

DISCLAIMER

This help file covers the hardwares **VNWA2** with firmware version **4.15** and **VNWA3/VNWA3E** with firmware version **5.22** and VNWA Software version **36.8.1**.

Disclaimer

The VNWA software is an experimental software targeted for educational purposes and for experimenters, e.g. radio amateurs.

I will take no responsibility for any damage that may result from its usage.

The VNWA software is still in beta-state . This means, that it may contain bugs and is subject to changes. If you find bugs, please report them.

►►► **Note:** Also file formats are subject to change, so **do not attempt to use older calibration files and ini-files on newer software versions!**

My home page

<https://www.dg8saq.darc.de/>

SDR-Kits home page

<http://www.sdr-kits.net/>

VNWA Web Forum

You are encouraged to post questions and suggestions regarding this software and the related hardware in the VNWA forum:

<https://groups.io/g/VNWA/>

E-mail

dg8saq@darc.de

Please read this help file carefully and consult the VNWA forum before mailing me questions.

This help file and its contents is intellectual property of Tom Baier, DG8SAQ. All rights reserved.

May 10th, 2023
Tom Baier DG8SAQ

PREREQUISITES

Prerequisites



This software was originally written to control a **DG8SAQ vector network analyzer (VNWA)** by means of a standard PC running Microsoft **Windows2000®** or **Windows XP®** via the **parallel printer port (LPT)**.

New software builds also support the **DG8SAQ USB_VNWA interface** and **Windows Vista / Windows 7 / Windows 8 / Windows 10** both 32bit and 64bit. Special thanks to Guido PE1NNZ for providing the 64bit WinUSB driver interface!

And sorry, it is not possible to compile this software for native Linux or MacOS support, as I make abundant use of Windows functions.

However, I have managed to run the software in **Linux** under a standard Wine installation (x86 processors only!) such that it can control the VNWA2/3/3E via USB interface. The Linux installation package including installation instructions can be found in the *Linux*-subfolder of the installation directory. In order to access it you must run the VNWA Windows-Installer under Wine. Omit driver and certificate installation.

Thanks to Michael Hartje DK5HH for helping me get going with Linux and for extensive testing.

The software can also control an N2PK VNA via LPT or via the USB interface of Dave Roberts G8KBB. Thanks to Paul Kiciak N2PK, Ivan Makarov VE3iVM, Andreas Zimmermann DH7AZ, Eric Hecker, Dave Roberts G8KBB and Roderick Wall VK3YC for supporting this part of the project. See here how to activate N2PK support.

Also thanks to all the beta-testers and users who have provided valuable feedback to optimize the software and this help file.

The software can also be run without hardware connected for data display and analysis purposes. If no VNWA is connected, select the LPT interface, otherwise a warning is issued (no VNWA detected) at every program start.

Software requirements:

-Windows2000, WindowsXP, Windows Vista, Windows7 or Windows8, alternatively a Wine installation emulating Windows in Linux, see above.

Hardware requirements:

-a Standard PC. Memory is not an issue, computational power is no longer an issue either, as the software has been optimized on CPU usage. The software has been verified to run properly on a 233 MHz PII in LPT mode and USB mode (with USB1.0 interface!).

If you still hit 100% CPU load during sweeping, your machine is too slow or your settings are unsuitable. See How to reduce CPU load in such a case.

-a sound capture device with stereo line input, which may also be connected externally via USB (integrated in the DG8SAQ VNWA_USB interface and in VNWA3 already).

-a parallel printer port interface or the DG8SAQ USB_VNWA interface (integrated in VNWA3 already).

ATTENTION: An USB to LPT adapter will not work due to critical timings! The DG8SAQ USB_VNWA interface takes over all the timing tasks from the PC.

OVERVIEW

Features of the VNWA2.* hardware in combination with the VNWA software

- Coverage from below 1 kHz to 500 MHz with dynamic range of up to 90dB, useful performance up to 1.3 GHz with reduced accuracy.
- Vector network analyzer mode and Spectrum analyzer mode
- Possibility to measure frequency converters and mixers in spectrum analyzer mode
- Low power consumption allows power supply out of a USB terminal.
- 2 port S-parameters S11, S12, S21 & S22, VSWR
- 3 port S-parameters and differential / common mode device analysis
- Component Measurements: Resistance, Admittance Capacitance, Inductance & Quality Factor (Q)
- Time domain reflectometry & gating in time domain + FFT
- Linear, Logarithmic and Listed sweep
- Matching tool and Complex Calculator
- Crystal analyzer tool for extracting equivalent circuit parameters
- User defined S-parameter calculus
- Optional S-parameter test set
- Optional USB-interface allows to run the VNWA with a single (USB) cable connection to the PC for highest mobility.

VNWA HARDWARE

Todate, the following different versions of DG8SAQ VNWAs are in use:

VNWA2 with LPT interface

VNWA2 with USB interface

VNWA3

VNWA3E = VNWA3 with Expansion Board

All variants share the same design und functional principle and thus the same technical data:

| **Technical Data**

- Unidirectional vector network analyzer for reflection and transmission measurements with integrated reflection bridge
- Frequency coverage <1 kHz...>1.3 GHz with overclocking 1)
- Dynamic range 1 kHz...500 MHz up to 90 dB, 500 MHz...1.3 GHz up to 60 dB 2)
- Maximum permissible external RF Input Level to TX-Out and RX-In ports: 0 dBm (0.225V RMS) 3)
- Maximum permissible external DC Input Level to TX-Out and RX-In ports: +/-0.225V 4)
- Power supply: 4.5...5.5V DC @500mA max, suppliable by a self powered USB interface
- Dimensions: Width 10.4 cm, Depth 8.0 cm, Height 4.6 cm, Weight 0.2 kg
- TX-Out max RF Output as signal generator -17dBm into 50 Ohms @ 1MHz decreasing at higher frequencies

1) Note, that clock multipliers of up to 20x leads to **dramatically overclocking the DDS chips with up to 720 MHz** core clock. This is well outside the DDS spec, which allows for 400 MHz maximum core clock. Nevertheless, the DDS chips just get hand warm when overclocked and we haven't seen a single failure in hundreds of shipped VNWAs.

2) Values apply to the slowest sweep rate of 100ms/frequency point. High accuracy results can only be expected up to 500 MHz, as this is the specification limit of the used mixers. Also, the DDSes show increasing spurs beyond 500 MHz.

3) We have tested with up to 7 dBm input level at various frequencies without failure.

4) Too much DC will burn the bridge resistors. All semiconductors are AC coupled.

| **History**

The VNWA was inspired by Paul Kiciac's N2PK VNA which was published in RadCom in October 2004.

In contrast to Paul's direct conversion approach I decided to go for low IF instead, as I wanted to use a PC sound card for signal processing from the beginning.

A design very similar to mine but without sound card was published by Jan Vandewege in 2005:

Java-enabled low cost RF vector network analyzer

De Mulder, B. Van Renterghem, K. De Backer, E. Suanet, P. Vandewege, J.

IMEC/Intec, Ghent Univ., Belgium

This paper appears in: IEEE-NEWCAS Conference, 2005. The 3rd International

Issue Date: 19-22 June 2005

I finally went public with my design in a public presentation and on my webpage in July 2006. The original presentation can still be viewed on <http://sdr-kits.net/DG8SAQ/VNWA/VNWA.pdf>

During August 2006 I submitted the VNWA for publication in the DARC CQ-DL magazine and in the ARRL QEX magazine. The articles appeared in 2007:

"Vektorieller Netzwerkanalyzer mit minimalem Hardwareaufwand", CQ DL, volumes 3,4,5/2007

"Low Budget Vector Network Analyzer from AF to UHF", QEX Mar/Apr 2007.

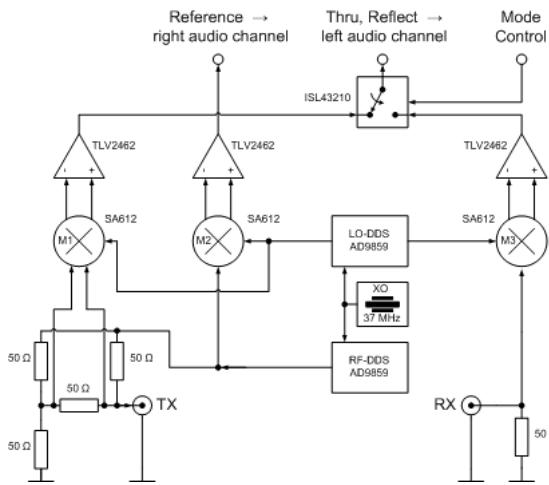
Further articles followed in 2007 and 2009:

"A Small, Simple, USB-Powered Vector Network Analyzer Covering 1 kHz to 1.3 GHz", QEX 1/2009.

"A Simple S-Parameter Test Set for the VNWA2 Vector Network Analyzer", QEX 5/2009

FUNCTIONAL PRINCIPLE

Function of the VNWA



The above schematic shows the fundamental design of the new VNWA. It consists of two digitally tunable Direct Digital Synthesizer (DDS) oscillators, realized by two fast Analog Devices AD9859 chips. The clock generation could be realized in a simple way, as the AD9859s contain an on-chip clock multiplier PLL circuit. Both DDSes are clocked from one and the same crystal oscillator delivering about 36 MHz. The exact crystal frequency is of no importance, as it can be accounted for in the VNWA software. It is crucial, that the two DDS cores are clocked with DIFFERENT frequencies, if one wants to omit anti-aliasing filters and make use of higher order alias frequencies. This is simply achieved here by setting the clock multipliers of the two DDS chips to two different values, e.g. 20 and 19, leading to clock frequencies of 703 MHz and 740 MHz respectively. Note, that in this frequency scheme, the DDSes operate well beyond their specification limits of 400 MHz maximum core clock. Quite remarkably, all tested DDS chips (many hundreds so far), work nicely without getting hot under these conditions. This is a very experimental approach, but pushing the DDS clock frequency means pushing the usable fundamental frequency range of the VNWA, which amounts to 600 MHz under the selected operating conditions.

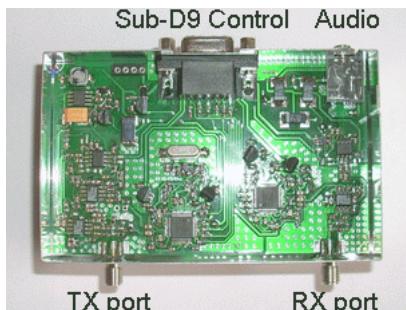
The RF-DDS output signal is fed into a VSWR-bridge formed by 50 Ohms resistors. The balanced bridge output signal is fed into the balanced inputs of the Gilbert cell mixer M1 (SMD type SA612), followed by an operational amplifier. The amplifier output signal is guided through a CMOS switch to the left channel 1) of the PC sound card line-in. The CMOS switch multiplexes this Reflect signal with the Thru measurement signal originating from mixer M3 and its following amplifier. The multiplexing is necessary, as standard sound cards only have a stereo line-in channel, which can only capture two signals simultaneously 2). But a third signal, the Reference signal, is required to acquire the phase information. The Reference signal is obtained by mixing the LO- and RF-DDS signals in M2 and amplifying the output signal with the following operational amplifier. It is then fed into the right channel 1) of the sound card line-in. Note that the VNWA works with an IF of about 1 kHz...12 kHz. The sound card is used as IF amplifier, the PC acts as a digital IF filter.

If a device under test (DUT) with two ports is placed between the TX- and RX-terminals of the VNWA, its scattering parameters S11 and S21 can be derived from the three measurement signals (Thru, Reflect, Reference). By manually inverting the device 3), also S12 and S22 can be measured.

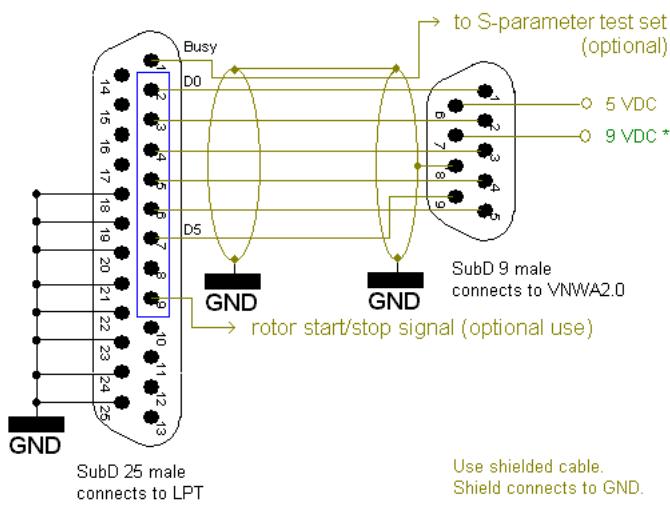
- 1) For the various VNWA versions the audio channel assignments may be flipped.
- 2) The VNWA software can simultaneously service two sound devices, thus the switch can be omitted if two sound cards are being used. Thru and Reflect can be measured simultaneously in this case.
- 3) or by using an automatic S-parameter test set

VNWA2 LPT INTERFACE

Connecting the VNWA with the PC using the LPT interface



As can be seen left, the VNWA2.* board has four connectors. The front SMA connectors TX port and RX port are the test ports for connecting the test object or DUT (device under test). The rear connectors are the VNWA's interface to the PC LPT port (or to the USB interface board). The audio-connector is a standard 3.5mm female one. Use a standard audio cable to connect the VNWA's audio port with the **stereo** line in of your PC's soundcard. **The microphone input is mono and thus won't work!** The female Sub-D9 connector carries both the control lines as well as the power supply lines. The connection scheme to the PC's parallel port (LPT) interface can be seen in the following:



The above schematic shows the cable connecting the PC's parallel (LPT) port with the VNWA Sub-D9 connector. **Don't apply 9V DC power, if you have mounted the step-up regulator MAX632 on the VNWA board.** If you have not mounted it, you need to provide two supply voltages, 5V DC and 9V DC.

If you want to power the VNWA from a USB port, you must have the step-up regulator in place and you need to connect an additional cable to the Sub-D9 connector with a USB connector on the other end.

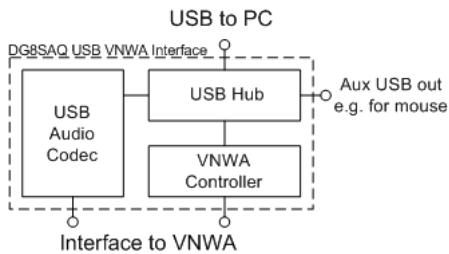
►►► **Hint:** In my instrument, I reroute the supply voltage through the parallel port cable and run the USB power cable out of the Sub-D25 LPT connector. This lets the LPT connector and the USB connector both end at the PC and leaves only one cable running to the VNWA (apart from the audio cable).

►►► **Hint:** In LPT mode, the software issues a high level on LPT port D7 on sweep start, which will return to low level on the end of the sweep. This signal is useful to start and stop an antenna rotor when performing antenna radiation pattern measurements.

►►► **Note:** I have observed extremely slow signal rise times on some new PCI LPT interface cards. This is a problem when controlling the VNWA. Should you encounter unreliable communication between PC and VNWA in LPT mode, you might need a signal conditioner, which sharpens the digital pulses.

VNWA2 USB INTERFACE

Functionality of the DG8SAQ USB_VNWA Interface



The left schematic shows the fundamental design of the DG8SAQ VNWA USB Interface. It connects to the host PC with a single USB cable and performs both control and data acquisition. At the same time it provides the power supply for the VNWA from the PC via the USB cable.

The host PC USB signal is fed into the USB hub, which provides 3 USB interfaces. One of them is used to connect an on board USB sound codec, which captures the VNWA audio signals. The second USB hub interface connects to the VNWA controller, a microcontroller, that takes care of the sweep timing and DDS control, thus relieving the host PC from a lot of realtime action compared to the LPT modes. The third USB hub interface is not needed and thus available for custom use like a memory stick or a second sound codec.

Warning: The free USB hub interface is only intended for low power devices like a mouse or a memory stick. You might damage your PC and / or your DG8SAQ USB_VNWA Interface by connecting high power USB devices as no power detection / control is implemented inside the DG8SAQ USB_VNWA Interface!

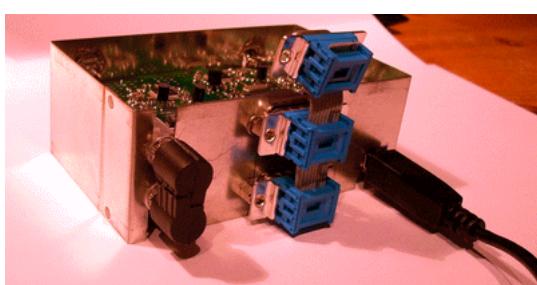
Note: The free USB hub interface does not support full speed, thus transfer rates to memory sticks will be limited to about 1MByte/s.

Connecting the DG8SAQ USB_VNWA Interface to the VNWA and to the host PC

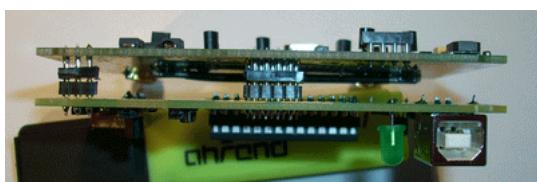
The DG8SAQ USB_VNWA Interface simply connects with a **single standard USB cable** to the host PC. The interface has been proven to work on PCs with USB1.0 and USB2.0 interfaces.

There are two possibilities to connect the DG8SAQ USB_VNWA Interface to the VNWA:

1. External connections



As can be seen in the picture above, the audio and data connections can be made via the external 3.5mm connectors and the Sub-D9 connectors. This allows to upgrade an original LPT mode VNWA to USB support. Also, VNWA and USB-Interface may be housed in different boxes. Note, that the unused top Sub-D connector at my ribbon cable is for diagnostic purposes.

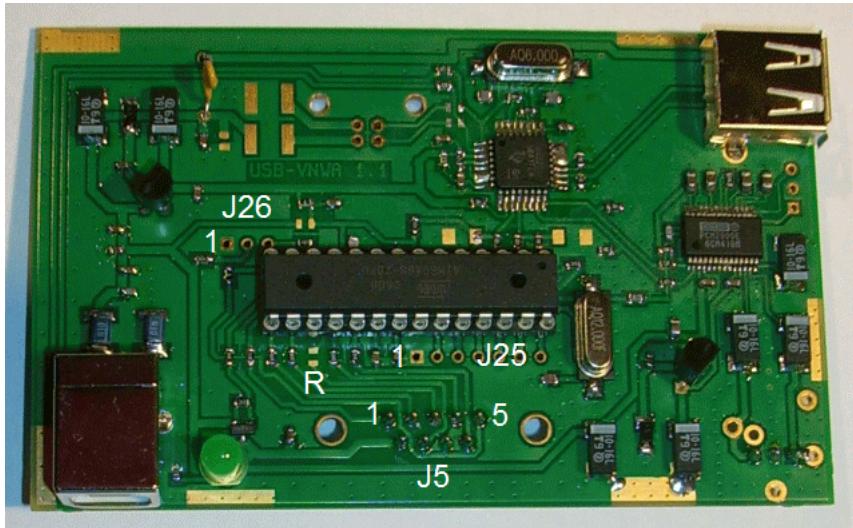


The late VNWA boards support direct plugging into the USB interface as can be seen above. The Sub-D connectors and audio connectors are replaced by appropriate 2.5mm inter-pcb connectors in this case. No additional external

connections will be required other than the USB cable to the host PC.

Optional External Connections:

The **control signal for the S-parameter test** set is available on the USB_VNWA Interface at pin 3 / J26 or alternatively pin 7 / J5 (Sub-D9) provided the appropriate resistor R=680 Ohms (not placed on below board) is in place:



Pinning overview of J26:

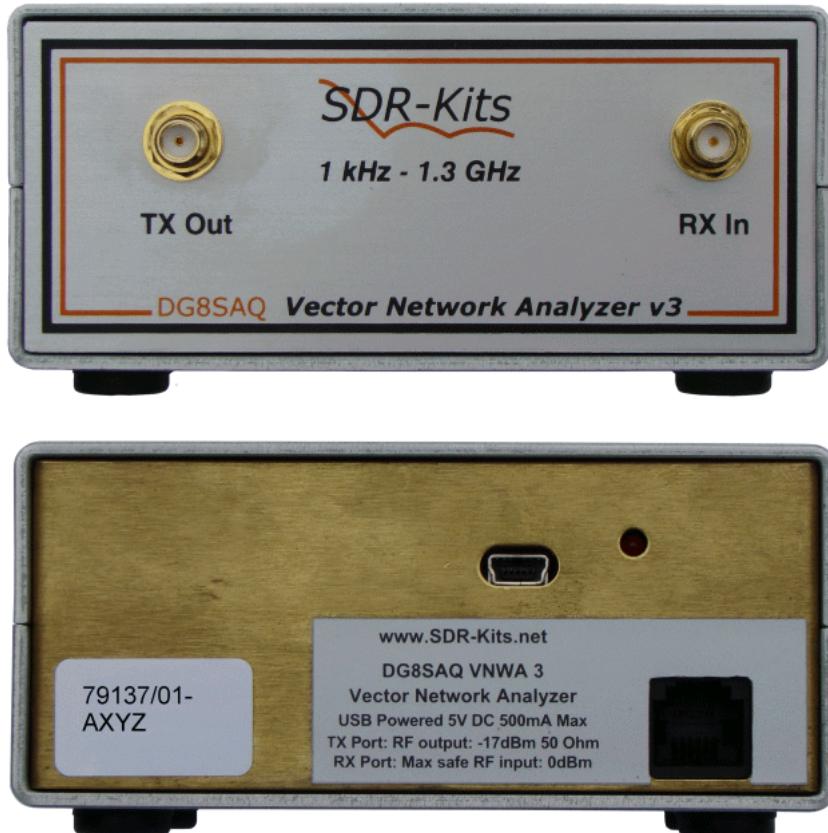
- Pin 1: Ground
- Pin 2: 5V DC (through 47 Ohms internal resistor for protection)
- Pin 3: S-Parameter test set control signal

From firmware v4.12 on a **Rotor start/stop** signal has been implemented on AVR Port C0 (=Pin 23), which is accessible on J25 pin 2. The pin will go high at start of a sweep and go low again when the sweep is finished. This signal can be used to control an antenna rotor for antenna radiation pattern measurements.

