is an optional keyword and the command could also be composed as
		
		`:SYSTem:ERRor?`

<>	Angle brackets around a parameter indicate a type and angle brackets should not be included in the user command.	`*RCL <int>`
		
		<int> is the type of parameter and an example of using this command would be
		
		`*RCL 1`
		
		Notice the angle brackets are not included.

7.2 Getting Started

See the SCPI examples found in the SDK download on the Signal Hound website. The examples use the C programming language and a common VISA library implementation.

Instrument control is performed by connecting to the VNA software on TCP/IP port 5026. On this port, a user can send and receive raw SCPI commands. It is not necessary to use a I/O library like VISA to communicate with the VNA software but it can simplify several operations. It is possible to communicate directly over the socket with socket programming. The computer that is communicating with the VNA software does not have to be the same computer running the VNA software and does not have to be a Windows platform.

It is recommended to use a VISA library if available. Several implementations of VISA exist. Commonly used ones include Keysight's I/O libraries, and NI's VISA libraries. You can also use VISA implementations that exist in other languages/environments such as MATLAB, LabVIEW, and Python.

Connecting to the socket interface using VISA looks like this

```
viOpen(rm, "TCPIP::localhost::5025::SOCKET", VI_NULL, VI_NULL, &inst);
```

Additionally, when using a VISA library, it is necessary to set the VI_ATTR_TERMCHAR_EN attribute to true. This will terminate the read operation when the termination character is received. The termination character should be set to the newline ('\n') character if it is not set by default. The code for this is below.

```
viSetAttribute(inst, VI_ATTR_TERMCHAR_EN, VI_TRUE);

viSetAttribute(inst, VI_ATTR_TERMCHAR, '\n');
```

Only one connection to the VNA software can be active at a time. The connection can be terminated by either closing the socket connection, either through the socket library you are using, the viClose function if you are using a VISA library, or by closing your application. The software will immediately begin waiting for another socket connection when the previous one is ended.

7.2.1 Programming Examples

All SCPI programming examples are located in the Signal Hound SDK on the Signal Hound website, https://signalhound.com/software/signal-hound-software-development-kit-sdk/.

7.3 Command Reference

7.3.1 Common Commands

7.3.1.1 *IDN?

Returns a string containing the serial number and firmware version of the VNA device connected in the software. An example string might look like,

```
SignalHound,VNA400A,24229010,3
```

If no device is connected, the returned string will look like,

```
SignalHound,,,
```

7.3.1.2 *OPC?

Waits for all channels that have a sweep/trigger mode of SINGLE to complete. Once all channels return to HOLD or CONTINUOUS, a "1" will be returned.

This function is crucial for triggering and waiting for a measurement to complete.

7.3.1.3 *RST

Performs an application preset. This will return the software to the default power on state. This has the same effect as pressing the Preset button on the control panel.

7.3.2 SYSTem

7.3.2.1 ERRor

The VNA software maintains a list of system errors available to the user. Errors are stored with a unique ID, name, and description. The types of issues represented in the error list are settings conflicts, SCPI issues such as invalid parameter types or instructions, file I/O errors, etc.

See the SCPI examples to see how to poll for errors.

The errors are returned in the form

```
"ID,description;error information"
```

ID is a unique integer for the error. The description is an ascii text description for the error, and error information is any additional context information for the error generated. An example error message is below.

```
"-2,Invalid Parameter;Expected frequency parameter"
```

This error indicates the SCPI parser was expecting a frequency parameter and was either unable to find it or was unable to parse it as a frequency.

Once the error queue is empty, the software will return the 'no error' error when the next system error is requested. 'No error' has an ID of 0.

7.3.2.1.1 :SYSTem:ERRor:COUNt?

Returns the number of errors in the error queue.

7.3.2.1.2 :SYSTem:ERRor[:NEXT]?

Returns the next error in the queue, and removes it from the queue. If there are no errors in the queue, an error of "0" will be returned, which represents no error.

7.3.2.1.3 :SYSTEM:ERRor:CLEAr

Remove all errors from the queue. When finished, error count will return 0.

7.3.2.2 DEVice

The device commands are used to connect and disconnect the VNA within the application. This might be needed if multiple VNAs are connected to the PC, or if in complex scenarios.

7.3.2.2.1 :SYSTem:DEVice:ACTive?

Returns whether or not a device is currently connected in the software.

7.3.2.2.2 :SYSTem:DEVice:COUNt?

Returns the number of VNAs connected to the PC.

This command is used when attempting to connect to a VNA via SCPI. This command should only be used when no device is currently connected in the software. This can be ensured by sending the disconnect command prior to this command.

7.3.2.2.3 :SYSTem:DEVice:LIST?

Returns a list of serials numbers (comma separated) of all VNA devices connected to the PC.

This command is used when attempting to connect to a VNA via SCPI. This command should only be used when no device is currently connected in the software. This can be ensured by sending the disconnect command prior to this command.

7.3.2.2.4 :SYSTem:DEVice:CURRent?

Returns the active devices serial number. If no device is connected in the software, returns 0.

7.3.2.2.5 :SYSTem:DEVice:CONnect? <int>

Connect to a VNA with a given serial number. The serial number should be provided as a parameter. Returns 0 or 1 based on the success of connecting to the VNA. A 0 is returned if a VNA is already active in the software. This command can take several seconds to complete, and may require increasing the VISA timeout.

7.3.2.2.6 :SYSTem:DEVice:DISConnect?

Disconnects any device actively connected in the software. Returns 1 when finished.

7.3.2.3 :SYSTem:COMMunicate:GTLocal

This command returns the application to local mode and closes the remote lockout dialog.

7.3.2.4 :SYSTem:IMAGe:SAVe <filename>

Saves an image of the current state of the software using the provided file name. The file name should include either the png or jpg extension.

7.3.2.5 :SYSTem:PRINt

Prints an image of the current state of the software using the default printer settings. You do not have the ability to modify the printer settings with this command, including which printer to use.

7.3.2.6 :SYSTem:CLOSe

Stops any active measurements, disconnects the VNA, and closes the application. This will also terminate the socket connection to the application.

7.3.2.7 :SYSTem:VERsion?

Returns an application version string. An example version string is,

```
"2.0.3"
```

Where the quotes are not part of the string.

7.3.2.8 :SYSTem:PREFerence:FCAL[:STATe] <bool>

Enable or disable the use of the factory cal. This command corresponds to the “Use Factory Cal” setting found in the preferences menu.

7.3.2.9 :SYSTem:PREFerence:FCAL[:STATe]?

Returns whether the software is currently using the factory cal. This command corresponds to the “Use Factory Cal” setting found in the preferences menu.

7.3.2.10 :SYSTem:PRESet[:USER]:SAVE <filename>

Saves the current measurement configuration to a file specified by the filename parameter. The filename should have the extension “.ini”. This command is equivalent to the “File → Save Instrument State” selection in the software.

7.3.2.11 :SYSTem:PRESet[:USER]:LOAD <filename>

Loads a measurement configuration from a file specified by the filename parameter. The filename should have the “.ini” extension. This command is equivalent to to the “File → Recall Instrument State” selection in the software.

7.3.3 INITiate

7.3.3.1 :INITiate<ch>:CONTinuous

Set the sweep mode of the channel specified by the <ch> keyword to continuous.

7.3.3.2 :INITiate<ch>:CONTinuous?

Returns whether the sweep mode of the channel specified by the <ch> keyword is set to continuous.

7.3.3.3 :INITiate<ch>[:IMMediate]

Triggers a sweep on the specified channel. This has the effect of setting the sweep mode to SINGLE, performing a single sweep, then going to HOLD.

7.3.4 CONFigure

7.3.4.1 :CONFigure:CHANnel:CATalog?

Returns a list of enabled channels (comma separated).

7.3.4.2 :CONFigure:CHANnel<ch>[:STATe] <bool>

Enable or disable the channel specified by the <ch> keyword parameter.

7.3.4.3 :CONFigure:CHANnel<ch>[:STATe]?

Return the on/off state of the channel specified by the <ch> keyword parameter.

7.3.5 OUTPut

7.3.5.1 :OUTPut[:STATe] <bool>

Turns the RF power from the test ports on and off.

7.3.5.2 :OUTPut[:STATe]?

Returns the current RF power state.

7.3.6 SENSe

7.3.6.1 :SENSe<ch>:AVERage:CLEar

Resets the trace averaging on the specified channel.

7.3.6.2 :SENSe<ch>:AVERage:COUNt <int>

Sets the trace averaging count.

7.3.6.3 :SENSe<ch>:AVERage:COUNt?

Returns the trace averaging count.

7.3.6.4 :SENSe<ch>:AVERage:STATe <bool>

Enables/disables trace averaging.

7.3.6.5 :SENSe<ch>:AVERage:STATe?

Returns whether trace averaging is enabled/disabled.

7.3.6.6 :SENSe<ch>:BANDwidth <freq>

Set the IF bandwidth.

7.3.6.7 :SENSe<ch>:BANDwidth?

Retrieve the IF Bandwidth.

7.3.6.8 :SENSe<ch>:BWIDth <freq>

Set the IF bandwidth.

7.3.6.9 :SENSe<ch>:BWIDth?

Retrieve the IF bandwidth.

7.3.6.10 :SENSe<ch>:FREQuency:CENTer <freq>

Set the measurement center frequency.

7.3.6.11 :SENSe<ch>:FREQuency:CENTer?

Retrieve the measurement center frequency.

7.3.6.12 :SENSe<ch>:FREQuency:SPAN <freq>

Set the span.

7.3.6.13 :SENSe<ch>:FREQuency:SPAN?

Retrieve the span.

7.3.6.14 :SENSe<ch>:FREQuency:SPAN:FULL

Set the span to the full measurement range of the instrument.

7.3.6.15 :SENSe<ch>:FREQuency:STARt <freq>

Set the measurement start frequency.

7.3.6.16 :SENSe<ch>:FREQuency:STARt?

Retrieve the measurement start frequency.

7.3.6.17 :SENSe<ch>:FREQuency:STOP <freq>

Set the measurement stop frequency.

7.3.6.18 :SENSe<ch>:FREQuency:STOP?

Retrieve the measurement stop frequency.

7.3.6.19 :SENSe<ch>:FREQuency:STEP <freq>

Set the measurement step frequency.

7.3.6.20 :SENSe<ch>:FREQuency:STEP?

Retrieve the measurement step frequency.

7.3.6.21 :SENSe:ROSCillator:SOURce INTernal | EXTernal

Set the VNA timebase reference.

7.3.6.22 :SENSe:ROSCillator:SOURce?

Retrieve the VNA timebase reference.

7.3.6.23 :SENSe<ch>:SWEep:MODE HOLD |SINGle | CONTinuous

Set the channels sweep mode, or trigger configuration.

7.3.6.24 :SENSe<ch>:SWEep:MODE?

Return the channels sweep mode.

7.3.6.25 :SENSe<ch>:SWEep:POINts <int>

Set the number of measurement points.

7.3.6.26 :SENSe<ch>:SWEep:POINts?

Retreive the number of measurement points.

7.3.6.27 :SENSe<ch>:SWEep:TYPE LINear | LOGarithmic | SEGMent

This command is currently not implemented.

7.3.6.28 :SENSe<ch>:SWEep:TYPE?

This command is currently not implemented.

7.3.6.29 CORRection

7.3.6.29.1 :SENSe<ch>:CORRection[:STATe] <bool>

Sets whether calibration is enabled on the channel specified by the <ch> keyword parameter. A calibration must also be activated on the channel. Once a calibration is active, this setting controls whether that calibration is applied to the channel.

7.3.6.29.2 :SENSe<ch>:CORRection[:STATe]?

Returns whether calibration is enabled on the channel specified by the <ch> keyword parameter.

7.3.6.29.3 :SENSe<ch>:CORRection:INTerpolation[:STATe] <bool>

Set whether calibration interpolation is enabled on the channel specified by the <ch> keyword parameter.

7.3.6.29.4 :SENSe<ch>:CORRection:INTerpolation[:STATe]?

Returns whether calibration interpolation is enabled on the channel specified by the <ch> keyword parameter.

7.3.6.29.5 :SENSe<ch>:CORRection:CSET:ACTivate <filename>

Loads the calibration set given by filename parameter on the channel specified by the <ch> keyword parameter. When loading a calibration set via SCPI, the stimulus of that calibration set is automatically loaded. This is in contrast to manually loading a calibration set, in which the user is asked whether they wish to load the stimulus used in the calibration set.