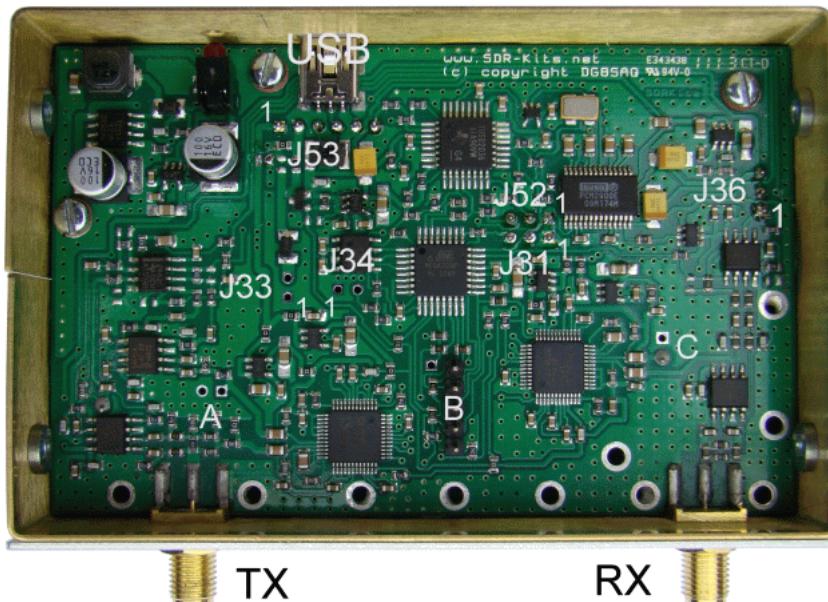
Pinning overview of J25:

- Pin1: Ground
- Pin2: Rotor start/stop signal
- Pins 3...7: not used

VNWA3



The VNWA3 combines the VNWA2 and the USB interface functionality on a single 100x60mm² board:



The VNWA3 only requires a single USB cable to connect to a PC. Nevertheless, it offers many hardware interfacing options:

VNWA3 Pinning

J31 Control port

Pin 1: Ground

Pin 2: Rotor start/stop control signal

Pin 3: S-parameter test set control signal

J33 External Power

Pin 1: Ground

Pin 2: external +5V VNWA power supply

J34 External Clock

Pin 1: Ground

Pin 2: Internal clock output or external clock input (1kOhm impedance, 400mVpp out or -6dBm...+3dBm / 50 Ohms in)

J36 Aux Audio Output

Pin 1: RX out

Pin 2: Ground

Pin 3: Reference output

J52 Aux USB Port

Pin 1: Ground

Pin 2: USB Hub Port 2 D+

Pin 3: USB Hub Port 2 D-

J53:

Pin 1: external DDS power in (+2.7...5V, 0.3A), see footnote 1)

Pin 2: switched +5V power out

Pin 3: USB cable Ground

Pin 4: USB cable D+

Pin 5: USB cable +5V

Pin 6: USB cable D-

A:

RF DDS test point

B:

socket for production purposes only, no user functionality.

C:

LO DDS test point

VNWA3E



1. Introduction:

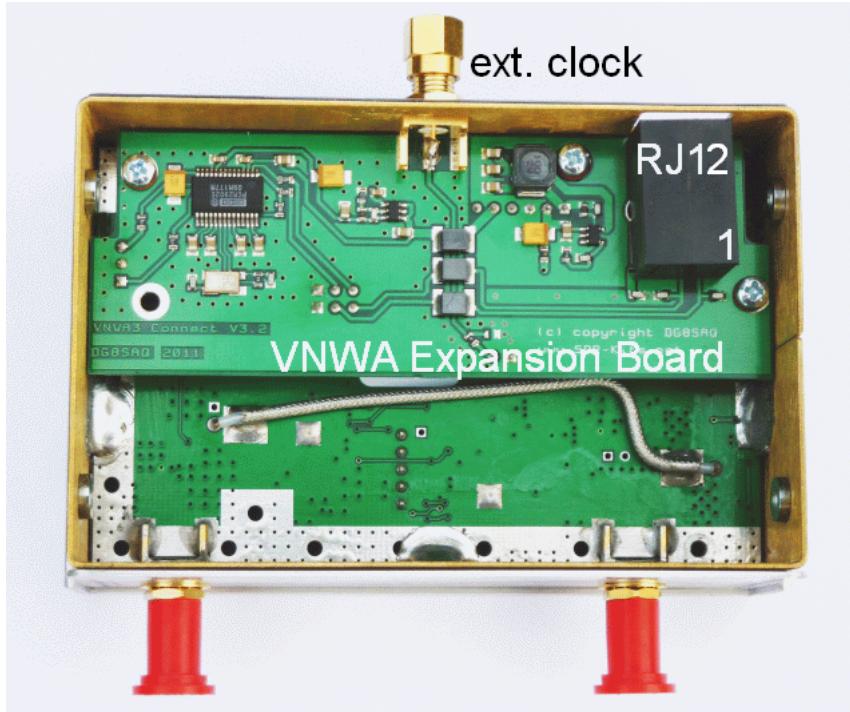
VNWA3E was formerly named VNWA3+ but both are the same product. The DG8SAQ VNWA3E is the VNWA3 fitted with the VNWA3 Expansion Board which provides additional functionality as follows:

- 2nd Audio Codec allowing S11 and S21 measurements to be performed in a single sweep instead of 2 sweeps.
- USB power consumption reduced from 0.41A without Expansion board to 0.33A despite additional power consumption of 2nd Audio Codec on the Expansion board.
- RJ12 connector making the VNWA3 control signals and DC power accessible to the outside world.
- Optional SMA connector to obtain the multiplied 12 MHz internal TCXO clock or to feed in an external 36 MHz clock source.

►►► **Note:** All VNWA3 instruments can be upgraded to VNWA3E. Consult <http://www.sdr-kits.net/> for details.

2. VNWA3 Expansion Board

The VNWA3 Expansion Board plugs into several of the VNWA3 connectors. No additional connections are required. The expansion board contains an additional audio codec, a switcher supply for the DDSes, which more than compensates the additional codec's power consumption (see footnote 1) and an RJ12 connector, that makes the VNWA3 control signals accessible from the outside world.



RJ12 Pinning overview:

- Pin 1: Rotor start/stop control signal
- Pin 2: not connected
- Pin 3: S-parameter test set control signal
- Pin 4: not connected
- Pin 5: internal 5V DC power out or external 5V power in, see footnote 2)
- Pin 6: Ground

Signal Details:

RJ12 pin	Signal Description	Cable colour	Remarks: (see also VNWA helpfile for info on options)
Pin 1:	Rotor start/stop control signal Output 0V or 3.3V DC - 680 Ohm Series Resistor	white	Output signal, normally +0V. When VNWA sweep is active Output signal is +3.3V DC – This signal can be used to start and stop Rotor Engine.
Pin 2:	Not Connected	black	
Pin 3:	S-Parameter Test set Control Signal (output)	red	0 or 3.3V Logic DC Signal from AVR via 680 Ohm series resistor. See Helpfile for configuration details
Pin 4:	Not Connected	green	
Pin 5:	Internal 5V DC Power out or External 5V Power In (See Note 1)	yellow	A maximum of about 100mA may be drawn from internal power provided PC USB Hub can supply 500mA. See Note 1; when using external power
Pin 6	Ground	blue	VNWA Ground Connection

Note 1: For operating the VNWA3 from an external 5V DC power supply, a zero Ohm resistor must be removed, see footnote 2).

Note 2: Cable colour refers to the colours of the optional RJ12 cable available from SDR-Kits

Optional External Clock input/output

An Optional SMA connector may be fitted for connecting an external 36 MHz clock output or to tap the internal multiplied system clock. After the VNWA 3 Expansion Board is fitted, the SMA connector is fitted in the 8mm round hole provided only in VNWA3 with s/n 2201 and higher.

- >With a scalpel cut a round hole in the rear panel label and fit the SMA connector.
- >solder a short wire between the SMA center connector to the PCB track. Soldering of ground connections is not required.

If the SMA connector is not used then it is recommended to fit an SMA screening cap to prevent radiation from and damage to the SMA connector.

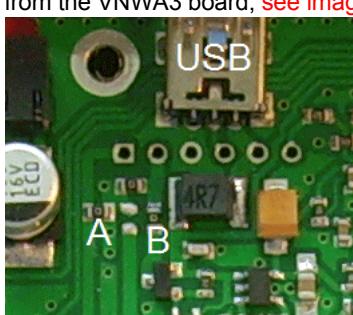
Specification of the VNWA Reference Clock output or External Clock input:

- >Internal Clock Output: TCXO 12 MHz*x (x=2..8) 2ppm Out - 1kOhm impedance, 400mVpp output
- >External Reference Clock: ideally 36 MHz (=> VNWA2 functionality achieved) with level of -6dBm...+3dBm / 50 Ohms

Footnotes

1) **Important!** For operation with external DDS power e.g. from the expansion board, zero Ohm resistor A must be removed from the VNWA3 board, [see image below](#).

2) **Important!** For operating the VNWA3 from an external 5V DC power supply, zero Ohm resistor B must be removed from the VNWA3 board, [see image below](#).



3. Configuring VNWA3E Audio Codec

3.1.1 Plug the VNWA3E with Expansion Board fitted into USB port of Personal Computer.

Start the VNWA Application, then select "File" and "Setup" "USB Settings" and perform "Test USB Interface" to confirm that VNWA is operational.

3.1.2 Select Tab "Audio Level" and tick the box "Auxiliary Audio Capture Device Available" in fig 1. which is applicable for Windows XP operating system. This will bring up the configuration of the Auxiliary Audio Capture Device. Change the settings as shown in fig 1.

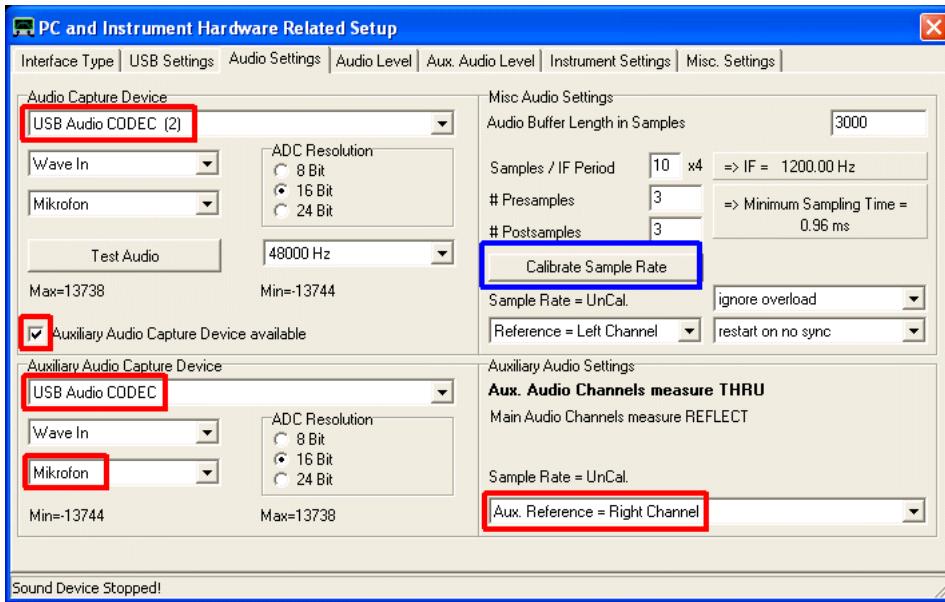


Fig 1. Configuration of Auxiliary Audio Capture Device

Troubleshooting:

If the "Auxiliary Audio Capture Device available" checkbox is NOT shown in fig 1., then exit VNWA application and add the line AuxAudio=1 to the file VNWA.ini with a text editor and then restart the VNWA Application.

►►► **Note:** With the VNWA 3 Breakout PCB fitted there are now two USB Codec devices, a) USB Audio Codec and b) USB Audio Codec 2. The on-VNWA3-board USB codec must be the main capture device, as it is hard wired to the multiplexing switch. Configuration is done in the following step:

3.1.3 Press the button "Calibrate Sample Rate" as shown in fig 1. If Sample Rate Calibration fails (First pulse not detected) then reverse the two USB Audio Codec devices and test again (i.e. Audio Capture becomes "USB Audio Codec" and Aux. Audio Capture becomes "USB Audio Codec 2").

3.1.4 Now select the tab "Audio Level" and select "Reflect" mode with no termination on the TX-port, and check the blue main Reference signal and blue Reflected signal are as shown in fig 2. Do not worry about the lighter red Aux Audio signal, which may also be visible.

►►► **Note:** If the sinewaves are "clipped" (Vista or Windows 7) then the Recording level of the relevant USB Audio Codec needs to be changed. Also check whether the USB Audio Codec is set to Stereo 48000Hz.

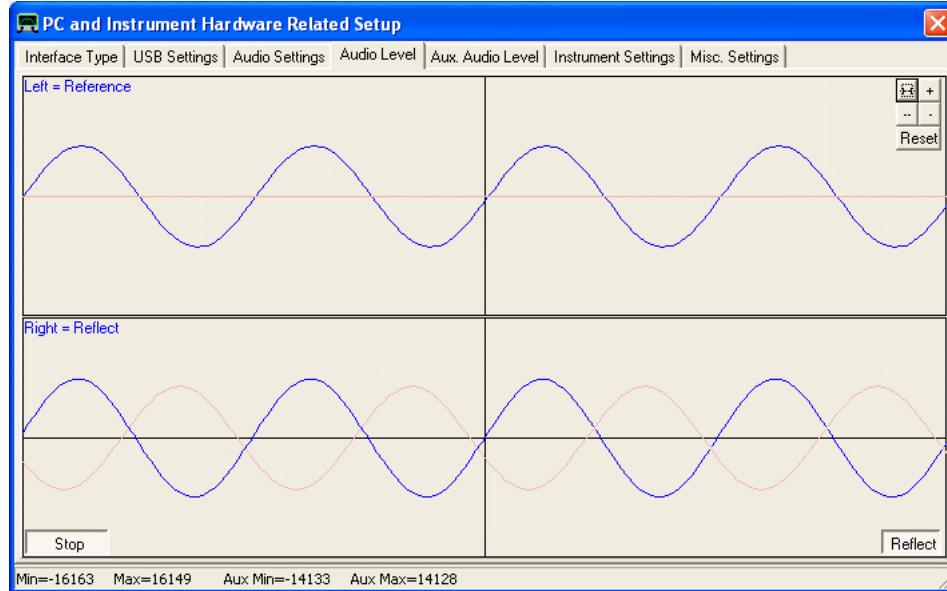


Fig 2. Audio Level with TX port not terminated (full Reflection)

3.1.5 Connect a short coax cable between TX and RX ports and check the Reflect sinewave in fig 1. now becomes a flat blue line.

3.1.6 Next select the tab "Aux. Audio Level" and check the display as shown in fig 2. is obtained.

►►► **Note:** If the sinewaves are "clipped" (Vista or Windows 7) then the Recording level of the relevant USB Audio Codec needs to be changed. Also check whether the USB Audio Codec is set to Stereo 48000Hz.

USB Audio

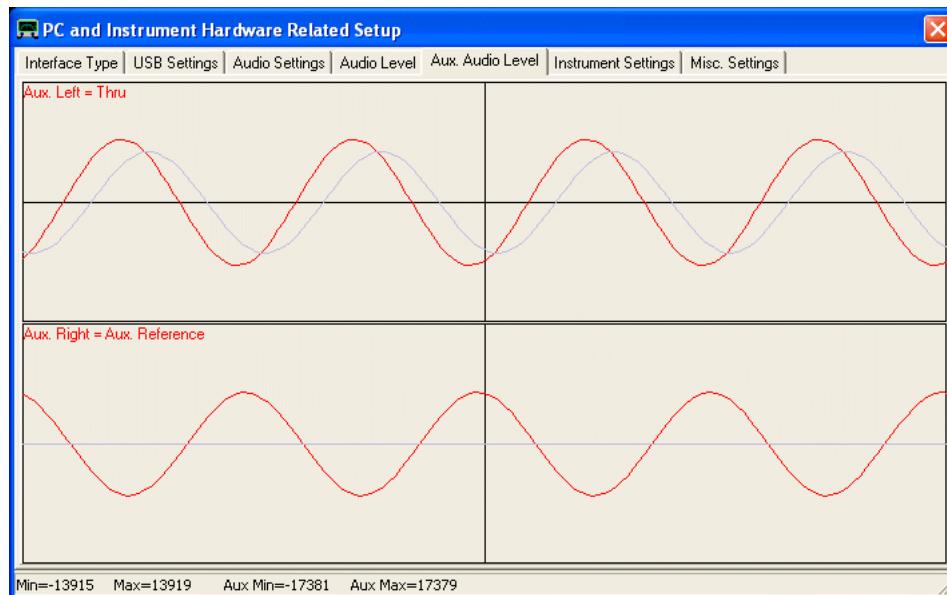


Fig 2. Aux. Audio Level with TX and RX ports connected

Troubleshooting: If the above displays are not obtained then check the USB-Audio Settings in fig 1. and perform step 3.1.3 again.

STARTUP PROCEDURE

Steps to operate the DG8SAQ VNWA instrument

- 1) Preparing the VNWA hardware for connection
- 2) Software and driver installation
- 3) Software configuration
- 4) Adjusting sweep Settings
- 5) Instrument calibration
- 6) Performing a Measurement

AUTOMATIC DRIVER AND SOFTWARE INSTALLATION

From VNWA software version V35.0 on an **automatic installer file (VNWA-installer.exe)** is available which will automatically guide you through the installation of the VNWA software, the USB driver and the help file system. It can be downloaded from https://www.sdr-kits.net/index.php?route=web/pages&page_id=30_30.

From VNWA software version V36.2.z, the VNWA software will guide you after first launch through an automatic setup, see page "Automatic Configuration of the Software for Usage with a DG8SAQ VNWA (Auto-Setup)".

Software beta-versions must still be installed manually by simply replacing the old VNWA.exe in the installation directory by the new one or by creating a separate new installation manually.

Note, that you can have several installations in different directories at the same time, which can even operate several VNWAs simultaneously connected at the same time.

►►► **Warning:** If you use the automatic installer on **64 bit operating systems**, it is recommended that you do NOT install into protected system directories like "c:\Program Files" or "c:\Program Files (x86)". As you will not be the owner of the installed files, VNWA might not be able to access its ini-files and you might not be able to manually replace an old VNWA.exe version by a newer one.

►►► **Warning:** A similar issue might arise when you intend to **run VNWA in a non-Administrator user account**: In this case, you should NOT start VNWA as administrator after software installation. Make sure to **initially start the VNWA software in the non-Administrator account** which you intend to use in future. This will make sure that the non-privileged user is the owner of the ini-files generated by VNWA upon first program start. Otherwise the VNWA software will not be allowed to update the ini-files upon program termination.

A detailed installation guide for the automatic installer can be downloaded from here:

<http://sdr-kits.net/documents/VNWA3-installation.pdf>

MANUAL SOFTWARE AND DRIVER INSTALLATION

►►► **Note:** It is good practice to **make regular backups of your data** on your PC, particularly before installing drivers. Installing the VNWA software and drivers is by no means any more dangerous than installing any other Windows software. **SDRKits and the author (DG8SAQ) will not be responsible for any damage or data loss, though.**

Todate the USB-version of the VNWA is supported by all Windows versions from Windows 98 to the latest Windows 7 64 bit. If the VNWA is to be controlled via an LPT port instead of USB, generally only 32 bit Windows systems are supported.

Please consult the "Driver Compatibility Table and Driver Installation".

The latest driver files and software updates are always available from the **VNWA Groups.io Newsgroup**

If you want to use the VNWA application without VNWA hardware, you may go straight on to **installing the VNWA application**.

DRIVER COMPATIBILITY TABLE AND DRIVER INSTALLATION

USB support

Thanks to Fred PE0FKO there is a **unified signed LibUSB driver available for all Windows versions** from Windows98 to Windows8 64bit.

Note, that you **must use VNWA V33.0 or newer and firmware v4.6 or newer to use this signed driver**. For earlier firmware versions use the unsigned LibUSB driver (only installable on 32 bit systems).