7.3.6.29.6 :SENSe<ch>:CORRection:CSET:DEACtivate

Unloads the calibration on the channel specified by the <ch> keyword parameter.

7.3.6.29.7 :SENSe<ch>:CORRection:CSET:ACTive?

Returns whether a correction is activated on the selected channel.

See also :SENSe<ch>:CORRection[:STATe].

7.3.6.29.8 :SENSe<ch>:CORRection:CSET:NAME?

Returns the name of the calibration applied to the specified channel. If no calibration is loaded, returns the string “empty”.

7.3.6.29.9 :SENSe<ch>:CORRection:CSET:DATE?

If a calibration is applied to the specified channel, return the date the calibration was performed as a string, otherwise return the string “empty”. An example string might look like,

```
"Tue Jan 7 11:24:58 2025"
```

7.3.6.29.10 :SENSe<ch>:CORRection:CSET:TYPE?

Returns the type of calibration currently applied on the active channel.

The following strings can be returned

- “none” (if no calibration is active on the specified channel)
- “sol”
- “solt”
- “solr”

7.3.6.29.11 :SENSe<ch>:CORRection:CSET:ETERm? <eterm_name>

Request the error terms from the active calibration on the specified channel.

The error term is returned as a complex array that matches the sweep length as set in the specified channel's stimulus. This means, if the sweep length is currently set to 1601, then 3201 comma separated values will be returned, representing the complex error term.

If there is no active calibration, this command returns nothing, and an error will be generated.

The error term name is a string and should be one of the following values,

- "edf" – Forward directivity
- "esf" – Forward source match
- "erf" – Forward reflection tracking
- "elf" – Forward load match
- "etf" – Forward transmission tracking
- "exf" – Forward isolation
- "edr" – Reverse directivity
- "esr" – Reverse source match
- "err" – Reverse reflection tracking
- "elr" – Reverse load match
- "etr" – Reverse transmission tracking
- "exr" – Reverse isolation

The returned array has the following format,
<pt1.re>, <pt1.im>, <pt2.re>, <pt2.im>, … <ptN.re>, <ptN.im>
Where each real/imaginary value is a floating point value.

7.3.6.29.12 :SENSe:CORRection:CKIT:CATalog?

TODO

7.3.6.29.13 :SENSe:CORRection:CKIT:COUNt?

7.3.6.30 CORRection:COLLect

These commands are used to perform a "remote calibration".

Comment: A simpler approach to performing a remote calibration (when possible), is to manually perform a calibration, and then simply load the calibration file using the CORRection:CSET:ACTivate command.

Only one remote calibration can be performed at a time. If multiple channels require calibration, they must be performed sequentially. Once a remote calibration has been started, it must be finished. It is not possible to save a partial calibration.

A remote calibration will follow this procedure.

- Configure your channel stimulus

- Use he CKIT:SELect command to choose which cal kit to use.
- Use the METHod command to choose the cal type and start the calibration.
- The the ACQuire commands to measure the required standards.
    - Alternatively, the user can manually perform the standard measurements using the dialog that appears after the METHod command is sent.
- The SAVE command then finished the calibration.

If the calibration dialog is closed prior to measuring all the standards and sending the SAVE command (by manually closing it in the application for example), the calibration progress is lost and must be restarted.

7.3.6.30.1 :SENSe<ch>:CORRection:COLLect:CKIT:SELect <ckit_name>

This command sets the cal kit to be used for the remote calibration. Ckit_name must be one of the calibration kit names listed in the CKIT:CATalog? command.

<ch> is ignored.

7.3.6.30.2 :SENSe<ch>:CORRection:COLLect:CKIT:SELect?

Query the cal kit name currently configured for use in the remote calibration.

<ch> is ignored.

7.3.6.30.3 :SENSe<ch>:CORRection:COLLect:METHod SOL | SOLT | SOLR

Start the calibration using the specified method. If the parameter does not match one of the desired methods, this command will fail with no effect.

The calibration is started using the stimulus of the specified channel.

If a remote calibration is already in progress, this command will fail with no effect, and the existing calibration in progress will be unaffected.

If the specified channel is not active, then this command will fail with no effect.

If this command succeeds, a calibration dialog will appear in the application.

7.3.6.30.4 :SENSe<ch>:CORRection:COLLect[:ACQuire]:SHORt? <port>,<stdID>

Measure a short for a remote calibration on the specified port. Specify which cal kit standard used with the stdID parameter. This ID should match the ID found in cal kit editor for that standard.

This function returns 1 for success and 0 for failure. This command can fail if a remote calibration is not currently in progress, or the ID provided does not match the correct standard type.

<ch> is ignored.

<port> should be 1 or 2. If the remote calibration is using the SOL method, port is ignored, but must still be provided.

7.3.6.30.5 :SENSe<ch>:CORRection:COLLect[:ACQuire]:OPEN? <port>,<stdID>

Measure an open for a remote calibration on the specified port. Specify which cal kit standard used with the stdID parameter. This ID should match the ID found in cal kit editor for that standard.

This function returns 1 for success and 0 for failure. This command can fail if a remote calibration is not currently in progress, or the ID provided does not match the correct standard type.

<ch> is ignored.

<port> should be 1 or 2. If the remote calibration is using the SOL method, port is ignored, but must still be provided.

7.3.6.30.6 :SENSe<ch>:CORRection:COLLect[:ACQuire]:LOAD? <port>,<stdID>

Measure a load for a remote calibration on the specified port. Specify which cal kit standard used with the stdID parameter. This ID should match the ID found in cal kit editor for that standard.

This function returns 1 for success and 0 for failure. This command can fail if a remote calibration is not currently in progress, or the ID provided does not match the correct standard type.

<ch> is ignored.

<port> should be 1 or 2. If the remote calibration is using the SOL method, port is ignored, but must still be provided.

7.3.6.30.7 :SENSe<ch>:CORRection:COLLect[:ACQuire]:THRU? <stdID>

Measure a thru for a remote calibration. Specify which cal kit standard used with the stdID parameter. This ID should match the ID found in cal kit editor for that standard.

This function returns 1 for success and 0 for failure. This command can fail if a remote calibration is not currently in progress, or the ID provided does not match the correct standard type.

<ch> is ignored.

7.3.6.30.8 :SENSe<ch>:CORRection:COLLect[:ACQuire]:ISO?

Measure the isolation for a remote calibration.

This function returns 1 for success and 0 for failure. This command can fail if a remote calibration is not currently in progress.