►►► **Important for Vista / Windows 7 64 bit users:** Only VNWA firmware v4.6 or newer and VNWA software v33.0 or newer are supported.

►►► **Important for Windows 8 users:** Only VNWA firmware v4.6 or newer and VNWA software v36.2 or newer are supported.

Windows 8 / 64 bit requires installation of the amateur radio root certificate PRIOR to driver installation. The automatic VNWA installer takes care of this.

If you have received your VNWA after April 2010, you have already received firmware v4.6 or higher. If you have a working VNWA installation, you can also check your firmware version with the VNWA software in "**setup-USB settings**" by pressing "**Test USB**". After the test is performed successfully, the firmware version is being displayed in the bottom status line.

Installation examples:

Example 1: Installing the LibUSB driver on Windows XP (English screenshots)

Example 2: Installing the LibUSB driver on Windows 7 (mostly English screenshots)

Example 3: Installing the LibUSB driver on Windows XP (German screenshots)

Example 4: Installing the LibUSB driver on Windows 7 (German screenshots)

LPT support

LPT mode is NOT supported on Windows 8.

LPT support compatibility table

Windows Driver	Windows				
	98	2000	XP 32 bit+64 bit	Vista + W7 32 bit	Vista + W7 64 bit
zportio	no	<input checked="" type="checkbox"/>	XP 32 bit only!	<input checked="" type="checkbox"/> *	not installable

* only, if program started with administrator rights

►►► **Note:** In order to enable LPT support, you must copy the driver file **zportio.sys** into your VNWA program directory.

►►► **Note:** If you do not use the VNWA USB interface and you do not intend to install the LibUSB driver, you must manually provide the file **libusb0.dll**. You must copy the libusb0.dll version from the LibUSB driver package into your VNWA program directory in this case.

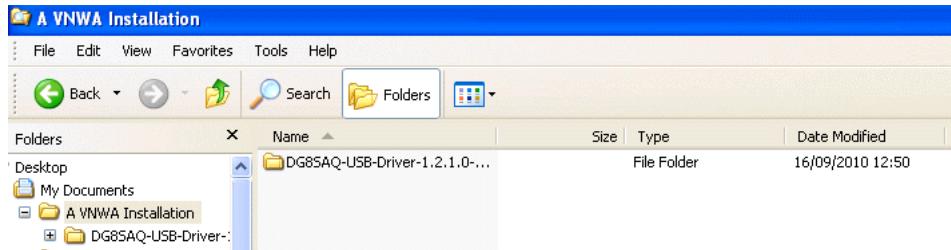
►►► **Note:** Since the parallel printer interface (LPT) is subject to extinction, LPT support for 64 bit OSes will not be implemented.

If you do not intend to install USB drivers, you can go straight on to installing the VNWA application software.

EXAMPLE 1: INSTALLING THE LIBUSB DRIVER ON WINDOWS XP (ENGLISH SCREENSHOTS)

►►► **Note:** This step is only required if the **DG8SAQ USB_VNWA** Interface is used. **Skip this step if you control your VNWA via the parallel printer port.**

a) Download and unpack the LibUSB driver files to any empty directory:



b) Connect the DG8SAQ USB_VNWA Interface with your PC. The VNWA board doesn't need to be connected to the interface board. Windows will notify "Found New Hardware" and will ask to install a driver.

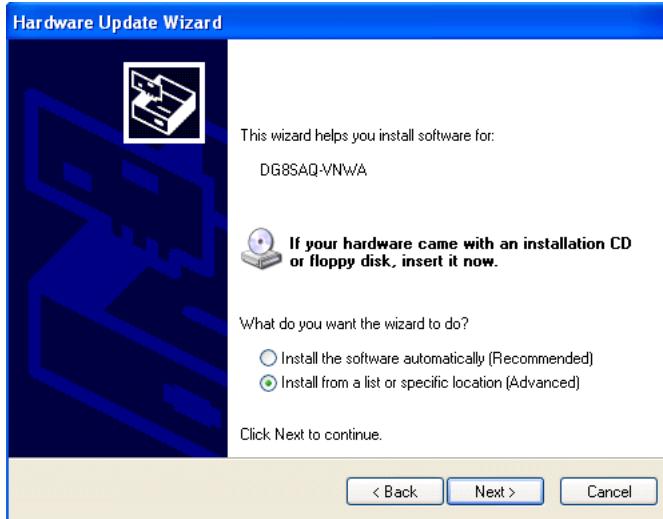


►►► **Hint:** In case Windows does not detect your USB_VNWA interface, continue with the USB troubleshooting guide. In case Windows does not start driver installation automatically, proceed with manual driver installation via the Windows device manager as is described in Example 2: Installing the LibUSB driver on Windows 7 (mostly English screenshots).

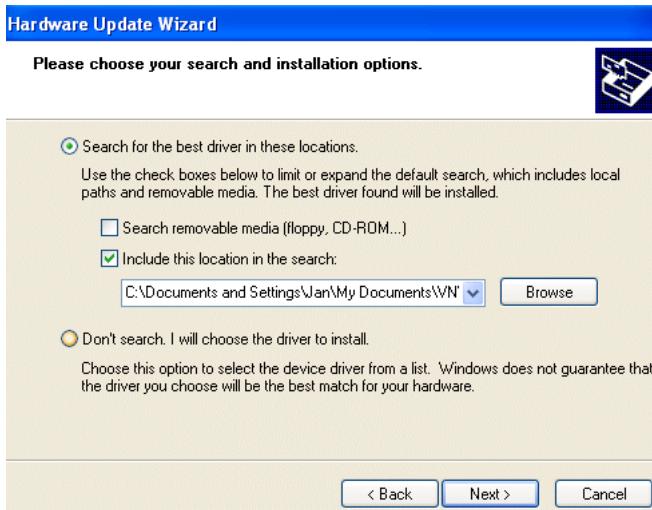
c) Click on "no, not this time" and press continue to prevent driver search on the web.



d) Click on "install from a list or specific location (Advanced)" and press continue to install the driver from your hard disk.

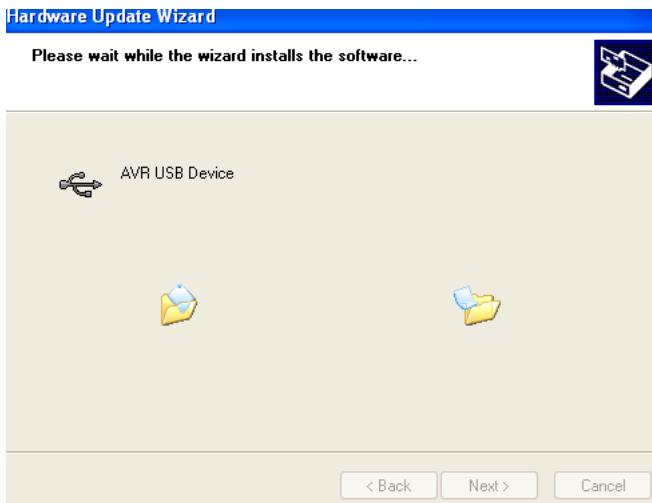


e) Browse for the path to your driver files ...



confirm path with ok ... and press >Next on the previous window.

f) The driver will be installed...



Finally you will be notified about the successful installation. Press "Finish" and you can use the USB_VNWA interface on your computer.



Now you are all set to install the VNWA application software.

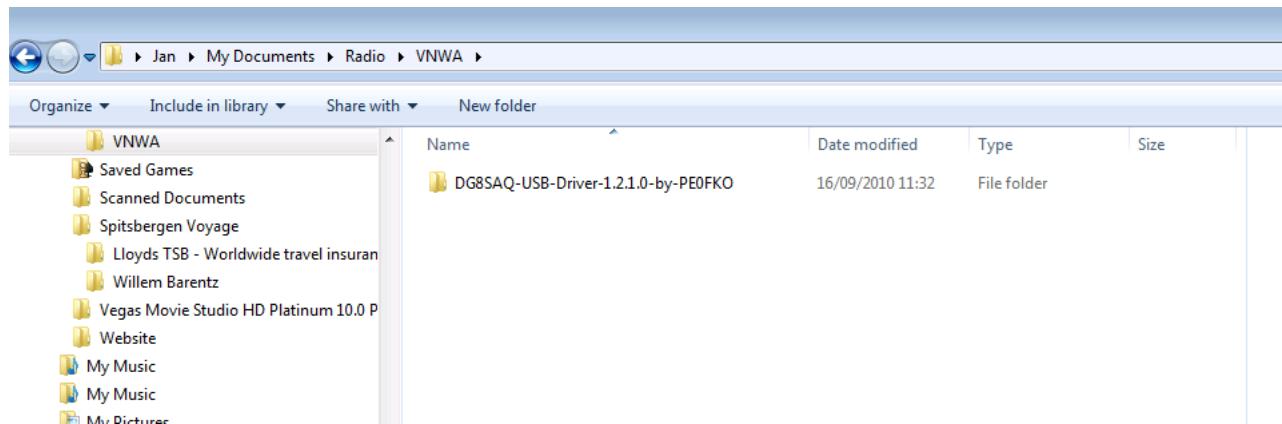
EXAMPLE 2: INSTALLING THE LIBUSB DRIVER ON WINDOWS 7 (MOSTLY ENGLISH SCREENSHOTS)

►►► **Note:** This step is only required if the **DG8SAQ USB_VNWA** Interface is used. Skip this step if you control your VNWA via the parallel printer port.

►►► **Note:** You need VNWA software version **V33.0** or higher and firmware version **v4.6** or newer to work with the signed LibUSB driver.

►►► **Note** The following describes a step by step installation procedure on a Windows 7 / Windows Vista machine, where no previous attempt has been made to install a VNWA driver. If you previously have tried to install different USB drivers for the VNWA **make sure to completely remove all previously installed driver relicts from your system**.

0. Download and unpack the LibUSB driver files to any empty directory your hard disk.



1. Connect your VNWA to a USB port of your computer

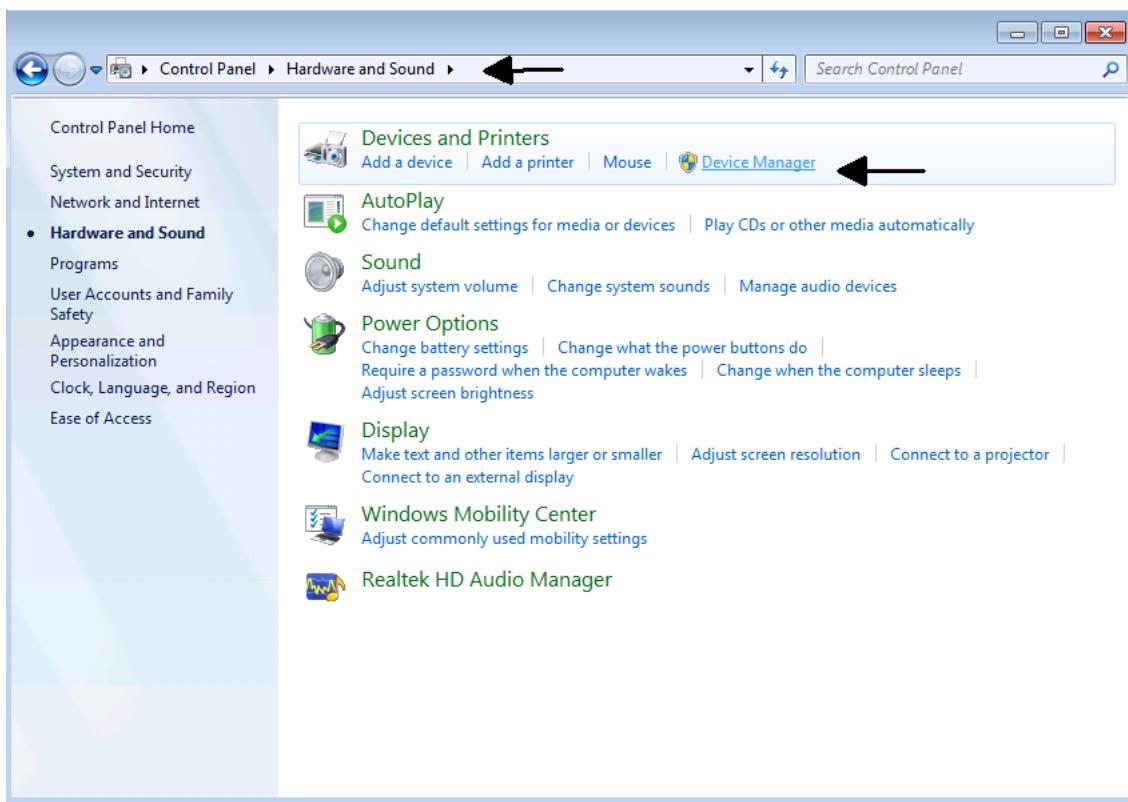
Windows 7 will attempt to install USB drivers for the VNWA ...



... but it will fail as it can't find any drivers.

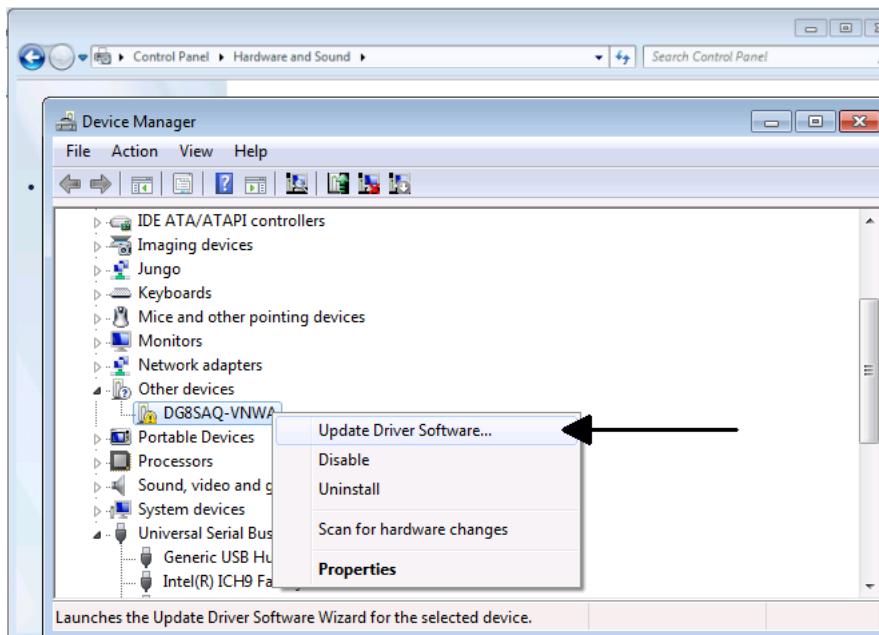


2. Open your device manager



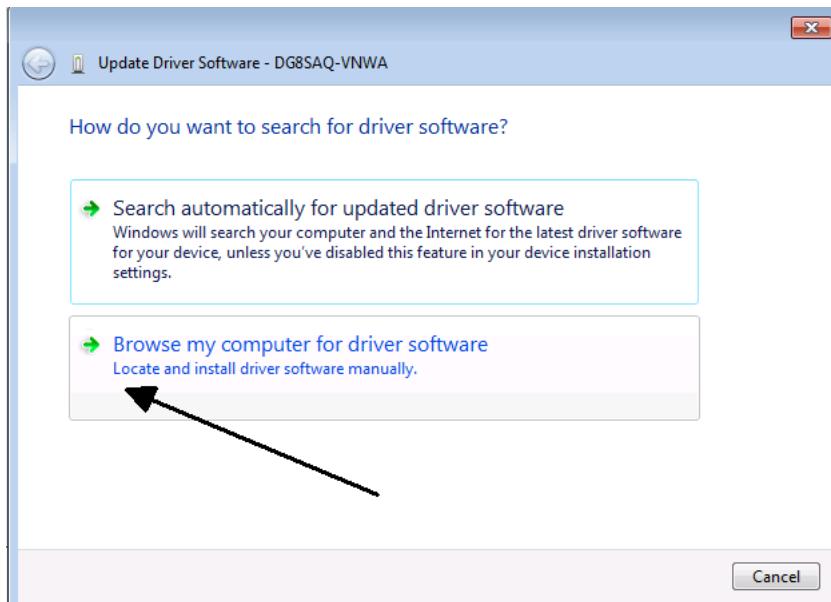
You will see the VNWA as up to now non functional DG8SAQ-VNWA device under "other devices":

3. Right-click DG8SAQ-VNWA and select "update driver software"

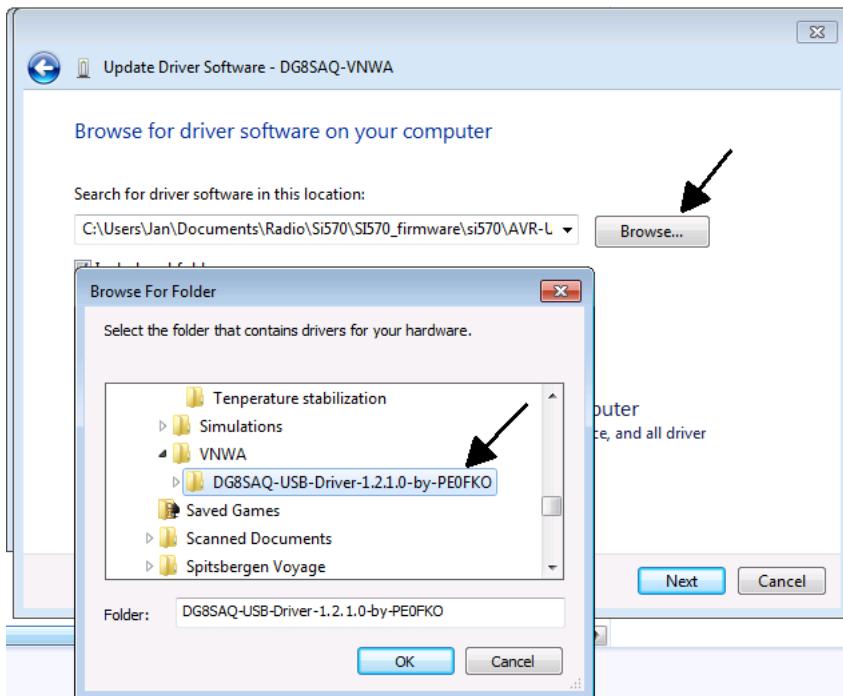


Windows 7 will ask you how you would like to search for driver files.

4. Select "Browse my computer for driver software"



5. Select the unzipped directory which contains the drivers...

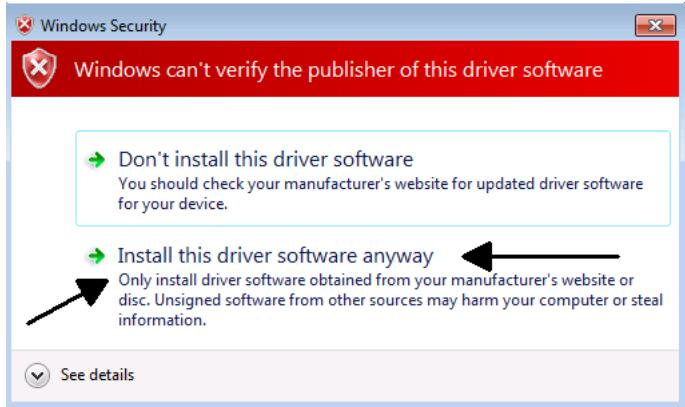


... and press Next.

Windows 7 will issue a security warning "publisher of driver software could not be verified".

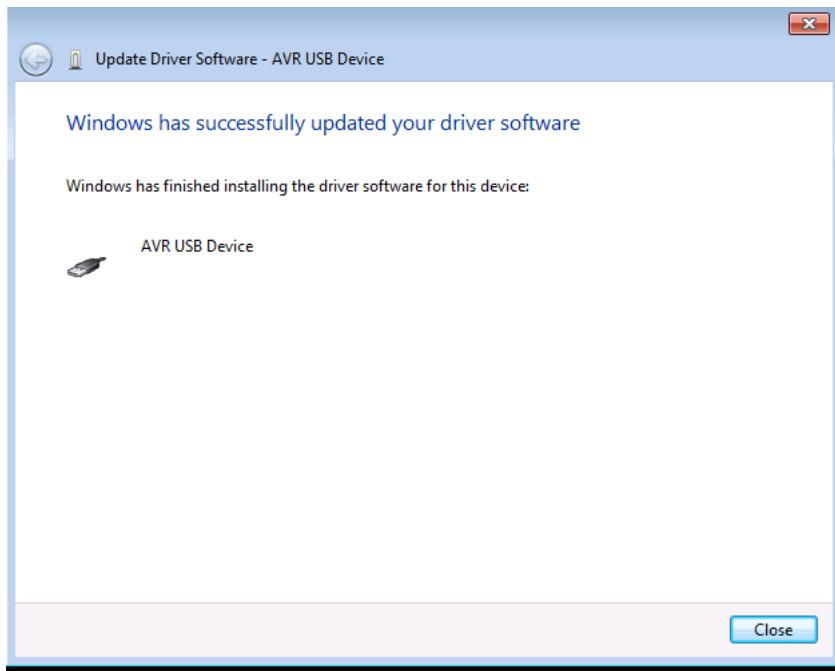
Even though the USB driver is signed, Windows 7 will issue a security warning "publisher of driver software could not be verified". This is ok and no problem.

If you desire, the warning can be avoided by installing the root certificate prior to driver installation, but this is not required for installing and running the driver.



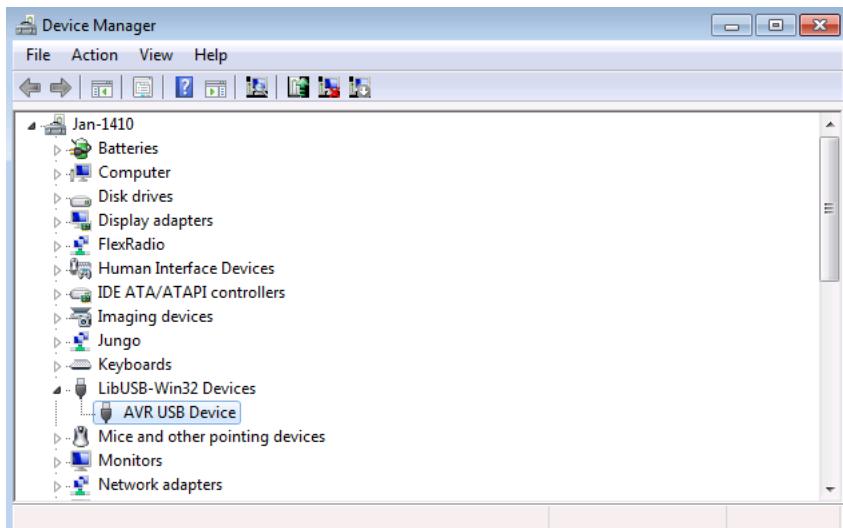
6. Select "install this driver software anyway ..."

... and Windows 7 will start installing the driver software. This might take a couple of minutes until done:



7. Press the close button

The USB driver is now installed and the VNWA device is operational as can be seen in the device manager, where the VNWA will show as AVR USB device:



Now you are all set to install the VNWA application software.

EXAMPLE 3: INSTALLING THE LIBUSB DRIVER ON WINDOWS XP (GERMAN SCREENSHOTS)

►►► **Note:** This step is only required if the **DG8SAQ USB_VNWA** Interface is used. **Skip this step if you control your VNWA via the parallel printer port.**

a) Download and unpack the LibUSB driver files to any empty directory:

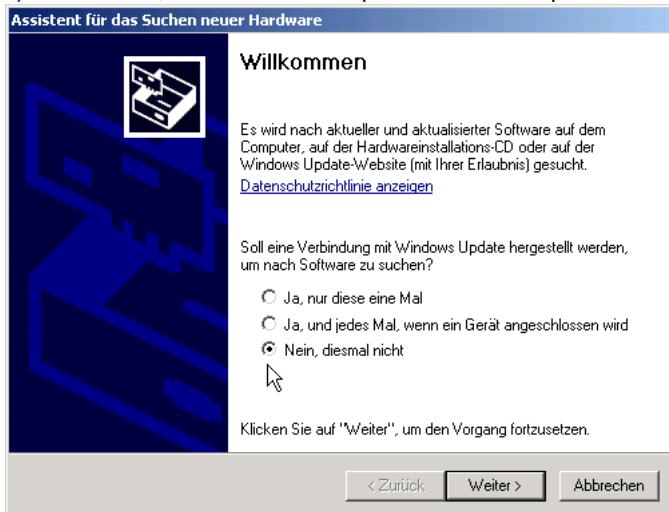


b) Connect the DG8SAQ USB_VNWA Interface with your PC. The VNWA board doesn't need to be connected to the interface board. Windows will notify "new hardware found" and will ask to install a driver.

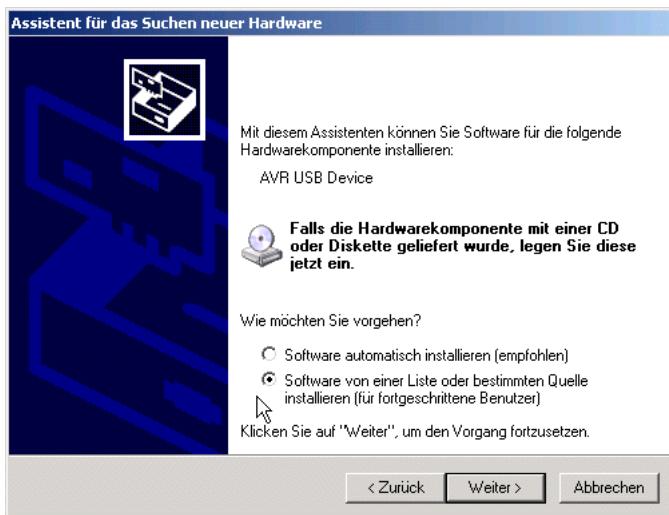


►►► **Hint:** In case Windows does not detect your USB_VNWA interface, continue with the USB troubleshooting guide. In case Windows does not start driver installation automatically, proceed with manual driver installation via Windows device manager as is described in Example 4: Installing the LibUSB driver on Windows 7 (German screenshots).

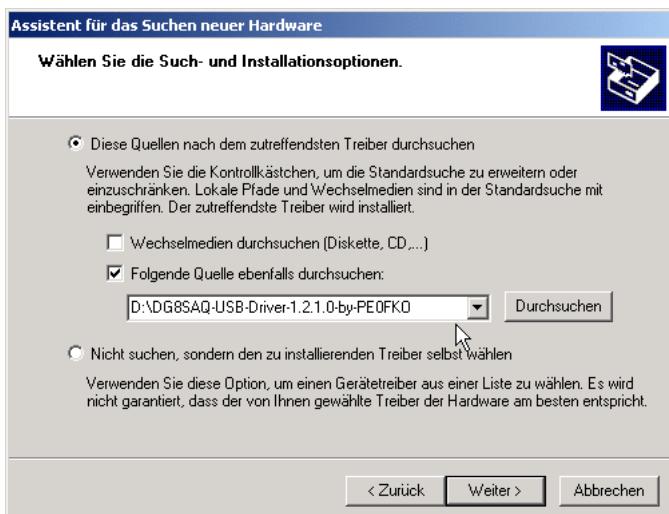
c) Click on "no, not this time" and press continue to prevent driver search on the web.



d) Click on "install from list..." and press continue to install the driver from your hard disk.



e) Enter the path to your driver files and press continue.



f) The driver will be installed. Finally you will be notified about the successful installation. Press "finish" and you can use the USB_VNWA interface on your computer.



Now you are all set to install the VNWA application software.

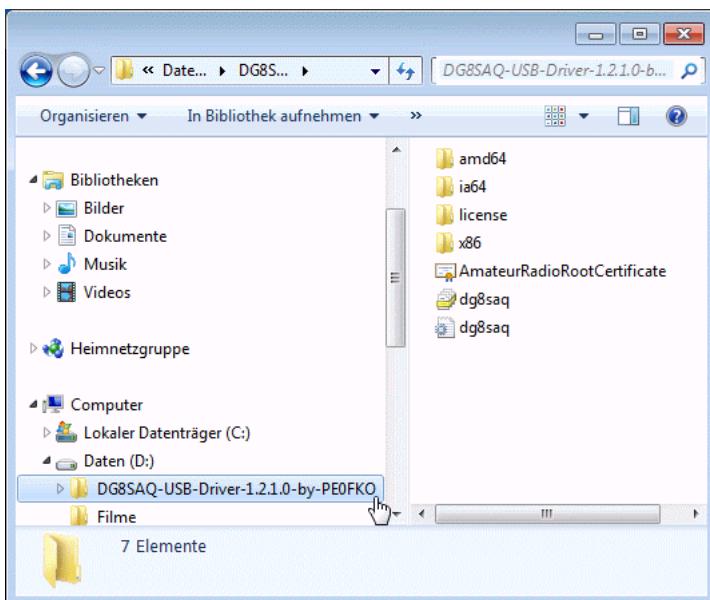
EXAMPLE 4: INSTALLING THE LIBUSB DRIVER ON WINDOWS 7 (GERMAN SCREENSHOTS)

►►► **Note:** This step is only required if the **DG8SAQ USB_VNWA** Interface is used. Skip this step if you control your VNWA via the parallel printer port.

►►► **Note:** You need VNWA software version **V33.0** or higher and firmware version **v4.6** or newer to work with the signed LibUSB driver.

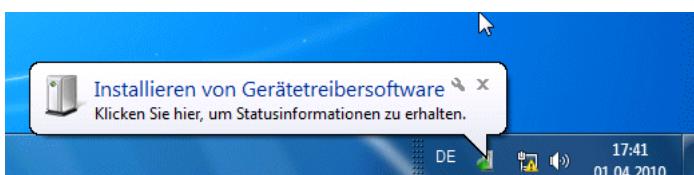
►►► **Note** The following describes a step by step installation procedure on a Windows 7 / Windows Vista machine, where no previous attempt has been made to install a VNWA driver. If you previously have tried to install different USB drivers for the VNWA **make sure to completely remove all previously installed driver relicts from your system**.

0. Download and unpack the LibUSB driver files to any empty directory your hard disk.

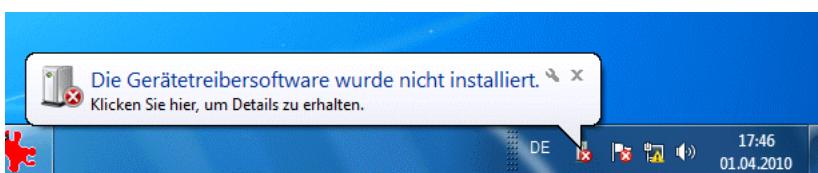


1. Connect your VNWA to a USB port of your computer

Windows 7 will attempt to install USB drivers for the VNWA ...

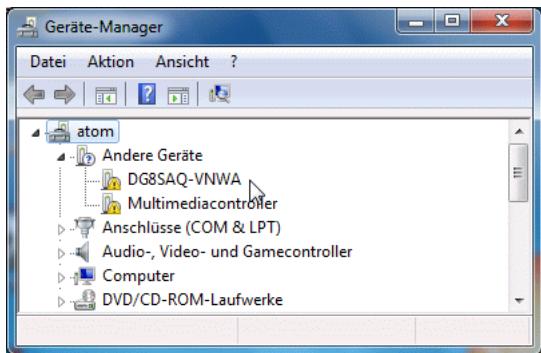


... but it will fail as it can't find any drivers.

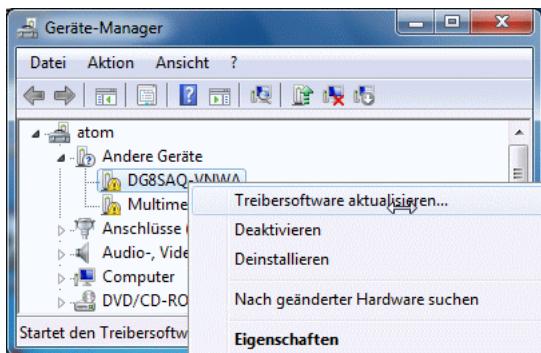


2. Open your hardware manager

You will see the VNWA as up to now non functional DG8SAQ-VNWA device under "other devices":



3. Right-click DG8SAQ-VNWA and select "update driver software"

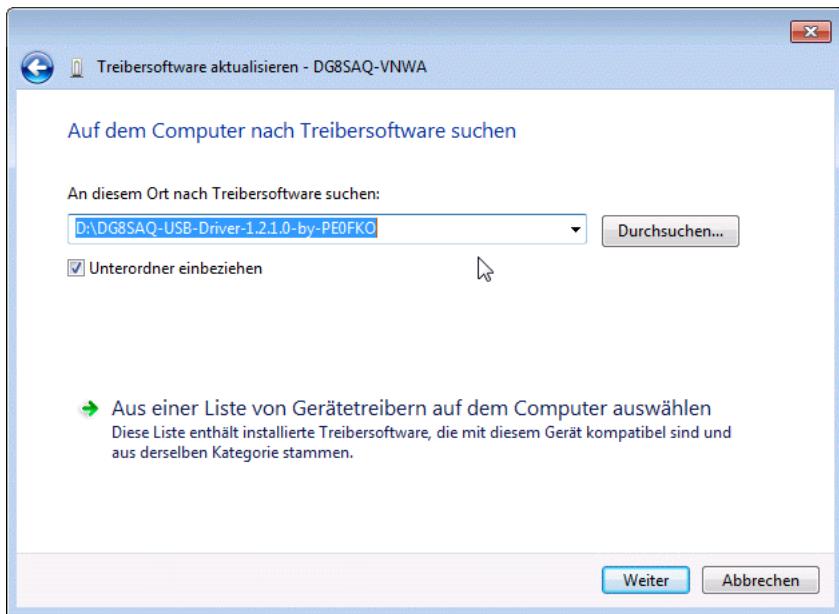


Windows 7 will ask you how you would like to search for driver files.

4. Select "search on computer for driver files"



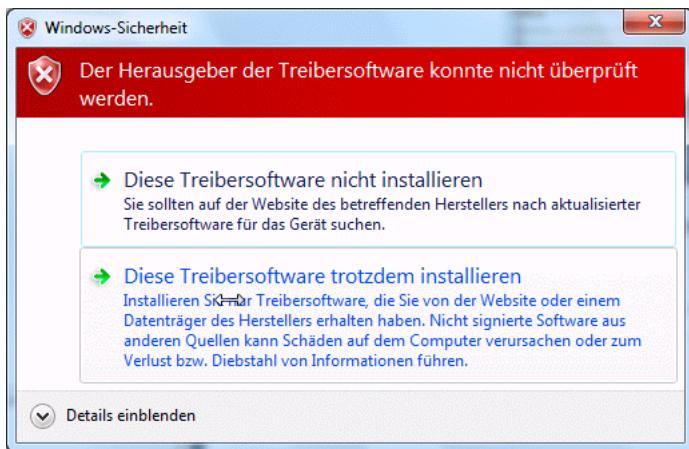
5. Select the unzipped directory which contains the drivers...



... and press continue.

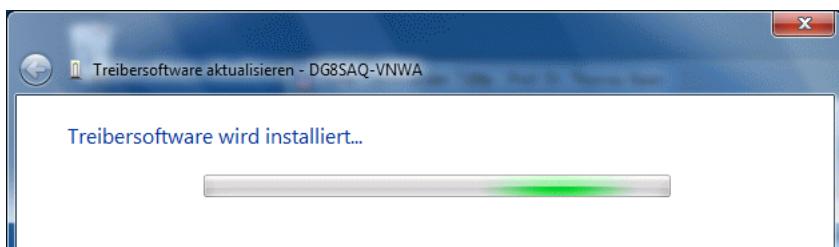
Even though the USB driver is signed, Windows 7 will issue a security warning "publisher of driver software could not be verified". This is ok and no problem.

If you desire, the warning can be avoided by installing the root certificate prior to driver installation, but this is not required for installing and running the driver.

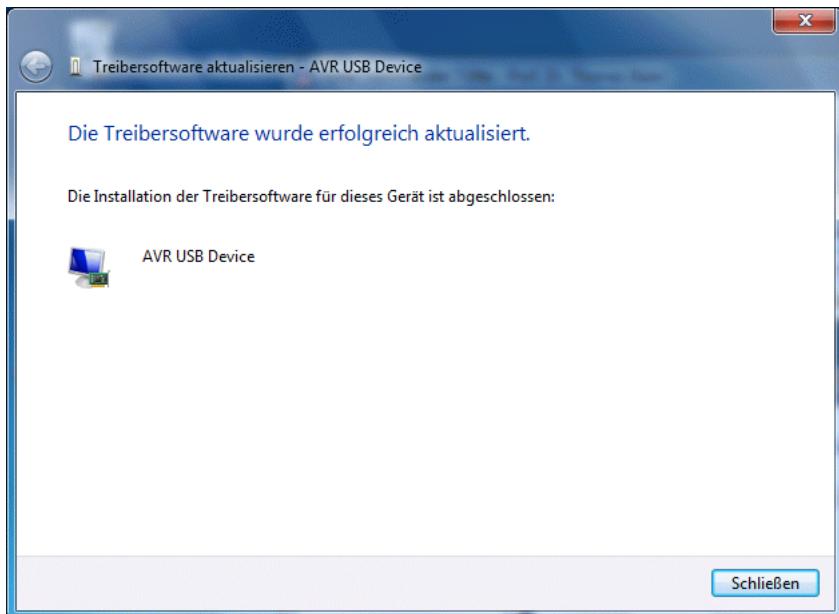


6. Select "install anyway ..."

... and Windows 7 will start installing the driver software.

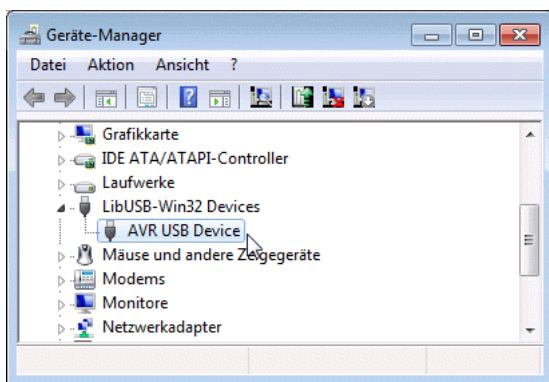


This might take a couple of minutes until done:



7. Press the close button

The USB driver is now installed and the VNWA device is operational as can be seen in the hardware manager, where the VNWA will show as AVR USB device:



Now you are all set to install the VNWA application software.

MANUAL INSTALLATION OF THE VNWA APPLICATION

The VNWA hardware is controlled by the **VNWA application software**. Installing the software is a means of simply copying the file VNWA.exe to any directory you wish to use as program directory.

You can even install the VNWA application onto a memory stick which may remain attached to the extra VNWA USB port of your VNWA2. The advantage of doing so is that when moving the VNWA from one computer to another you also move all your settings and calibration files along.

►►► **Windows Vista / Windows 7 / Windows 8 note:** On Windows Vista / Windows 7 or 8 the "program files" directory on the system partition is particularly write protected. Since the VNWA application writes many files (ini-files, calibration files,...) into the program folder, it is recommended to **NOT install the VNWA application into any system folder like "program files"** in order to avoid conflicts. Use any folder where you have normal write access instead.

Installation

You should start out with an empty program folder into which you unzip the VNWA application **VNWA.exe**, the LPT driver file **zlpportio.sys** if LPT support is desired, special function dlls (if desired) and the help file system "**VNWA.HLP**" and "**VNWA.cnt**". No special software installation procedure is required as the files are ready to run. The VNWA application does not make use of the Windows registry. This is particularly interesting for trying out software beta versions in a folder separate from the main installation folder.



If you have received a **key file** (*vnwakey*.ini or *vnwakey*.txt) together with your VNWA hardware from SDRKits, copy it into the same installation directory. From software version 35.0 on VNWA will look for a key file on first program start after installation and read the key information. This way, you do not have to enter the key manually.

►►► **Note:** If you have not installed a USB driver for the VNWA, then you must copy the file **libusb0.dll** from the LibUSB package into this directory.

►►► **Note:** You need to have **administrator privileges** in order to run the application in **LPT** mode. If you attempt to run it with lower privileges, the LPT driver will fail to load. The driver will also fail to load on systems that do not support the LPT driver zlpportio.sys (e.g. Win98 and all 64 bit Windows versions).

►►► **Note:** You can **add an icon** to your desktop by right-clicking onto VNWA.exe and selecting *copy*. Then right-click onto your Windows desktop and select *add link*.

►►► **Windows Vista / Windows 7 or 8 note:** Please consult "**Configuring and running the VNWA application under Windows Vista, 7 or 8**" for special Vista / Windows 7 or 8 issues. Note, that Windows Vista and Windows 7 / 8 require that you set the VNWA audio device inside Windows to stereo operation and that you manually adjust the recording level.

Now you're ready to control your DG8SAQ VNWA. See the **configuration guide** for details

If you want to control an N2PK VNA with the VNWA application, then you need to create an additional file named **N2PK.ini** with arbitrary contents in the program folder. You can do so by e.g. typing `echo > n2pk.ini` on the console prompt. If you want to control your N2PK VNA via **Dave Robert's G8KBB USB Interface**, you must provide the files **delphivna.dll** and **vnadll.dll** in your VNWA program directory. Configuration instructions for controlling an N2PK VNA can be found here.

DESCRIPTION OF FILES IN INSTALLATION FOLDER

The automatic VNWA software installer creates the following subdirectories files inside the VNWA program directory.

The colors refer to the following categories:

- Mandatory for usage of VNWA
- Only needed for special operations
- Not required by VNWA application

Directories:

./DG8SAQ-USB-Driver Directory containing USB drivers. Once installed, the directory is no longer needed.

./firmwares Directory containing all published firmwares. Only needed for firmware upgrades

Files:

CodeclF.dll Audio level control interface dll for Windows Vista, 7, 8. If omitted, recording levels have to be set via Windows manually.

delphivna.dll Interface dll for N2PK-VNA, not needed for DG8SAQ VNWA

DelZip190.dll zipper dll used to pack debug info, not needed if debug emailer function is not used

gfx_dll.dll dll used to load background image. Not needed if no background image is used

helpfile change history.txt help file change history, not needed

logo.bmp SDRKits logo, VNWA also runs without this file.

svg_dll.dll dll used to load background image in *.svg format. Not needed if no background image is used

unins000.dat uninstaller data base, not needed for VNWA operation

unins000.exe VNWA application uninstaller, not needed for VNWA operation

VNWA.cnt Help file table of contents, only needed if help file is used.