<ch> is ignored.

7.3.6.30.9 :SENSe<ch>:CORRection:COLLect:SAVE <filename>

This command finished the remote calibration, saves the calibration file, applies the calibration to the specified channel, and enables it.

<filename> should be a fully specified path with file name containing extension .vnacal. The calibration will be saves to this location. If no filename is provided a default path of

C:/Users/Username/Documents/SignalHound/vna/RemoteCal.vnacal

Will be used.

If the remote calibration is not complete, this command will fail with no effect. A calibration is not complete until all standards have been measured across the full frequency range of the stimu

7.3.6.31 CORRection:DEEMBed

7.3.6.31.1 :SENSe<ch>:CORRection:DEEMBed[:STATe] <bool>

Enable deembedding for the specified channel.

7.3.6.31.2 :SENSe<ch>:CORRection:DEEMBed[:STATe]?

Returns whether deembedding is enabled for the specified channel.

7.3.6.31.3 SENSe<ch>:CORRection:DEEMBed:PORT<port>:LOAD <filename>

Load a .s2p file to be used for deembedding for the specified port on the specified channel.

7.3.6.31.4 SENSe<ch>:CORRection:DEEMBed:PORT<port>:LOAD?

Returns whether an .s2p file is loaded for deembedding on the specified port on the specified channel.

7.3.6.31.5 SENSe<ch>:CORRection:DEEMBed:PORT<port>:CLEar

Clears the deembedding .s2p file on the specified port on the specified channel.

7.3.6.32 CORRection:EMBed

7.3.6.32.1 SENSe<ch>:CORRection:EMBed[:STATe] <bool>

Enable embedding for the specified channel.

7.3.6.32.2 :SENSe<ch>:CORRection:EMBed[:STATe]?

Returns whether embedding is enabled for the specified channel.

7.3.6.32.3 SENSe<ch>:CORRection:EMBed:PORT<port>:LOAD <filename>

Load an .s2p file to be used for embedding for the specified port on the sapecified channel.

7.3.6.32.4 SENSe<ch>:CORRection:EMBed:PORT<port>:LOAD?

Returns whether a .s2p file is loaded for embedding on the specified port on the specified channel.

7.3.6.32.5 SENSe<ch>:CORRection:EMBed:PORT<port>:CLEar

Clears the embedding .s2p file on the specified port on the specified channel.

7.3.7 SOURce

7.3.7.1 :SOURce<ch>:POWer:MODE HIGH | LOW | ATTen | POWer

Set the power output mode on the specified channel. See Power Level for more information.

7.3.7.2 :SOURce<ch>:POWer:MODE?

Return the power output mode for the specified channel.

7.3.7.3 :SOURce<ch>:POWer:ATTenuation <double>

Set the manual attenuation for the specified channel. The value provided is interpreted as dB. See Power Level for more information.

7.3.7.4 :SOURce<ch>:POWer:ATTenuation?

Return the manual attenuation for the specified channel.

7.3.7.5 :SOURce<ch>[:LEVel][:IMMediate][:AMPLitude] <double>

Set the manual output power for the specified channel. The value provided is interpreted as dBm. See Power Level for more information.

7.3.7.6 :SOURce<ch>[:LEVel][:IMMediate][:AMPLitude]?

Return the manual output power for the specified channel.

7.3.8 CALCulate:MEASure

7.3.8.1 :CALCulate<ch>:MEASure:CATalog?

Returns a list of traces (as integers) that are assigned to the channel. The list is returned as CSV.

7.3.8.2 :CALCulate<ch>:MEASure<tr>:CORRection:EDELay[:TIME] <time>

Specify the electrical delay

7.3.8.3 :CALCulate<ch>:MEASure<tr>:CORRection:EDELay[:TIME]?

Returns the current electrical delay as seconds.

7.3.8.4 :CALCulate<ch>:MEASure<tr>:DATA:X?

Returns the x axis values for the trace. For an s-parameter sweep these are the frequencies of each point in the sweep. For a time domain trace, these are the times.

A CSV list is returned and is equal to the length of the sweep.

7.3.8.5 :CALCulate<ch>:MEASure<tr>:DATA:FDATA?

Returns the formatted measurement data for the specified trace. The trace is returned as a CSV, equal to the length of the sweep.

When a calibration is enabled, the data is corrected, otherwise the data returned is uncorrected.

If the data format is in the Smith chart format, 2 values are returned per data point, otherwise 1 value is returned per data point.

See the data processing map for more information.

7.3.8.6 :CALCulate<ch>:MEASure<tr>:DATA:SDATA?

Returns the complex measurement data for the specified trace. The trace is returned as a CSV, with 2 values per data point.

When a calibration is enabled, the data is corrected, otherwise the data returned is uncorrected.

See the data processing map for more information.

7.3.8.7 :CALCulate<ch>:MEASure<tr>:DATA:RDATA?

Returns the raw s-parameters associated with the specified trace and channel. The trace is returned as a complex array, where the number of complex values returned is equal to the current stimulus sweep size.

See the data processing map for more information.

7.3.8.8 :CALCulate<ch>:MEASure<tr>:DATA:RCVR? <recv_name>

Returns an unratioed receiver array. The data is returned as a complex array, where the number of complex values returned is equal to the current stimulus sweep size.

<ch> specifies the target channel.

<tr> is ignored. Any of the receiver arrays can be queried regardless of which traces are active in the software. Please note, that the traces configured in the software determine which ports are stimulated. For instance, a trace measuring S12 or S22 will require port 2 to be stimulated. One of these traces must be configured for the reverse receiver data to be populated. If the port is not stimulated during the sweep, the receiver arrays associated with the port will be populated typically with a sentinel value of “-200”.

<recv_name> is a string specifying which receiver array is returned. Valid values are,

- “a1fwd”
- “a2fwd”
- “b1fwd”
- “b2fwd”
- “a1rev”
- “a2rev”
- “b1rev”
- “b2rev”

Where [a|b][1|2] refer to one of the ports in the picture below, and [fwd|rev] refer to whether port 1 or port 2 is being stimulated. All four receivers are available for both ports.