VNWA.exe VNWA application

VNWA.hlp VNWA help file, not needed for VNWA operation

VNWAKey.ini VNWA license key, needed for VNWA hardware operation

After running and terminating the VNWA application for the first time, the following additional files are created.

The colors refer to the following categories:

- Will be recreated after each program termination
- Recreation can be suppressed

FList.csv Listed sweep frequency list. Will only be accessed upon program start and when a listed sweep is edited.

MemorySpaces.dat Storage for data spaces, creation and access can be deactivated

Paths.ini Links to directories visited by user in open/save dialogs will be stored here. File will only be read once upon program start.

PreSetup.* Set of files containing a backup of the instrument state before entering setup. Will only be created when setup is opened. Creation may be suppressed.

VNCommon.ini General software settings. Will only be read once upon program start.

VNWA.cal Current instrument calibration. Will only be read once upon program start.

VNWA.ini DG8SAQ VNWA hardware related settings. Will only be read once upon program start.

VNWA.mul DG8SAQ VNWA2 clock multiplier table for auto-clock. Will only be read once upon program start.

VNWA3.mul DG8SAQ VNWA3 clock multiplier table for auto-clock with fixed premultiplier. Will only be read once upon program start.

VNWA3A.mul DG8SAQ VNWA2 clock multiplier table for auto-clock with auto-premultiplier. Will only be read once upon program start.

Depending on selected instrument and mode, the following additional calibration files might have been generated:

VNEB.cal Calibration for "external bridge" mode

VN2B.cal Calibration for "internal+external bridge" mode

VNIV.cal Calibration for "RF-IV mode"

SA.cal Calibration for "Spectrum Analyzer" mode

n2pk.cal Calibration for "N2PK VNA" instrument

If in any of the above mentioned modes a master calibration has been performed, then there will be additional master calibration files of the form *.cal with user defined master calibration file names.

After opening the Setup window in the VNWA application for the first time, a set of files with name PreSetup.* are created.

These files are present a so called **Instrument State**, i.e. the complete instrument settings at the point of time before entering Setup. Thus, a wrong setup can be undone by manually retrieving this instrument state. Note, that the files PreSetup.* are being updated every time you enter the Setup menu. Automatic creation of these files can be deactivated in Setup-Misc. Settings-Special Settings section.

A basic instrument state contains the following files (shown for the example instrument state MyState.*):

MyState.bck_mdat	Copy of MemorySpaces.dat (not saved for PreSetup)
MyState.bck_mul	Copy of VNWA.mul
MyState.bck_mul3	Copy of VNWA3.mul
MyState.bck_mul3a	Copy of VNWA3A.mul
MyState.bck_vcal	Copy of VNWA.cal
MyState.bck_vncom	Copy of VNCommon.ini
MyState.bck_vnwa	Copy of VNWA.ini

The following additional mode calibration files may be included in the instrument state if available:

MyState.bck_vmcal	Copy of VNWA mode master calibration
MyState.bck_vebc	Copy of VNEB.cal
MyState.bck_vmebc	Copy of "external bridge" mode master calibration file
MyState.bck_ve2c	Copy of VN2B.cal
MyState.bck_vme2c	Copy of "internal+external bridge" mode master calibration file
MyState.VNIV.cal	Copy of VNIV.cal
MyState.bck_vmivc	Copy of "RF-IV mode" mode master calibration file
MyState.bck_scal	Copy of SA.cal
MyState.bck_smcal	Copy of "Spectrum Analyzer" mode master calibration file
MyState.n2pk.cal	Copy of n2pk.cal
MyState.bck_nmcal	Copy of "N2PK VNA" instrument master calibration file

The user may manually create the aditional file:

N2PK.ini N2PK VNA general hardware settings

links.url User customizable internet links for Help-Home menu

When emailing debug information or downloading software updates from the internet, the following directory is created:

tmp Directory will hold compiled debug information or downloaded files

Internet access can be suppressed by deactivating the automatic check for software updates.
In remote-control mode internet access is automatically blocked.

FIRMWARE INFORMATION

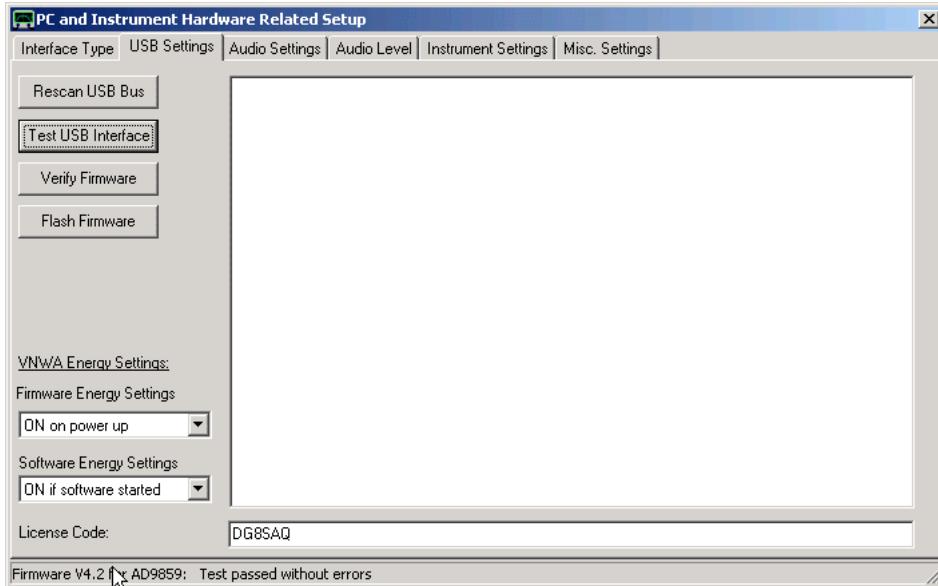
The VNWA hardware contains a microcontroller running its own software. This software is called firmware.

Current firmware versions are:

V4.15 for all VNWA2 versions

V5.22 for all VNWA3 versions

You can check your firmware version by using the "**Test USB**" feature in Setup-USB Settings. The result will be displayed in the bottom status line:



It can be seen in the bottom status line (see mouse pointer above), that firmware version v4.2 is running on attached VNWA2. Note, that V4.2 is OLDER than 4.15.

The versions increment like 5.1...5.0,5.10...5.19,5.20...5.22, likewise for the 4.x versions.

Note, that firmwares can no longer be updated by the user. If a firmware update is necessary, it must be performed using ISP-progammer hardware.

VNWA SOFTWARE UPDATES

VNWA software updates can either be performed all manually via the SDRKits software update webpage https://www.sdr-kits.net/index.php?route=web/pages&page_id=30_30 or from within VNWA using the main menu "File - Software Updates".

AUTOMATIC CONFIGURATION OF THE SOFTWARE FOR USAGE WITH A DG8SAQ VNWA (AUTO-SETUP)

From VNWA software version V36.2.z on an **automatic setup wizard** is available. It will automatically start on first program launch after installation, **provided the VNWA hardware is connected**.

The following describes a typical setup run on a Windows 8 machine, which behaves very similar to Windows7 and Vista. Notes will point out differences to Windows XP.

1. Connect the VNWA hardware and launch the VNWA application

If you haven't already entered the license code during software installation, make sure you have your license code at hand.

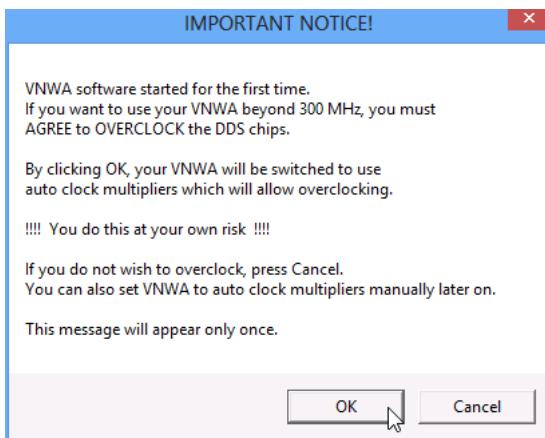
Connect the VNWA hardware to your computer using a USB cable and give your computer a minute to automatically configure the VNWA hardware drivers.

Then start the VNWA software by double-clicking the VNWA icon on the desktop (or VNWA.exe after manual installation):



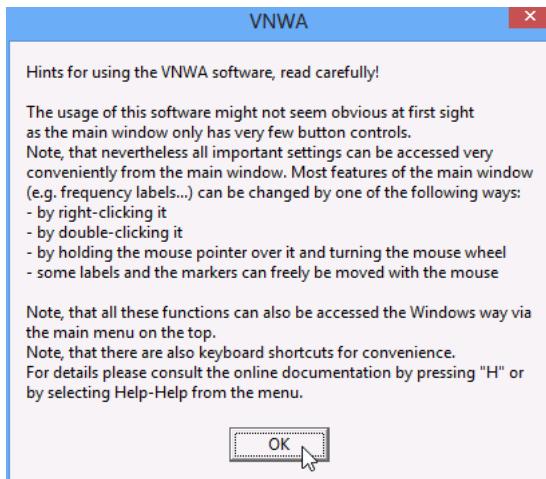
2. First informations

In order to use your VNWA over the full frequency range (up to 1.3GHz and beyond), the DDS chips inside must be overclocked. In the long history of the VNWA several thousand DDS chips have been overclocked and we have not observed a single failure. You still do overclock at your own risk.



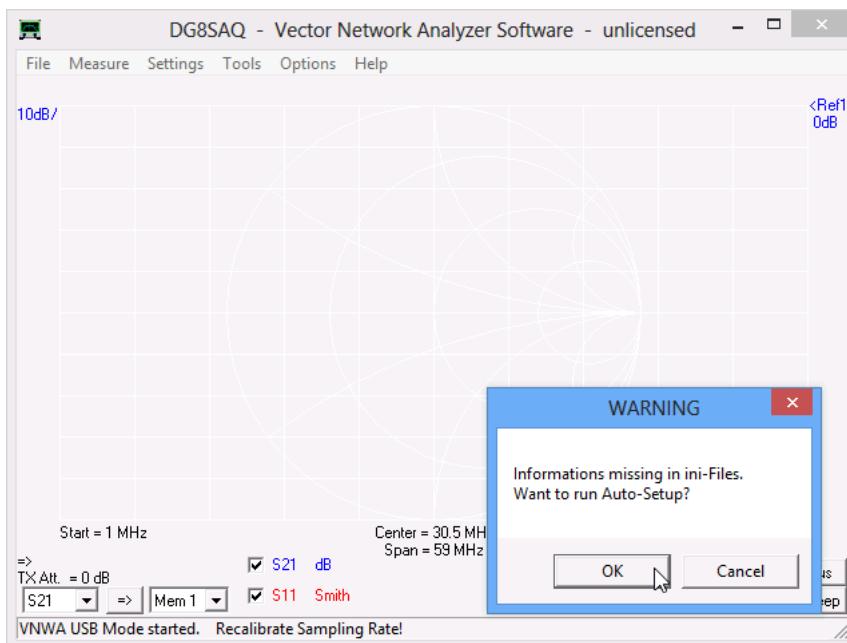
Press OK if you agree to overclock. Press Cancel or close the window using the X-button if you do not want to overclock.

Next, some useful first hints about not so obvious ways to operate the software are shown:



Also this window will show only once. Close it after having read by pressing OK or using the X-button.

The application main window comes up and the software detects that a setup is necessary:



Pressing Cancel or closing the Warning window with the X-button will abort the guided setup and will activate the main window.

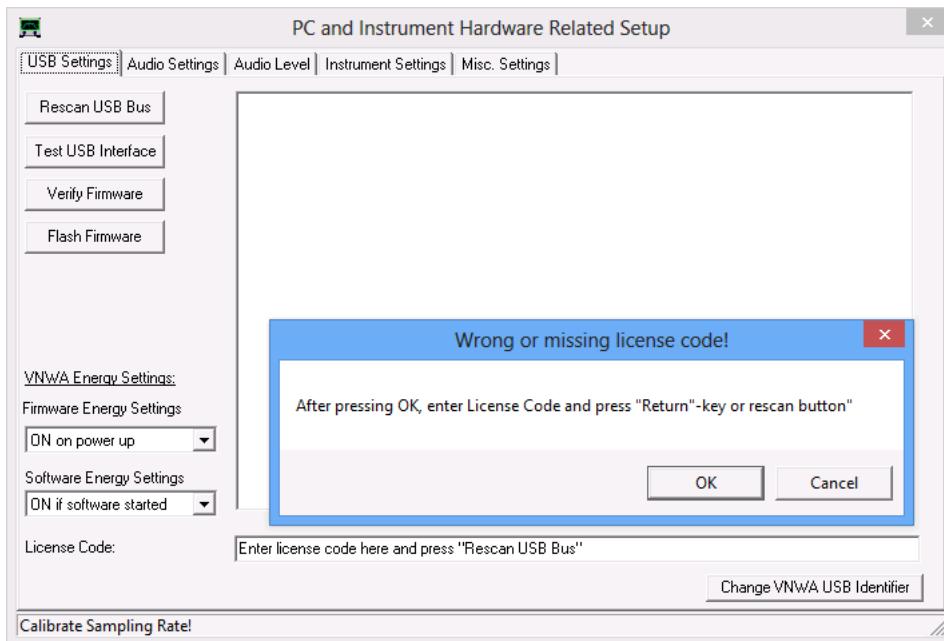
In order to proceed with the guided setup press OK.

3. Automatic setup, entering license code

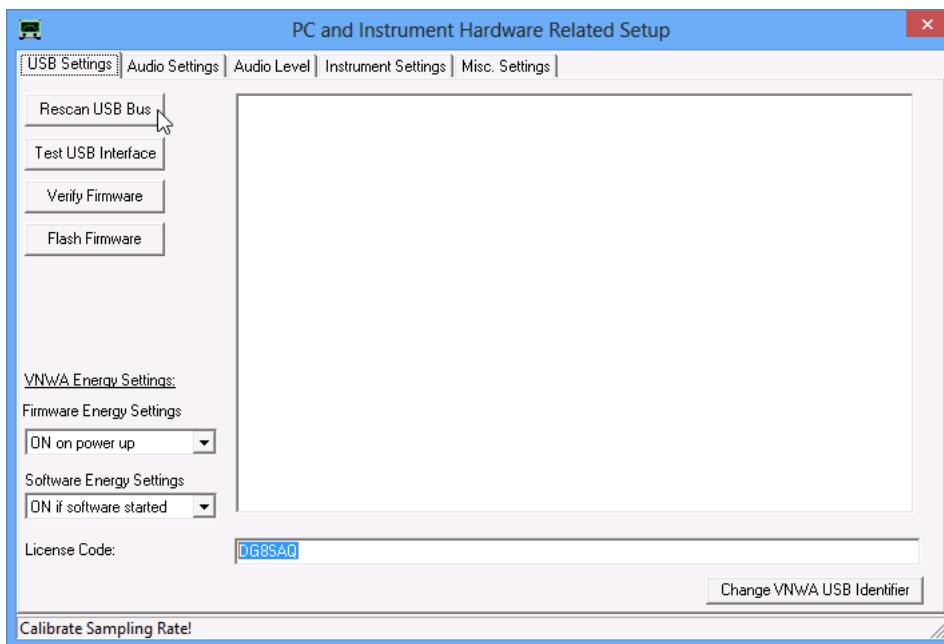
►►► **Note:** You can interrupt the guided setup at any point by not pressing the OK button or by closing the dialogs or the setup window with the X-button. Auto-Setup can be restarted by going to the "Audio Settings" tab and pressing the "Auto-Setup Audio Devices" button on the lower left.

If a license code had been entered previously during software installation, the next step will be skipped automatically and the setup process will continue in section 4.

If no license code has been entered previously, then you are asked to do so now:



Confirm the notification by pressing OK, then enter your license code...



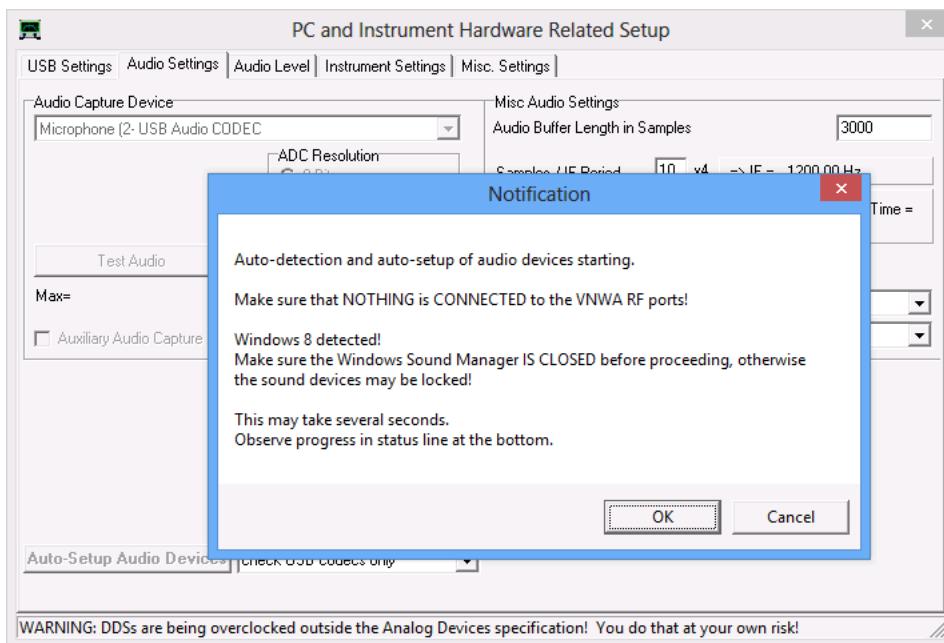
... and terminate by pressing the return keyboard key or by pressing the "Rescan USB Bus" button.
If the entered license key is incorrect, you will be asked again to enter a correct license key.

If the key is correct and connection to the VNWA hardware could be established, then the setup wizard proceeds with the automatic audio setup:

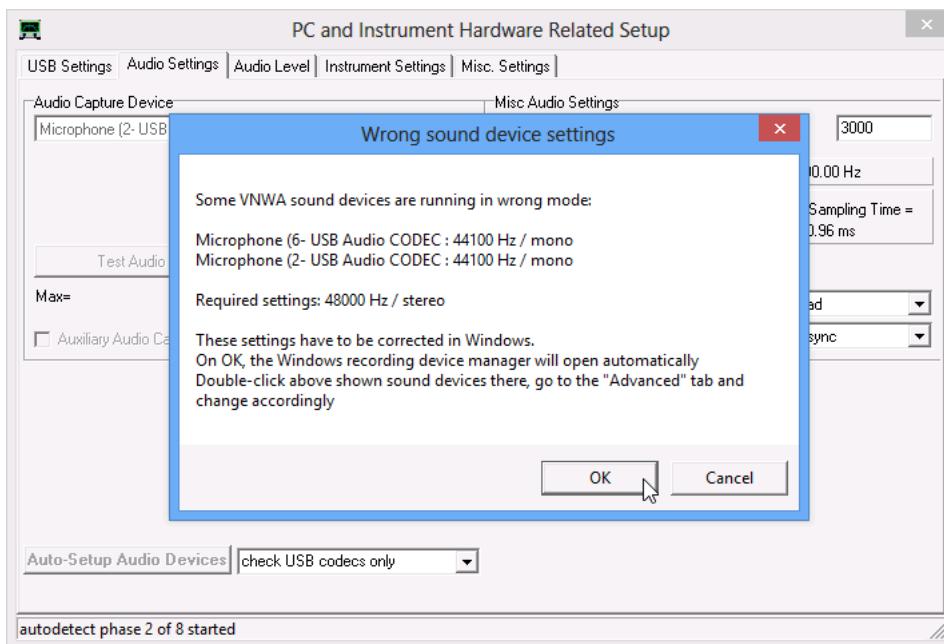
4. Audio devices setup

Make sure that nothing is connected to the VNWA TX and RX ports before proceeding with the auto-detection of audio devices.

Also, I have noticed that on some Windows 8 installations, sound devices cannot be opened while the Windows sound device control window is open. So make sure it is closed before you proceed.