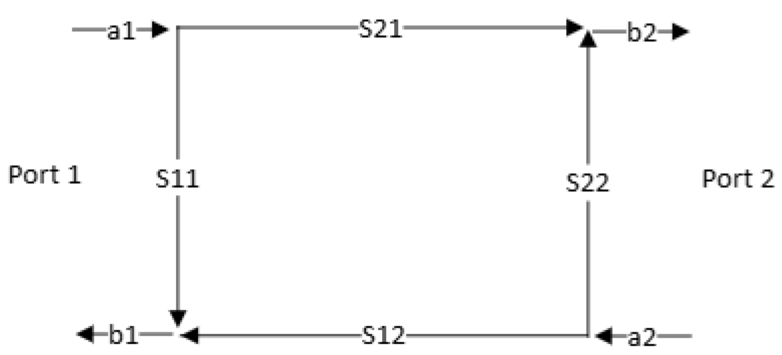


Figure 22: Diagram of receiver names.

See the data processing map for more information.

7.3.8.9 :CALCulate<ch>:MEASure<tr>:DEFine

Creates a new trace on the specified channel with the specified trace number. If there is already an active trace with that number, then this command does not do anything and a system error is logged.

7.3.8.10 :CALCulate<ch>:MEASure<tr>:DELete

Deletes the specified trace and removes it from the application. This trace will no longer be able to be accessed and will need to be defined again before being used. Any measurement data associated with this trace will be lost.

7.3.8.11 :CALCulate<ch>:MEASure<tr>:DELete:ALL

This command deletes all traces in the application.

The channel and trace keyword parameters are ignored.

7.3.8.12 :CALCulate<ch>:MEASure<tr>:DISPlay <int>

Moves the specified trace to the plot specified by the command parameter. A list of active plots can be retrieved with the :DISPlay:CATalog? command.

7.3.8.13 :CALCulate<ch>:MEASure<tr>:DISPlay?

Returns the plot of the specified trace.

7.3.8.14 :CALCulate<ch>:MEASure<tr>:PARameter S11 | S21 | S12 | S22

Set the trace measurement parameter.

7.3.8.15 :CALCulate<ch>:MEASure<tr>:PARameter?

Retrieves the trace measurement parameter.

7.3.8.16 :CALCulate<ch>:MEASure<tr>:FORMat MLINear | MLOGarithmic | RPHase | DPHase | UPHase | SWR | SMITh | SADMittance | REAL | IMAGinary | GDELay

Set the trace format.

7.3.8.17 :CALCulate<ch>:MEASure<tr>:FORMat?

Retreive the trace format.

7.3.8.18 :CALCulate<ch>:MEASure<tr>:HOLD:CLEar

Clears the held trace. If “Trace Hold” is Off, this command has no effect.

Equivalent to pressing “Restart Hold”.

7.3.8.19 :CALCulate<ch>:MEASure<tr>:HOLD:TYPE OFF | MINimum | MAXimum

Sets the trace hold setting.

7.3.8.20 :CALCulate<ch>:MEASure<tr>:HOLD:TYPE?

Retrieves the trace hold setting.

7.3.8.21 Y[:SCALe]

7.3.8.21.1 :CALCulate<ch>:MEASure<tr>:Y[:SCALe]:AUTO

Auto scale the specified trace.

Equivalent to pressing “Auto Scale” on the toolbar or trace control panel.

7.3.8.21.2 :CALCulate<ch>:MEASure<tr>:Y[:SCALe]:AUTO:ALL

Auto scales all traces. <ch> and <tr> are ignored.

Equivalent to pressing “Auto Scale All” on the toolbar or trace control panel.

7.3.8.21.3 :CALCulate<ch>:MEASure<tr>:Y[:SCALe]:DEFault

Returns the specified trace to the default scale.

Equivalent to pressing "Default Scale" on the trace control panel.

7.3.8.21.4 :CALCulate<ch>:MEASure<tr>:Y[:SCALe]:DEFault:ALL

Returns all traces to the default scale. <ch> and <tr> are ignored.

Equivalent to pressing "Default Scale All" on the trace control panel.

7.3.8.21.5 :CALCulate<ch>:MEASure<tr>:Y[:SCALe]:COUPle:METHod OFF | PLOT | ALL

Set the sytem scale coupling. <ch> and <tr> are ignored.

7.3.8.21.6 :CALCulate<ch>:MEASure<tr>:Y[:SCALe]:COUPle:METHod?

Retrieve the current scale coupling. <ch> and <tr> are ignored.a

7.3.8.22 MATH

These commands can be used to modify the settings in the math control panel.

7.3.8.22.1 :CALCulate<ch>:MEASure<tr>:MATH:MEMorize

Stores the memory for the specified trace.

Equivalent to pressing "Data → Memory" in the user interface.

7.3.8.22.2 :CALCulate<ch>:MEASure<tr>:MATH:FUNCtion NORMal | ADD | SUBTract | MULTiply | DIVide | DATA

Set the math memory operation.

7.3.8.22.3 :CALCulate<ch>:MEASure<tr>:MATH:FUNCtion?

Retrieve the math memory operation.

7.3.8.22.4 :CALCulate<ch>:MEASure<tr>:MATH:INTerpolate[:STATe] <bool>

Enable/disable math memory interpolation.

7.3.8.22.5 :CALCulate<ch>:MEASure<tr>:MATH:INTerpolate[:STATe]?

Retrieve the math memory interpolation state.

7.3.8.23 OFFSet

7.3.8.23.1 :CALCulate<ch>:MEASure<tr>:OFFSet:MAGNitude <double>

Sets the trace magnitude offset in dB.

7.3.8.23.2 :CALCulate<ch>:MEASure<tr>:OFFSet:MAGNitude?

Retrieve the trace's magnitude offset. Returned as a double in dB units.

7.3.8.23.3 :CALCulate<ch>:MEASure<tr>:OFFSet:MAGNitude:SLOPe <double>

Set the trace offset slope in dB/GHz.

7.3.8.23.4 :CALCulate<ch>:MEASure<tr>:OFFSet:MAGNitude:SLOPe?

Retrieves the trace offset slope. Returned as a double in dB/GHz units.

7.3.8.23.5 :CALCulate<ch>:MEASure<tr>:OFFSet:PHASe <double>

Set the trace phase offset in degrees.

7.3.8.23.6 :CALCulate<ch>:MEASure<tr>:OFFSet:PHASe?

Returns the trace phase offset in degrees.

7.3.8.24 SMOothing

7.3.8.24.1 :CALCulate<ch>:MEASure<tr>:SMOothing[:STATe] <bool>

Enable/disable trace smoothing.