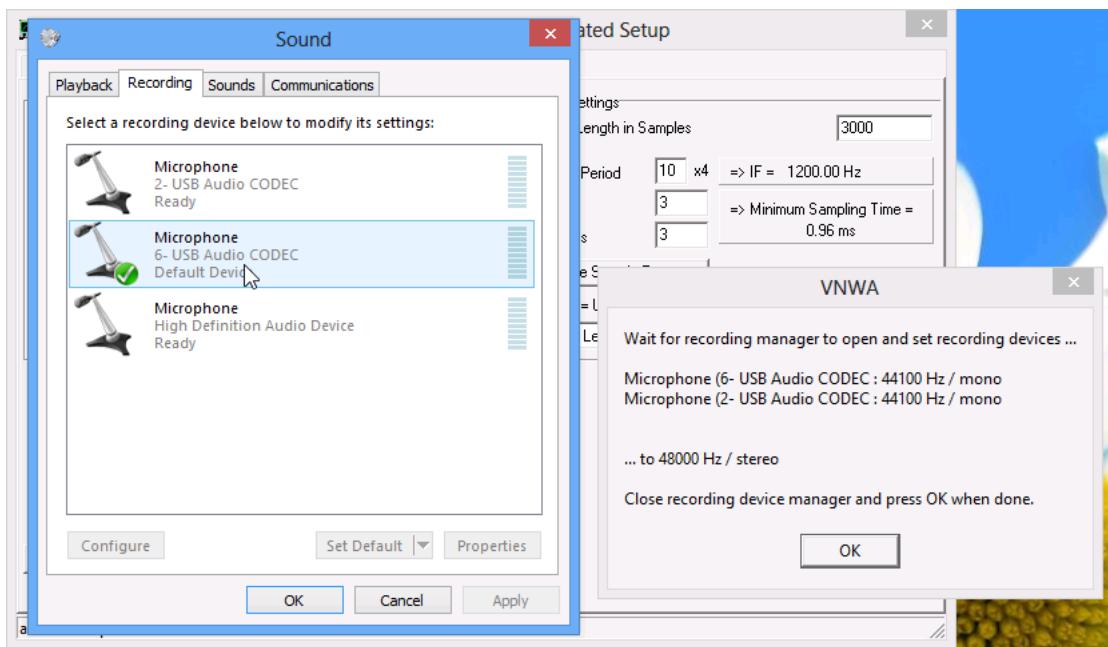


After having verified that nothing is connected to the VNWA RF ports, proceed by pressing OK.

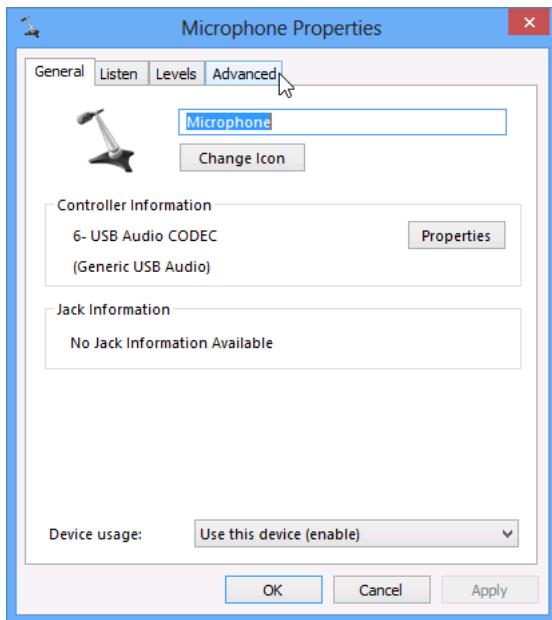


In this example I have connected a VNWA3E, which contains two sound devices. If a VNWA3 is being used, only one sound device will be detected.

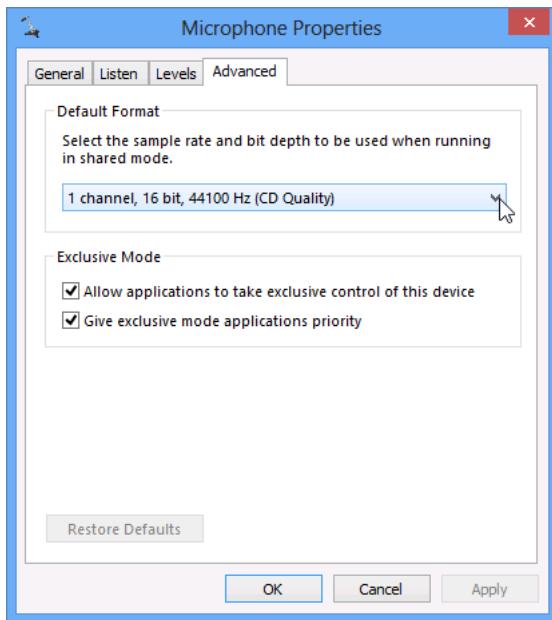
For Windows versions Vista and newer, the VNWA sound recording devices will be switched to 1 Channel / 44100 Hz upon first connection with the PC. The VNWA software will notice this and ask you to change it. As Windows does not allow user code to change these settings, you will be asked to change them inside the Windows **Sound manager** which will open automatically. Note, that this problem does not occur for Windows XP and older systems.
Press OK and watch another notification window and the Windows sound manager open:



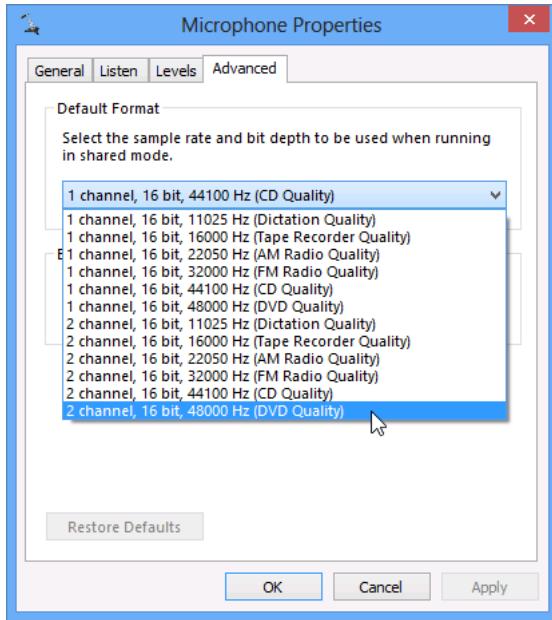
Double-click those sound devices in **Sound manager** which are shown in the notification window. By doing so, the Windows sound device **Properties manager** will open:



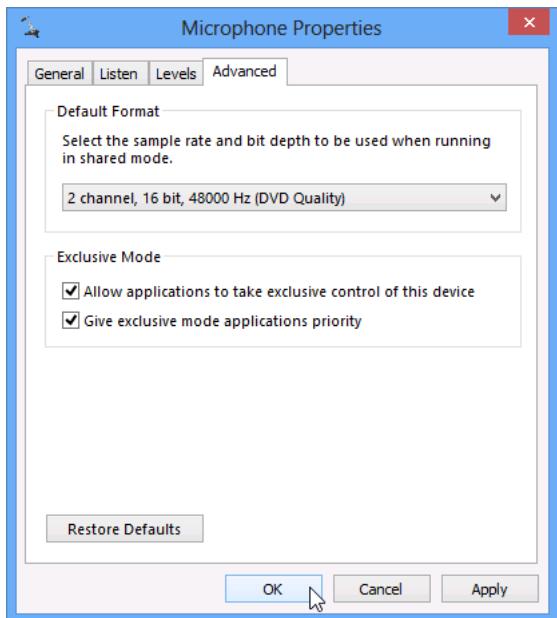
Go to the **Advanced** tab, ...



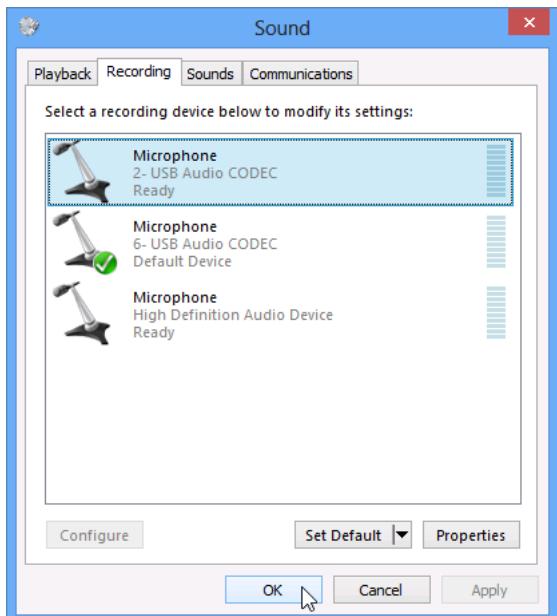
... and change from the current setting (1 channel 44100 Hz) ...



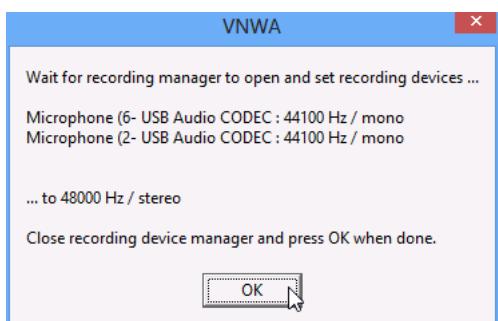
... to 2 channels 48000 Hz.



Press OK ...



... and repeat with the second sound device. Then close the Windows Sound devices manager by pressing OK...



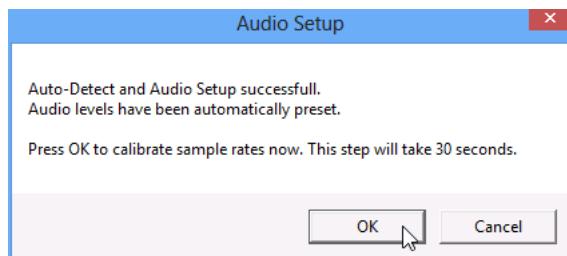
... and close above notification by pressing OK.

If you missed to set up all sound devices correctly or you selected a wrong setting, then the above process starts automatically again.

After all sound devices have been configured correctly, the auto detect process will start. Observe progress in the bottom status line of the setup window:

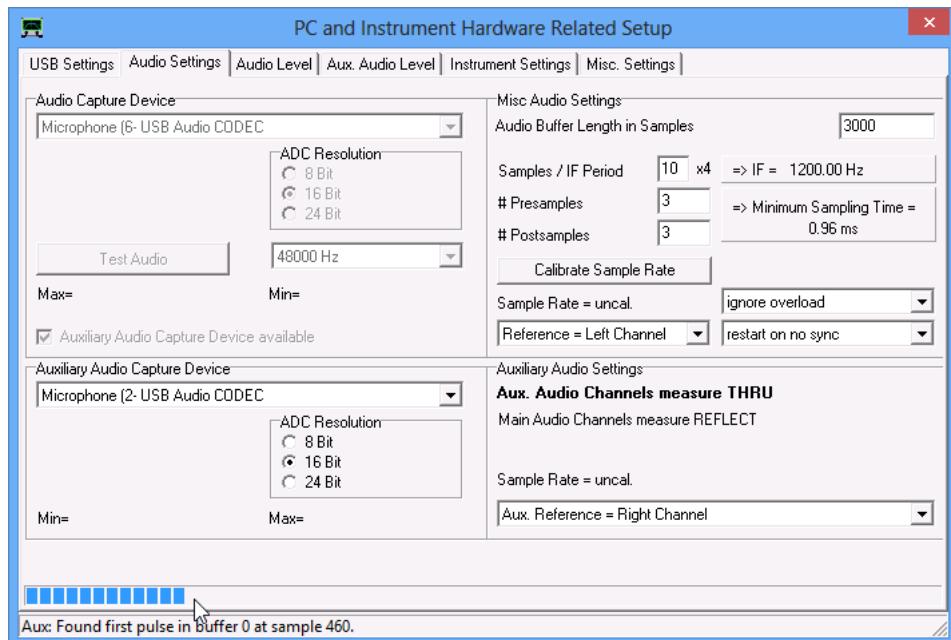


Successful completion of the auto-detect process ends with the automatic correction of the sound devices setup and a user notification:

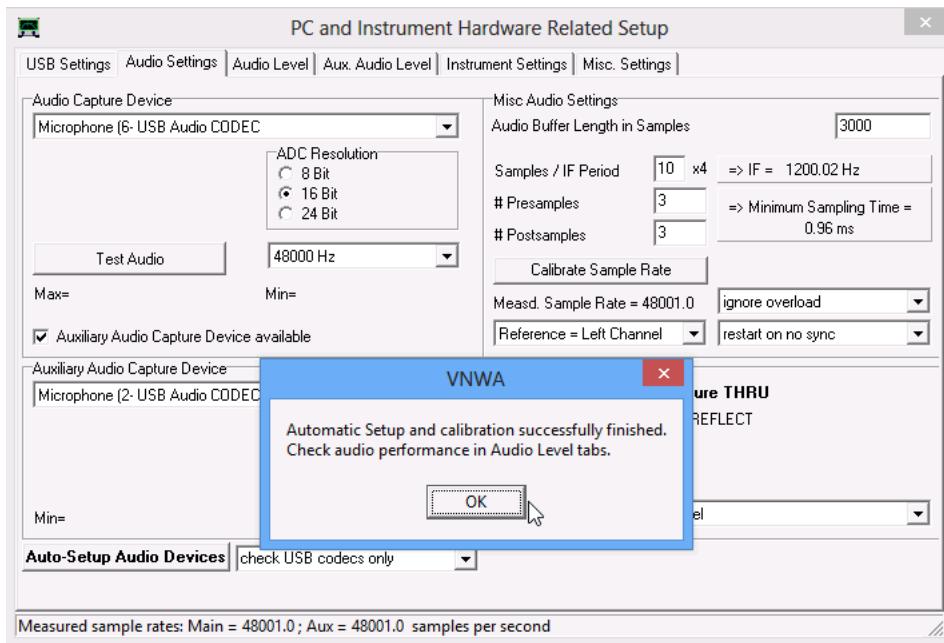


Note, that from VNWA software version V36.2.z on, the software is able to control the codec recording levels for Windows versions Vista and later, provided the codec interface library CodecIF.dll is properly installed (the automatic installer takes care of this, on manual installation, the file has to be copied to the program directory). At this stage, the levels have already been set to the correct values.

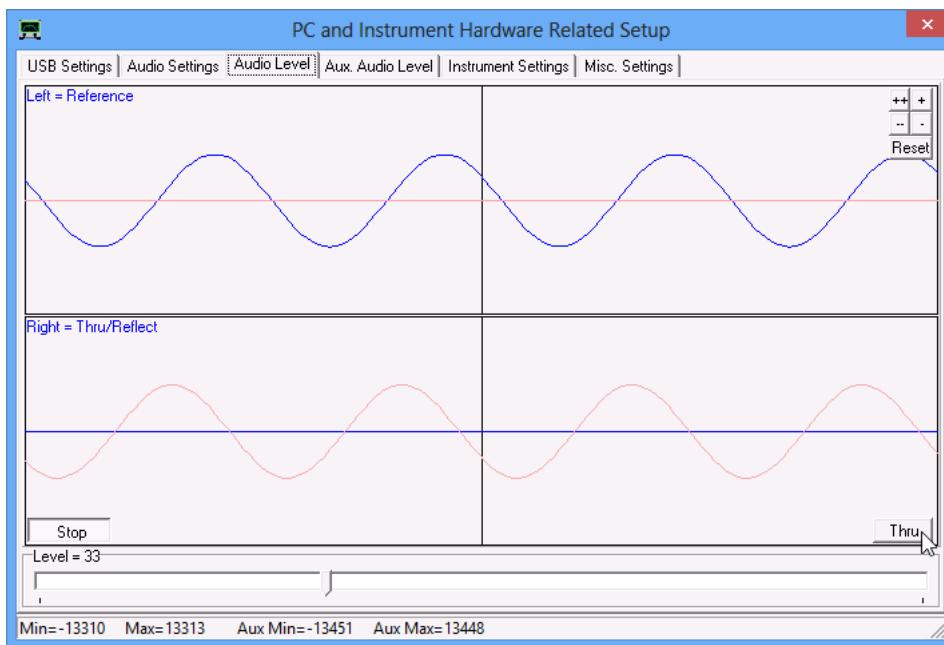
All that is left to do is to determine the exact codec sampling rates. This sample rate calibration step is started automatically after confirming by pressing the OK button. Observe the progress bar moving slowly to the right as the measurement proceeds:



After successful completion you will be notified. Please also observe the exact determined sample rates in the bottom status line of the setup window. If those values deviate very much from each other or from the nominal sample rate of 48000 Hz, then it is advisable to repeat this calibration step later manually by pressing the "Calibrate Sample Rate" button on the right side of the Audio Settings tab.



After closing the notification window by pressing OK, an audio test will be started and the software will switch automatically to the **Audio Level** tab:



Here you can see the VNWA signals as they reach the codec. With the first default settings you will see sine-waves which should show only a very slow phase shift over time. If they move rather fast, this indicates that the previous calibrate sample rate step has failed and should be repeated.

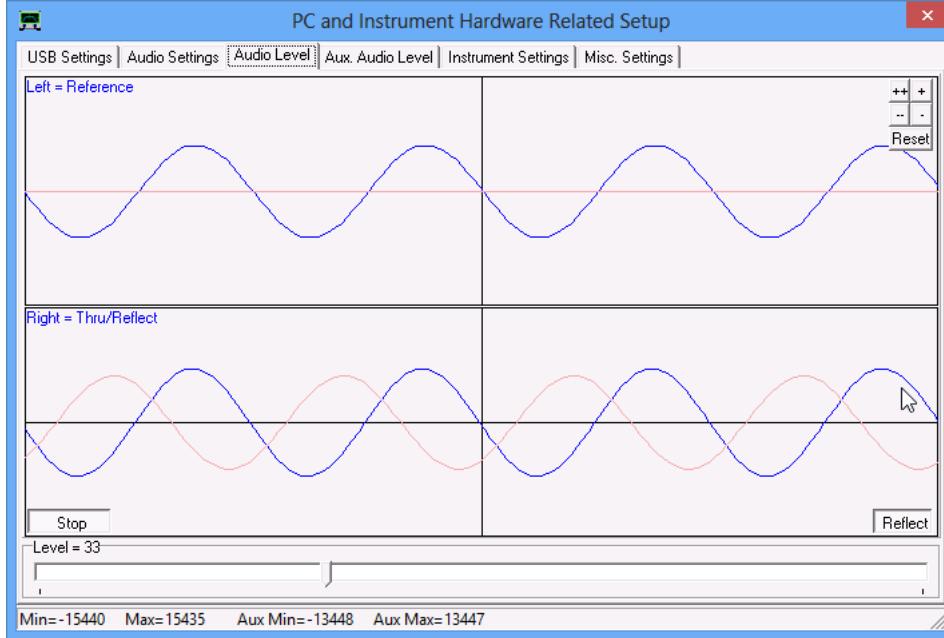
Note, that this audio test can be repeated manually later by going to the Audio Level tab and pressing the Start/Stop button on the lower left.

Note, that if you happen to have changed the number of sample points per IF period to a very low value, then the shape of the sine waves may become trapezoidal/triangular. This is perfectly ok.

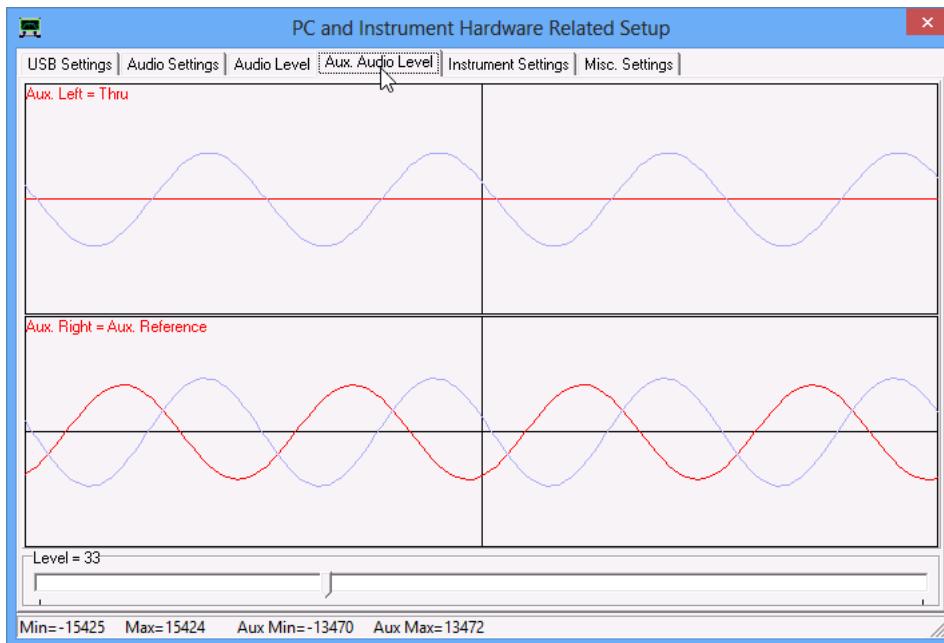
In above screenshot you do see the blue main reference signal and the faint red auxilliary reference signal (aux signal visible for VNWA3E only). The signal levels are preset to about 50% of full scale, which is the recommended setting. Note, that VNWA 36.2.z and later offers a single volume slider on the bottom, that controls the recording level of the main sound device for Windows versions Vista and later. Windows versions XP and earlier offer two sliders (separate for left and right channel) or none depending on availability of hardware level control. There is no thru-signal (red and blue flat lines) as nothing is connected to the VNWA RF ports.