7.3.8.24.2 :CALCulate<ch>:MEASure<tr>:SMOothing[:STATe]?

Return whether trace smoothing is enabled/disabled.

7.3.8.24.3 :CALCulate<ch>:MEASure<tr>:SMOothing:APERture <double>

Set the trace smoothing aperture as a percentage.

7.3.8.24.4 :CALCulate<ch>:MEASure<tr>:SMOothing:APERture?

Returns the trace smoothing aperture as a percentage.

7.3.8.25 STATistics

7.3.8.25.1 :CALCulate<ch>:MEASure<tr>:STATistics[:STATe] <bool>

Enable/disable trace statistics.

7.3.8.25.2 :CALCulate<ch>:MEASure<tr>:STATistics[:STATe]?

Retrieve whether trace statistics are enabled/disabled.

7.3.8.25.3 :CALCulate<ch>:MEASure<tr>:STATistics:AUTO[:STATe] <bool>

Set the auto range property of the trace statistics.

7.3.8.25.4 :CALCulate<ch>:MEASure<tr>:STATistics:AUTO[:STATe]?

Retrieve the auto range state of the trace statistics.

7.3.8.25.5 :CALCulate<ch>:MEASure<tr>:STATistics:STARt <freq>

Set the trace statistics range start frequency.

7.3.8.25.6 :CALCulate<ch>:MEASure<tr>:STATistics:STARt?

Retrieve the trace statistics range start frequency.

7.3.8.25.7 :CALCulate<ch>:MEASure<tr>:STATistics:STOP <freq>

Set the trace statistics range stop frequency.

7.3.8.25.8 :CALCulate<ch>:MEASure<tr>:STATistics:STOP?

Retrieve the trace statistics range stop frequency.

7.3.8.25.9 :CALCulate<ch>:MEASure<tr>:STATistics:DATA? MIN | MAX | MEAN |STDEV

Retrieve the trace statistics value for the associated parameter (min/max/mean/stddev). If no parameter is provided, no response is returned.

Only one value can be retrieved per command.

7.3.8.26 MARKer

7.3.8.26.1 :CALCulate<ch>:MEASure<tr>:MARKer:AOFF

Disables all markers on the specified trace.

7.3.8.26.2 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth[:STATe] <bool>

Enable/disable the bandwidth marker measurement.

7.3.8.26.3 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth[:STATe]?

Return whether the bandwidth marker measurement is enabled/disabled.

7.3.8.26.4 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth:THReshold <double>

Set the bandwidth marker measurement threshold. The value is provided in dB units.

7.3.8.26.5 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth:THReshold?

Returns the bandwidth marker measurement threshold in dB.

7.3.8.26.6 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth:TRACking <bool>

Enables/diables bandwidth marker tracking.

7.3.8.26.7 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth:TRACking?

Returns whether bandwidth marker tracking is enabled.

7.3.8.26.8 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:BWIDth:DATA?

Returns a list of measurement information as comma separated values for the bandwidth measurement. The response is of the form,

Bandwidth measurement result (0 or 1),
Lower frequency of measurement (in Hz),
Upper frequency of measurement (in Hz),
Loss (in dB)

The response of an enabled and successful bandwidth measurement might look like,

`“1,2300963610.217998,3673369707.3166747,-16.16”`

If the bandwidth measurement is disabled the following response will be returned,

`“0,0,0,0”`

7.3.8.26.9 :CALCulate:MEASure:MARKer:COUPling OFF | CHANnel | ALL

Sets the application wide marker coupling.

7.3.8.26.10 :CALCulate:MEASure:MARKer:COUPling?

Returns the marker coupling setting.

7.3.8.26.11 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>DELTa <bool>

Enable/disable the marker as a delta marker.

7.3.8.26.12 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:DELTa?

Returns whether the marker is a delta marker.

7.3.8.26.13 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:EXECute MAXimum | MINimum | RPEak | LPEak | NPEak

Executes a peak search based on the provided parameter. This command is equivalent to pressing the Peak Search, Min Peak, Peak Right, Peak Left, and Next Peak buttons.

7.3.8.26.14 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:PEAK:EXCursion <double>

Set the marker excursion value. No units should be provided. The value is in the units that match the trace format.

7.3.8.26.15 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:PEAK:EXCursion?

Return the marker excursion value. The units match the trace format.

7.3.8.26.16 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:PEAK:THReshold <double>

Set the marker threshold. No units should be provided. The value is in the units that match the trace format.

7.3.8.26.17 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:PEAK:THReshold ?

Return the marker threshold value. The units match the trace format.

7.3.8.26.18 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:TRACking OFF | MAXimum

Set the marker tracking function. This command is equivalent to setting the marker tracking to Off or Max.

7.3.8.26.19 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:FUNCtion:TRACking?

Return the marker tracking function.

7.3.8.26.20 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence[:STATe] <bool>

Enables/disables the reference marker.

7.3.8.26.21 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence[:STATe]?

Return whether the reference marker is enabled/disabled.

7.3.8.26.22 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence:TYPE NORMal | FIXed

Set whether the reference marker is fixed or updating (normal).

7.3.8.26.23 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence:TYPE?

Returns whether the reference marker is fixed or updating (normal).

7.3.8.26.24 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence:X <freq>

Set the x position of the reference marker.

7.3.8.26.25 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence:X?

Returns the x position of the reference marker. The value will be in either Hz or seconds depending on whether time domain is enabled.

7.3.8.26.26 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:REFerence:Y?

Returns the y position of the reference marker. The units of the value will depend on the trace format.

7.3.8.26.27 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>[:STATe] <bool>

Enables/disables the marker.

7.3.8.26.28 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>[:STATe]?

Returns whether the marker is enabled/disabled.

7.3.8.26.29 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:TYPE

Sets whether the marker is fixed or updating.

7.3.8.26.30 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:TYPE?

Returns whether the marker is fixed or updating (normal).

7.3.8.26.31 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:X <freq>

Set the x position of the marker.

7.3.8.26.32 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:X?

Returns the x position of the marker.

7.3.8.26.33 :CALCulate<ch>:MEASure<tr>:MARKer<mkr>:Y?

Returns the y position of the marker. The units of the value will depend on the trace format.

7.3.8.27 LIMit

These commands control the standard limit lines.

7.3.8.27.1 :CALCulate<ch>:MEASure<tr>:LIMit[:STATe] <bool>

Enable/disable the limit line test on the specified trace.