When toggling the Thru-button to reflect mode by pressing it, a second blue sine wave representing the reflect signal

appears:

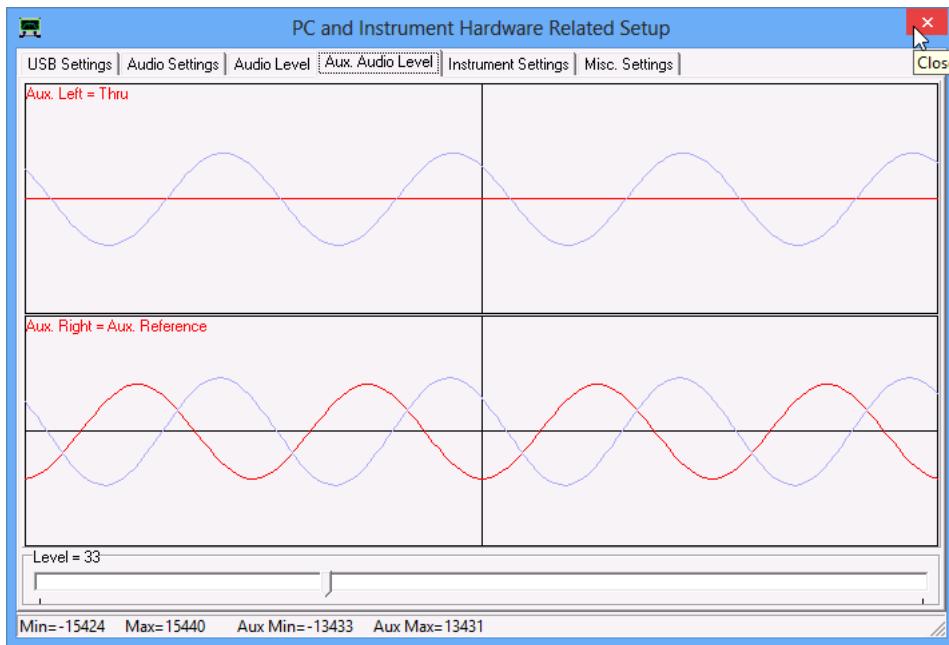


Note that the red aux signals remain unchanged. The level of the aux signal may be controlled by the volume slider on the bottom of the **Aux Audio Level** tab:



Manually changing levels should not be necessary, as auto-setup presets levels to recommended settings.

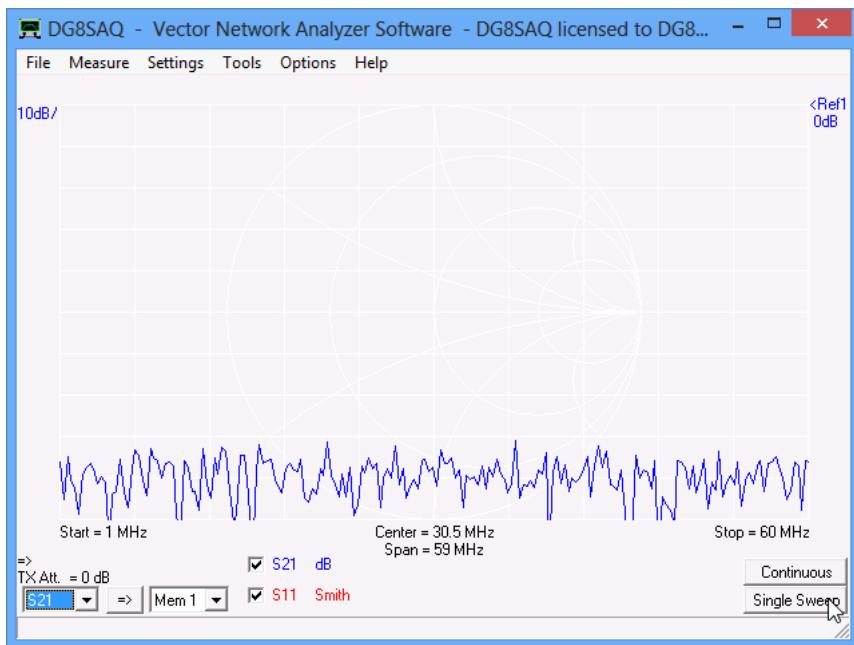
Now, stop the audio test by clicking the X-button once...



... and close the setup window by clicking the X-button again thus going back to the VNWA main window.

The instrument hardware setup is completed now and you are back on the VNWA main window. The instrument is ready to sweep.

Try it by pressing the **Single Sweep** button:



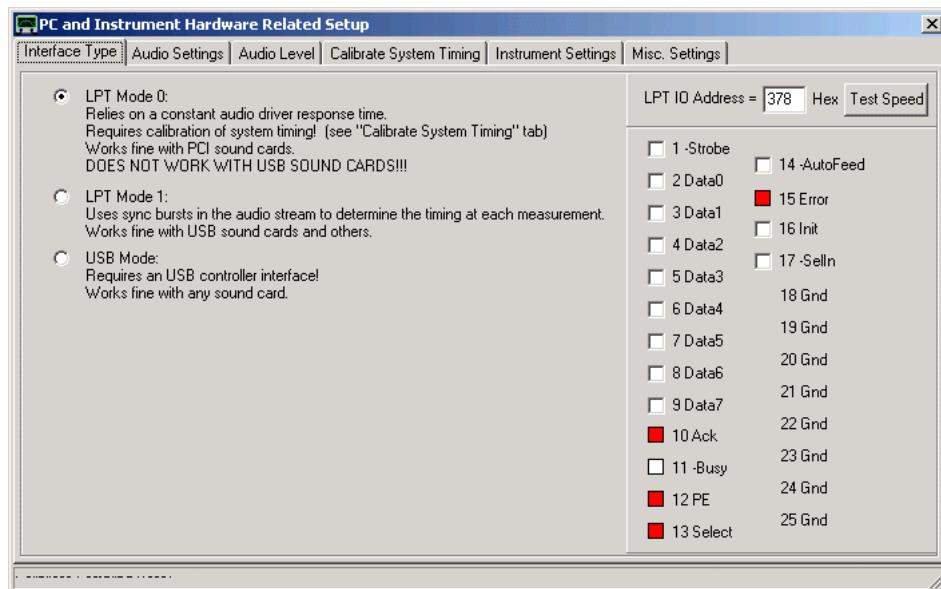
The S21 transmission measurement will show noise as there is no RX input signal yet. The S11 reflect signal is off screen as the instrument is not RF calibrated yet.

See page Instrument Calibration to learn about RF calibration.

MANUAL CONFIGURATION OF THE SOFTWARE FOR USAGE WITH A DG8SAQ VNWA

Starting the Software, Choice of Interface

To start the VNWA software, double-click VNWA.exe. Look here for special issues on Windows Vista.
On program start two windows should open. One of looks like this:



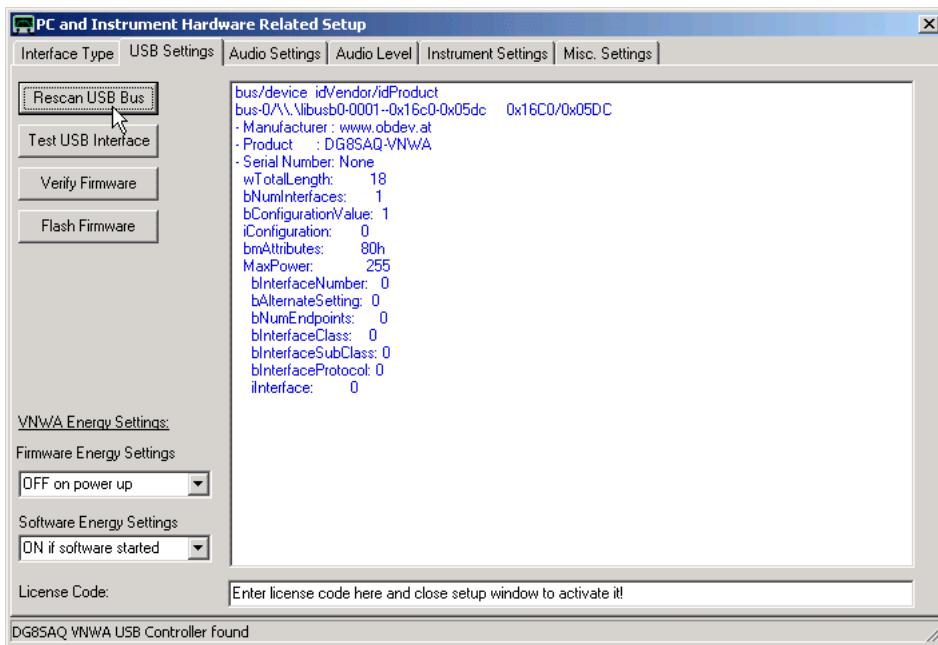
Check the desired interface type. Make sure to enter the correct address of your LPT port (378hex is the most common choice, see your PC BIOS). The button "Test LPT" allows you to evaluate the speed of your LPT port. You should obtain values between about 1us and 2us per byte. This is no functionality test of the LPT port hardware, though! You can actually test the LPT hardware with the feature to the right. You can toggle the states of the LPT lines and observe the changes with a scope. You can also make a loopback connector and observe the changes on the input lines.

►►► **Note:** USB to LPT converters won't work for this application due to latency of the USB transfer protocol.

Before you select *USB Mode*, the *USB_VNWA* interface must be plugged in!

If you select **USB Mode**, the LPT test features disappear and the **USB Settings tab** appears.

USB Settings (only applicable to the *USB_VNWA* interface)



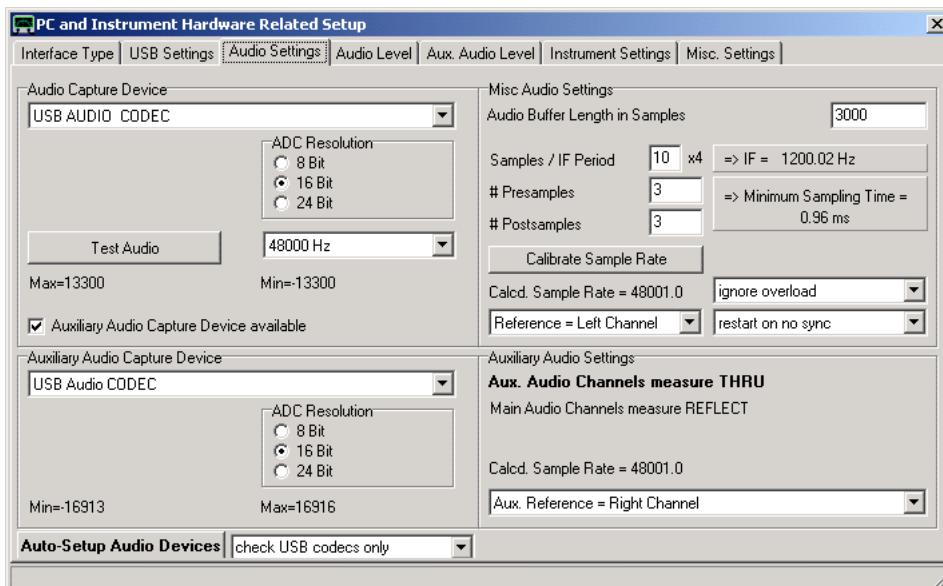
Press **Rescan USB Bus** to see if the USB driver and the USB_VNWA interface are being properly detected. You should see the above screen in this case.

►►► **Note:** Before continuing, you must enter the **license code** you have received together with your USB_VNWA interface into the bottom field. After this **you must close and reopen the hardware setup** window in order to activate the license key!

After activation you can press **Test USB Interface** to verify flawless communication.

Audio Settings

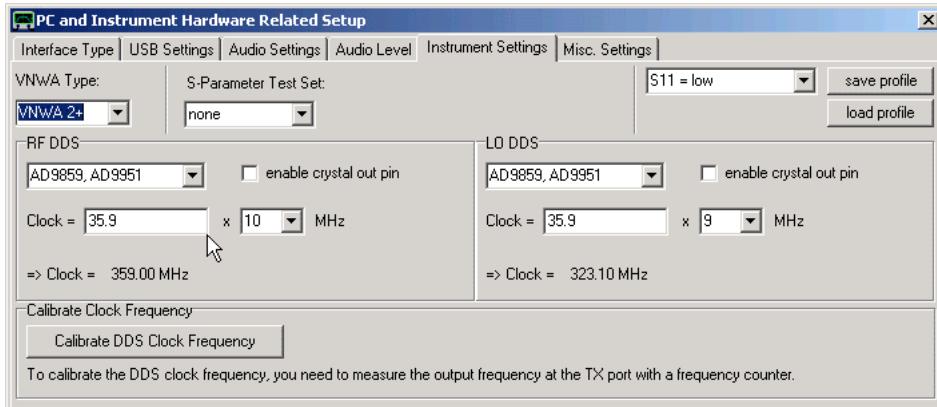
Next move on to the "**Audio Settings**" tab:



Select your preferred audio capture device on the left. 48kHz is a good choice of a sampling rate, which is supported by all modern sound cards. Lower sampling rates may be advantageous for very low frequency measurements, but limit the sweep speed. The indicated "Misc Audio Settings" are good choices. If you have a slow CPU you might want to increase the Audio Buffer Length. This will lead to fewer graphics updates.

Instrument Settings / External Hardware

Next move on a few tabs further to "Instrument Settings":

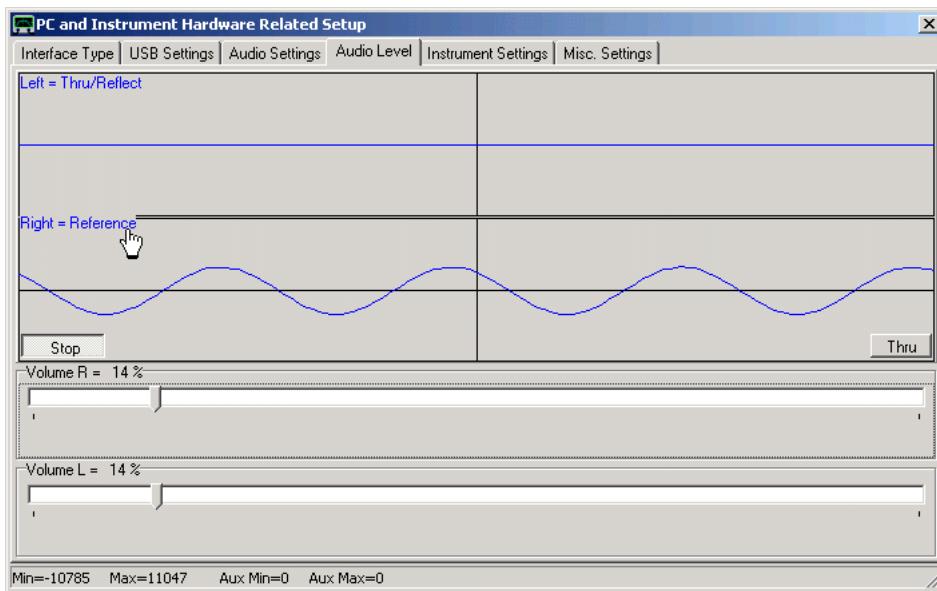


Here you can specify the hardware inside your VNWA. If you have a different crystal frequency, modify the data according to your hardware. The displayed frequency of 35.9 MHz is approximately obtained if a standard 12 MHz crystal is used in 3rd overtone mode. If you want to calibrate your DDS clock oscillator frequency, press the according button and follow the instructions. You need a frequency counter to do so. Note, that the two frequency multipliers need to be as high as possible and different from each other. If you want to use the frequency range 600 MHz...1.3 GHz, you need to select multipliers **auto**. When you do so, the prefilled auto clock multiplier table pops up. Simply close it as the default values should be ok. Now your PC is ready to send control data to your VNWA board and it is time to connect it to the PC, if you haven't done so already.

Note: With the displayed clock multiplier selections you stay within the Analog Devices DDS specification, but you can only measure up to about 300 MHz.

Adjusting the Audio Level (no adjustment available for the DG8SAQ USB_VNWA interface or for Windows Vista)

Move on to the "Audio Level" tab and press the button "Test Audio". You might want to increase the window size for better visibility. You should see the following screen:



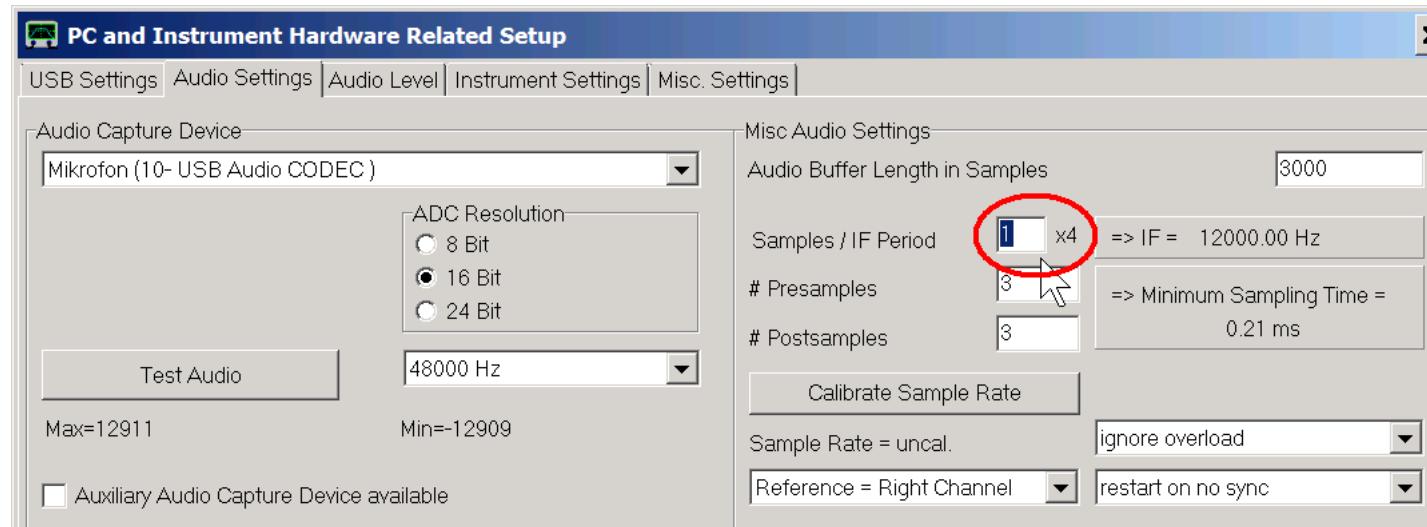
Left and right audio channels are monitored while the VNWA is set to a fixed frequency. On the above screen, you can see that the reference signal arrives on the right channel (bottom curve, if it's the left channel in your case, make sure to change the setting in the Audio Settings/right block accordingly). Set the volume slider for the reference channel such, that by no means clipping occurs. A good rule of thumb is to set the volumes such, that the amplitude spans 50% of the maximum amplitude. The left channel (top) shows a flat line, as the Thru signal is zero while the VNWA Tx and Rx ports are disconnected from each other. Press the right lower button showing "Thru". It will change to display reflect. At the same time, you'll see the Reflect signal on the left channel (top). Adjust the level with the according slider as for the reference signal. If you connect a short or a load resistor to the TX port, you can observe how the amplitude and the phase of the reflect signal changes. If you now connect the TX port with the RX port via a coaxial cable, the reflect

signal should disappear, as the 50 Ohms input resistance of the RX port does not reflect RF power. If you press the bottom right button again to show "Thru", you should see a strong thru signal instead. If its amplitude appears too large, reduce it with the according slider. Don't increase the amplitude at this time, as you would also increase the reflect signal amplitude, that you have already adjusted.