7.3.8.27.2 :CALCulate<ch>:MEASure<tr>:LIMit[:STATe]?

Retrieve whether the limit line test on the specified trace is enabled/disabled.

7.3.8.27.3 :CALCulate<ch>:MEASure<tr>:LIMit:DISPlay[:STATe] <bool>

Show/hide the limit line on the specified trace.

7.3.8.27.4 :CALCulate<ch>:MEASure<tr>:LIMit:DISPlay[:STATe]?

Retrieve whether the limit lines are shown/hidden on the specified trace.

7.3.8.27.5 :CALCulate<ch>:MEASure<tr>:LIMit:DATA

Used to add new ranges to the limit line test on the specified trace. Values are provided separated by commas. 5 values must be provided for each limit line range to insert. The number of ranges to insert is implied by the number of parameters sent. If the number of parameters provides is not a multiple of 5, the command will fail and no change will be applied to the software.

The 5 values should be provided for each range you want to set, in order.

- The type of range, 0 = Off, 1 = Min, 2 = Max
- Start frequency in Hz (no units)
- Stop frequency in Hz (no units)
- Start offset (no units)
- Stop offset (no units)

An example command that adds one new max limit range from 1 to 2 GHz with an offset of +10 looks like,

```
:CALC1:MEAS1:LIMIT:DATA 2,1000000000,2000000000,10,10
```

Any existing ranges are deleted and replaced with the limit line ranges set by this function.

7.3.8.27.6 :CALCulate<ch>:MEASure<tr>:LIMit:DATA:DELete

Clears all existing limit line ranges on the specified trace.

7.3.8.27.7 :CALCulate<ch>:MEASure<tr>:LIMit:FAIL?

Returns whether the limit line test for the specified trace passed or failed. If the limit line test is disabled, returns true.

7.3.8.28 RLIMit

These commands control the ripple limit lines.

7.3.8.28.1 :CALCulate<ch>:MEASure<tr>:RLIMit[:STATe] <bool>

Enable/disable the ripple limit line test on the specified trace.

7.3.8.28.2 :CALCulate<ch>:MEASure<tr>:RLIMit[:STATe]?

Retrieve whether the ripple limit line test on the specified trace is enabled/disabled.

7.3.8.28.3 :CALCulate<ch>:MEASure<tr>:RLIMit:DISPlay[:STATe] <bool>

Show/hide the ripple limit line on the specified trace.

7.3.8.28.4 :CALCulate<ch>:MEASure<tr>:RLIMit:DISPlay[:STATe]?

Retrieve whether the ripple limit lines are shown/hidden on the specified trace.

7.3.8.28.5 :CALCulate<ch>:MEASure<tr>:RLIMit:DATA

Used to add new ranges to the ripple limit line test on the specified trace. Values are provided separated by commas. 4 values must be provided for each ripple limit line range to insert. The number of ranges to insert is implied by the number of parameters sent. If the number of parameters provides is not a multiple of 4, the command will fail and no change will be applied to the software.

The 4 values should be provided for each range you want to set, in order.

- The type of limit, 0 = OFF, 1 = Absolute, 2 = Margin
- Start frequency in Hz (no units)
- Stop frequency in Hz (no units)
- Ripple limit in dB (no units)

An example command that adds one new max limit range from 1 to 2 GHz with an limt of 2dB looks like,

```
:CALC1:MEAS1:RLIMIT:DATA 2,1000000000,2000000000,2
```

Any existing ranges are deleted and replaced with the limit line ranges set by this function.

7.3.8.28.6 :CALCulate<ch>:MEASure<tr>:RLIMit:DATA:DELete

Clears all existing ripple limit line ranges on the specified trace.

7.3.8.28.7 :CALCulate<ch>:MEASure<tr>:RLIMit:FAIL?

Returns whether the ripple limit line test for the specified trace passed or failed. If the ripple limit line test is disabled, returns true.

7.3.8.29 TRANsform

7.3.8.29.1 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:STATe <bool>

Enables/disables time domain transform for the specified trace.

7.3.8.29.2 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:STATe?

Returns whether time domain transformation is enabled/disabled for the specified trace.

7.3.8.29.3 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME[:TYPE] BPASs | LPSTep | LPIMulse

Sets the time domain transform type.

7.3.8.29.4 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME[:TYPE]?

Retrieves the time domain transform type.

7.3.8.29.5 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:WINDow[:TYPE] MINimum | NORMal | MAXimum | CUSTom

Sets the time domain window filter coefficient.

7.3.8.29.6 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:WINDow[:TYPE]?

Retrieves the time domain window filter coefficient.

7.3.8.29.7 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:KBESsel <double>

Sets the custom time domain window filter coefficient.

7.3.8.29.8 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:KBESsel?

Retreives the custom time domain window filter coefficient.

7.3.8.29.9 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SLPFrequency

Sets the stimulus to be usable in low pass time domain modes. This has a similar effect to pressing the “Set Low Pass Frequencies” button in the time domain control panel.

Applies these changes immediately to the stimulus of the specified trace. Can change the start freq, stop freq, and number of points to accomplish this.

7.3.8.29.10 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SLPFrequency?

Query whether the stimulus for the specified trace is configured properly for low pas time domain measurements.

7.3.8.29.11 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:MARKer:MODE AUTO | REFLection | TRANsmission

Set the time domain marker mode.

7.3.8.29.12 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:MARKer:MODE?

Retrieve the time domain marker mode.

7.3.8.29.13 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:MARKer:UNIT TIME | METRS | FEET | INCHes

Set the time domain marker units.

7.3.8.29.14 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:MARKer:UNIT?

Retrieves the time domain marker units.

7.3.8.29.15 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:MARKer:VELFactor <double>

Sets the time domain velocity factor.

7.3.8.29.16 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:MARKer:VELFactor?

Retrieves the time domain velocity factor.

7.3.8.29.17 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:AUTO:STATe <bool>

Enables/disables time domain auto scaling.

7.3.8.29.18 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:AUTO:STATe?

Retrieves whether time domain auto scaling is enabled/disabled.

7.3.8.29.19 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:STARt <time>

Sets the time domain scale start time.

7.3.8.29.20 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:STARt?

Retrieves the time domain scale start time.

7.3.8.29.21 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:STOP <time>

Sets the time domain scale stop time.

7.3.8.29.22 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:STOP?

Retrieves the time domain scale stop time.

7.3.8.29.23 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:CENTer <time>

Sets the time domain scale center.

7.3.8.29.24 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:CENTer?

Retrieves the time domain scale center.

7.3.8.29.25 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:SPAN <time>

Sets the time domain scale span.

7.3.8.29.26 :CALCulate<ch>:MEASure<tr>:TRANsform:TIME:SCALe:SPAN?

Retreives the time domain scale span.

7.3.8.30 FILTer:GATE

7.3.8.30.1 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:STATe <bool>

Enable/disable time gating.

7.3.8.30.2 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:STATe?

Retrieves whether time gating is enabled/disabled.

7.3.8.30.3 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:STARt <time>

Sets the time gate start time.

7.3.8.30.4 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:STARt?

Retrieves the time gate start time.

7.3.8.30.5 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:STOP <time>

Sets the time gate stop time.

7.3.8.30.6 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:STOP?

Retrieves the time gate stop time.

7.3.8.30.7 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:CENTer <time>

Sets the time gate center.

7.3.8.30.8 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:CENTer?

Retrieves the time gate center.

7.3.8.30.9 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:SPAN <time>

Sets the time gate span.

7.3.8.30.10 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:SPAN?

Retrieves the time gate span.

7.3.8.30.11 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME[:TYPE] BPASs | NOTCh

Sets the time gate type.

7.3.8.30.12 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME[:TYPE]?

Retrieves the time gate type.

7.3.8.30.13 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:SHAPe MINimum | NORMal | MAXimum | WIDE

Sets the time gate filter shape.

7.3.8.30.14 :CALCulate<ch>:MEASure<tr>:FILTer[:GATE]:TIME:SHAPe?

Retrieves the time gate filter shape.

7.3.9 DISPlay

These commands control the plots visible in the application and the overall interface. The <win> keyword parameter should be a value between 1 and 32.

7.3.9.1 :DISPlay:MINimize

Minimizes the application to the taskbar.

7.3.9.2 :DISPlay:MAXimize

Maximizes the application.

7.3.9.3 :DISPlay:CATalog?

Returns a list of all active plots in the application. The list is returned as a CSV. If no plots are enabled in the application, this command returns "-1".

7.3.9.4 :DISPlay:WINDow<win>[:STATe] <bool>

Enables or disables the specified plot. Returns with no effect if attempting to enable a plot that is already active, or disable a plot that is not currently active.

Disabling a plot will delete all traces currently assigned to this plot.

When a new plot is activated, no traces will be assigned to this plot.

A list of all currently active plots can be acquired with the :DISPlay:CATalog? Command.

7.3.9.5 :DISPlay:WINDow<win>[:STATe]?

Returns whether the specified plot is active.

7.3.9.6 :DISPlay:WINDow<win>:TITLe:DATA <string>

Set the title of the specified plot. If the specified plot is not active, there is no effect.

7.3.9.7 :DISPlay:WINDow<win>:TITLe:DATA?

Returns the title of the specified plot. If no title is present on the specified plot or the specified plot is not active, no response is returned.

7.4 SCPI Recorder

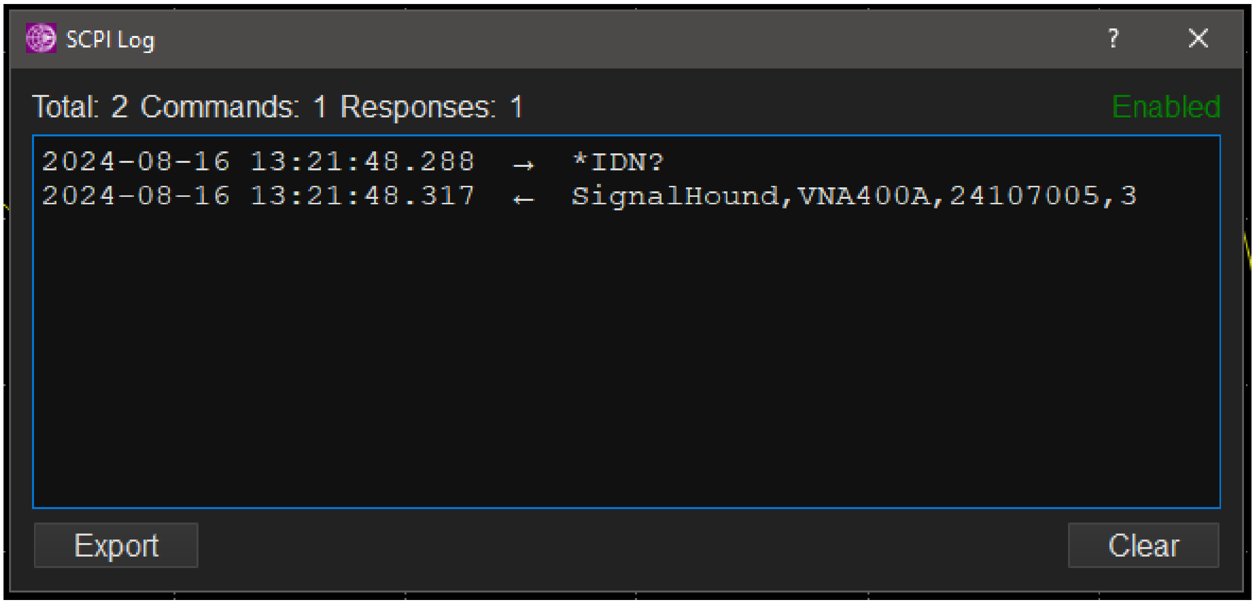
When logging is enabled, SCPI commands received, and responses sent by the software are logged with timestamps. The log can be viewed in a non-modal dialog via the Utilities -> SCPI Log menu item.

In the upper left of the dialog, running counters show the number of commands received, responses sent, and total messages handled.

The log's active state is shown in the upper right of the dialog. It can be enabled or disabled in the SCPI preferences.

The log can be exported as a plain text file using the "Export" button.

The log can be cleared using the "Clear" button.



8 Troubleshooting

For troubleshooting and sales inquiries, please contact support@signalhound.com.

9 Warranty and Disclaimer

The information contained in this manual is subject to change without notice. Signal Hound makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties or merchantability and fitness for a particular purpose.

Signal Hound shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Refer to the End User License Agreement for additional warranty and disclaimer information covering the VNA software.