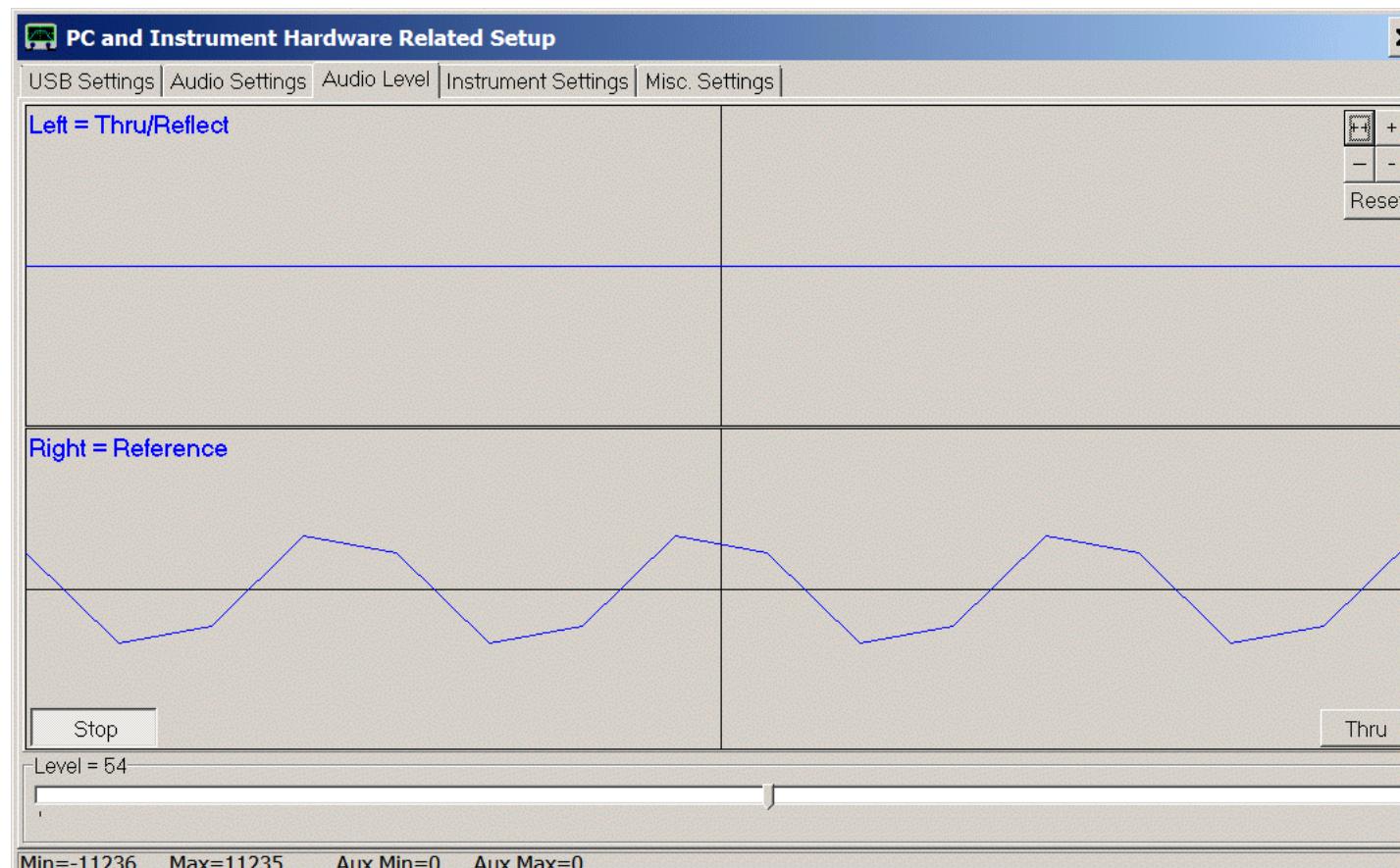
Note: No sliders will be available for USB codecs on Windows XP or older. A single slider will be available for on Windows Vista or later for VNWA software version 36.2.z and newer!

If you are done, press the "**Test Audio**" button again to stop sampling. Note, that the other tabs are deactivated while the sound device is capturing! Also, you cannot close the setup window while sampling.

Note, that the sine wave in above screen shot is sampled with the default setting of 10x4 samples per IF period, which makes it look smooth and sine-like. If you reduce the number of samples per IF period to 1x4 as is required for ultrafast sweeping ...



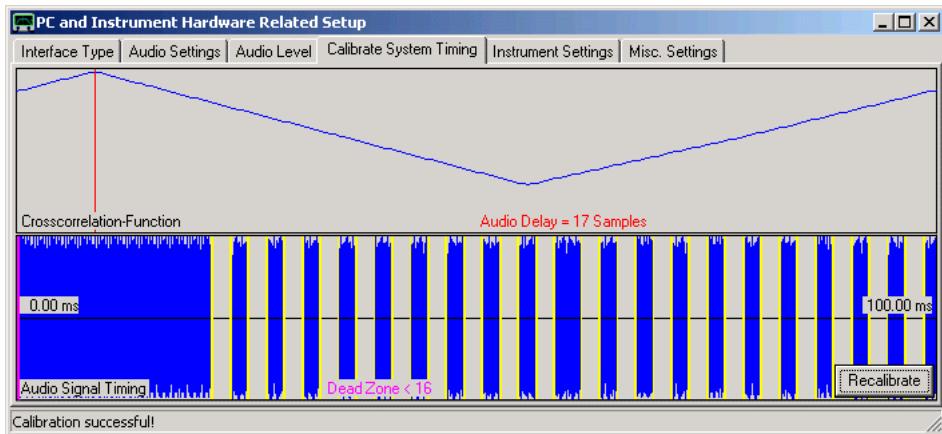
... the sine wave will rather look triangular or trapezoidal ...



... which is perfectly ok! It is still a sine wave which is approximated in the graph with 4 straight lines connecting the 4 sampled data points per period. Note that the last two screenshots were taken on a Windows7 machine with a VNWA2.

Audio Driver Timing Calibration (LPT Mode0 only) and Sample Rate Calibration

If you have selected interface type "LPT Mode 0" in the beginning, you need to calibrate the sound driver latency next. In this case move on to tab "**Calibrate System Timing**" (for "LPT Mode 1" you can skip this step). Press the button "Recalibrate". You should see the following picture:



Here, the DDSes are switched on and off and the data stream is analyzed to recover the switching moments. The vertical yellow lines should exactly match the changes in the blue audio stream. The cross correlation function (top) should have a sharp maximum. In the above example, an audio delay of 17 samples is determined. Press the recalibrate button a few times and observe the calculated audio delays. If they are constant to within about +/- 1 sample, then this operation mode is feasible for your PC. If not, then you need to run "LPT Mode 1". The latter is the case for all USB audio devices.

Next, you need to calibrate the sound card sampling rate. Move back to the tab "**Audio Settings**" and press the button "Calibrate Sample Rate" in the right lower half of the window. A blue progress bar should appear for about 30 seconds. If the calibration is successful, you should see the determined sampling rate just below the button. It should be very close to the one specified in the according selection box. The VNWA must be connected and running in order to perform this calibration.

Master Calibration Filename and Debug Settings

You may want to inspect the last tab "**Misc. Settings**".

Here, you may enter a filename, which is used to store an instrument master calibration, which will be loaded and interpolated automatically, in case no calibration for the actual sweep settings has been performed.

DO NOT TOUCH THE DEBUG SETTINGS UNLESS YOU EXACTLY KNOW WHAT YOU ARE DOING! All debug settings need to be unchecked for proper VNWA operation.

Now close the setup window and close the main window by pressing the buttons. Closing the main window will cause the entered data to be stored in ini-files, which you will now find in your VNWA programm directory.



This setup is required only once at first program start. On the subsequent program starts, the necessary data will be read from these ini-files and the setup menu will not be entered automatically any more. But you can still enter and modify the setup manually via the menu "Options-Setup".

CONFIGURING AND RUNNING THE VNWA APPLICATION UNDER WINDOWS VISTA, 7, 8 OR 10

The following procedures apply to Vista, Windows7, Windows8 and Windows10 32 bit and 64 bit operating systems. For 64 bit operating systems a special set of USB-driver is required. The automatic software installer takes care of this automatically. If manual driver installation is attempted, consult the "**Driver Compatibility Tables and Driver Installation**" page for details.

►►► **Help system note:** When you see this text on a Vista or Windows7/8/10 machine, you have already figured out how to **configure your OS to support old helpfile formats**. By default Vista and Windows 7,8 or 10 do NOT support *.hlp help files.

►►► **LPT note:** In order to use LPT mode on 32 bit Vista or Windows 7, you must run the VNWA application with administrator rights. See here for details. Note, that you **cannot use LPT mode on 64 bit OSes**, as the LPT drivers used by VNWA are not digitally signed by Microsoft.

►►► **USB audio codec note:** If you use VNWA3 or the built in audio codec of the **DG8SAQ USB_VNWA interface**, you must **manually configure the audio codec for stereo operation**.

VISTA / WINDOWS 7 / WINDOWS 8 / WINDOWS 10 HELP FILE ISSUES

Vista Help File Fix

In order to view this helpfile on a Vista machine, you need to download and install the appropriate **WinHlp32.exe** for your operating system.

See <http://support.microsoft.com/kb/917607> for details!

Windows7 Help File Fix

Finally, Microsoft has provided a fix for Windows7 also to support this help file format. In order to view this helpfile on a Windows7 machine, you need to download and install the appropriate **WinHlp32.exe** for your operating system.

See **Windows Help program (WinHlp32.exe) for Windows 7 for details!**

An alternate workaround is possible: You need to get the file **WinHlp32.exe** from a WindowsXP machine and copy it to your VNWA software directory. Next, modify VNWA.ini with a text editor by changing the line "HelpByExe=0" to "HelpByExe=1".

This done, the helpfile will open normally from within VNWA.

This fix will also work for Vista in combination with VNWA, but it will not fix help file problems for third party software.

Windows 8 Help File Fix

In order to view this helpfile on a Windows 8 machine, you need to download and install the appropriate **WinHlp32.exe** for your operating system.

See <http://www.microsoft.com/en-us/download/details.aspx?id=35449> for details!

This fix is tested to work on Windows8 Enterprise, but is reported to fail on Windows 8 Professional. A workaround is to manually copy winhlp32.exe from a Windows XP system to the VNWA installation directory. VNWA36.2 and later will automatically detect the program and use this instead.

Windows 10 Help File Fix

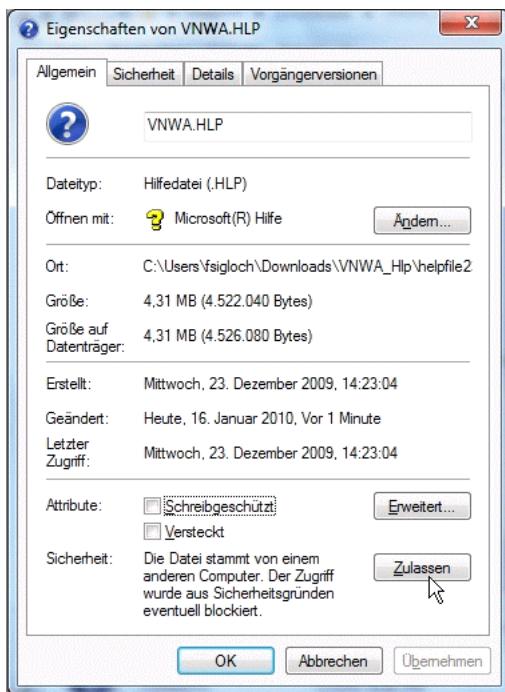
Todate, Microsoft no longer officially supports the *.hlp help file format, even though the Microsoft help file viewer *winhlp32.exe* still can run under Windows 10, but Microsoft actively blocks installation of this viewing tool.

There is a simple workaround, though. The Microsoft installer script *Install.cmd* for *winhlp32* can be modified such that the installer thinks the OS is Windows7 instead of Windows10 by modifying a line to "set WindowsVersion=7". This procedure is described in detail in Funkamateur magazine, 10/2016 on page 945. Patched installation files can be downloaded from there: https://www.funkamateur.de/tl_files/downloads/hefte/2016/WinHelpWin10.zip.

Alternatively, the open source program *viewhlp.exe* can be used. Installation procedures are described in chapter HELP - CONFIGURE HELP.

Vista and Windows7/8 Security Settings to allow to open VNWA.hlp

Even after having installed the Microsoft help fix you might not be able to open the helpfile with WinHlp32. In this case, the help file is blocked by Windows for security reasons. In order to qualify VNWA.hlp as safe, **right-click VNWA.hlp** and select **properties**. You will see the following window.



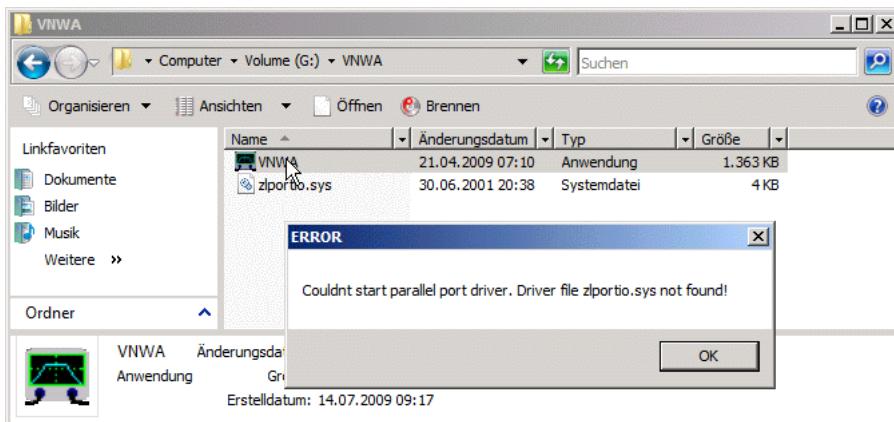
At the very bottom you see the item **Security** and a statement telling you file access is being blocked. **Press the release-button** right next to it (see mouse pointer).

You might have to repeat this step with the accompanying content file **VNWA.cnt**.

VISTA / WINDOWS 7 LPT MODE ISSUES

LPT mode for Vista or Windows 7 (32 bit only!)

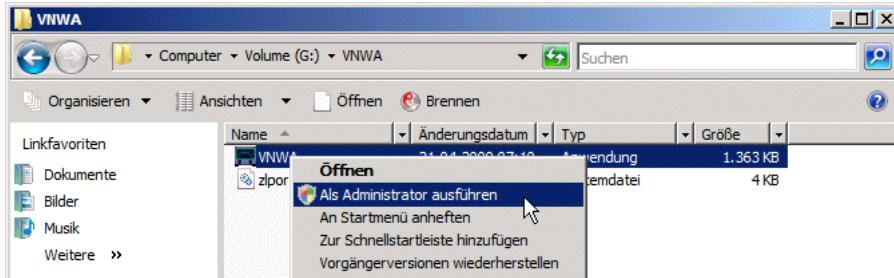
When the VNWA software is used in combination with the LPT driver **zlpportio.sys** on a Windows Vista or Windows7 machine, then **by default Windows blocks the port driver** and upon double-clicking VNWA.exe, this is what you will see:



There are two solutions to this problem:

Solution 1:

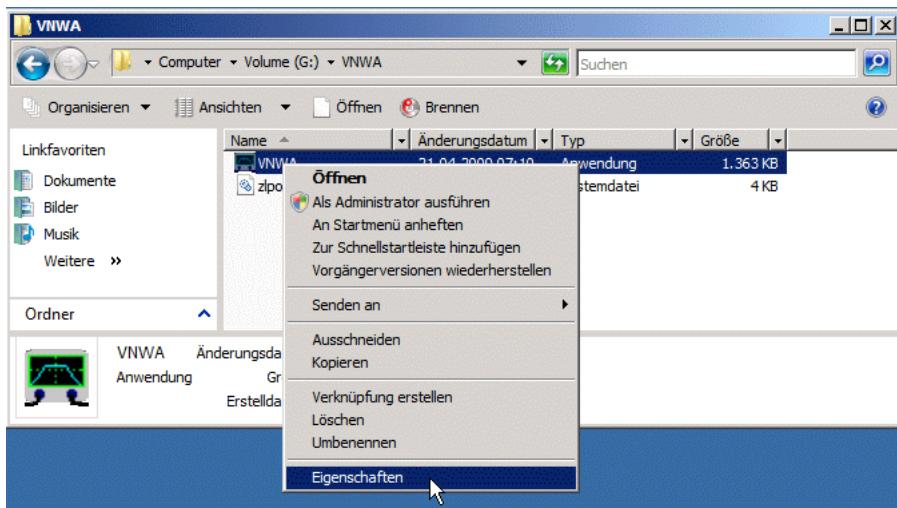
Right-click VNWA.exe and select "run as administrator":



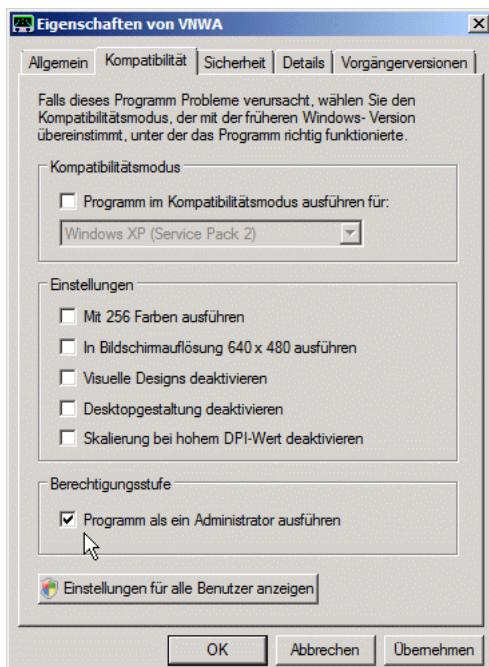
Confirm the warning popping up by agreeing to start the program. Then VNWA.exe should be started flawlessly with LPT support. Note, that you have to do this upon every program start. More convenient is solution 2:

Solution 2:

Right-click VNWA.exe and select the menu item "properties":



Select the tab-sheet "**compatibility**" and activate "**run program as administrator**". Do **not** select any compatibility mode! Confirm with "**ok**".



From now on you can double-click VNWA.exe to run it with administrator rights. A warning window will still pop up upon every program start unless you deactivate it inside Windows.

VISTA / WINDOWS 7 / WINDOWS 8 AUDIO ISSUES

►►► **Warning:** In Windows 7, 8 and Vista the **name of the USB Audio Codec depends on which USB port you plug the VNWA into.** This means, if you do below configuration with the VNWA connected to one specific USB port, you have to redo it again if you connect the VNWA to a different USB port. Moreover, the VNWA software will try to autodetect the USB audio codec, when you change connection from one USB port to another. This works reliably if you have only one USB audio codec connected. Otherwise, you probably have to manually select the correct sound codec in "setup - audio settings" after changing the USB port.

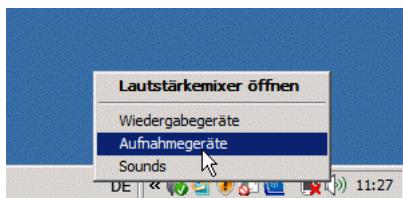
Summarizing, it is a good idea to **connect the VNWA always to one and the same USB port.**

Configuring the USB Audio Codec in Vista, Windows 7 and Windows 8

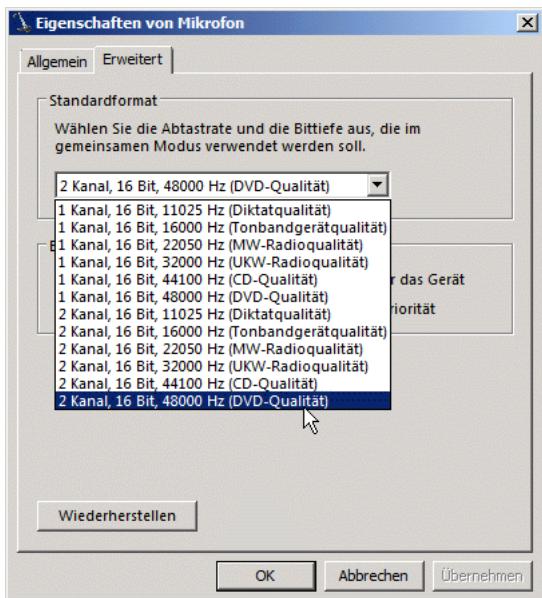
Note, that some people have reported they had to run VNWA/USB in administrator mode as described before for the LPT mode in order to make it work. This is not generally the case but might be worth a try in case of problems.

Configuring the audio codec:

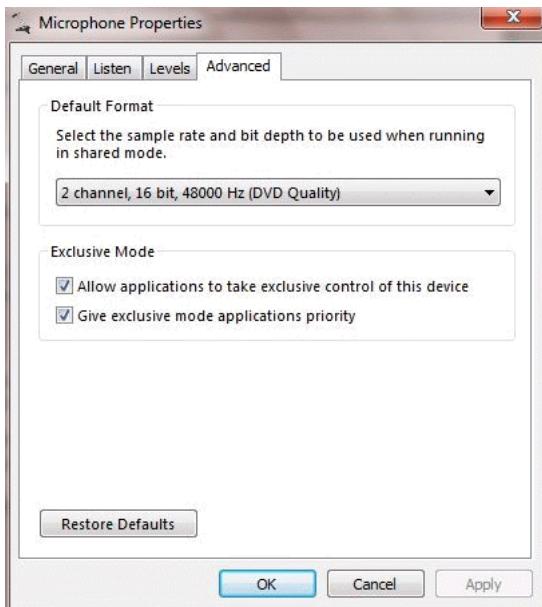
1. Right-click the little loudspeaker icon on the lower right Windows bar of your desktop and select "**capture devices**" from the popup menu:



2. Highlight the USB audio codec by clicking on it as seen above and press the "**properties**" button:



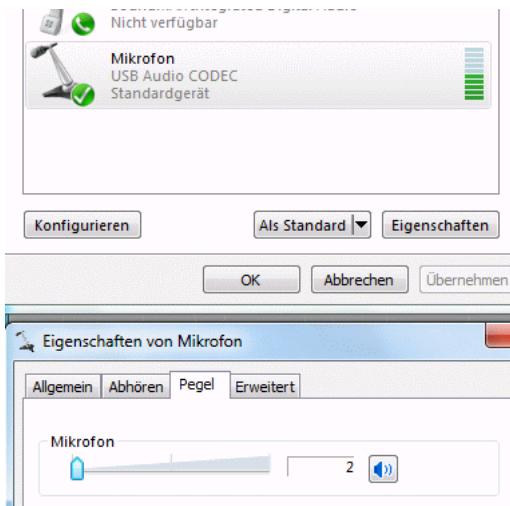
3. Select **2 channel 16 bit 48000Hz** as operation mode for the codec as seen above.



4. Confirm by pressing the **ok-button**.

[Windows 7 / Windows 8 Volume Control](#)

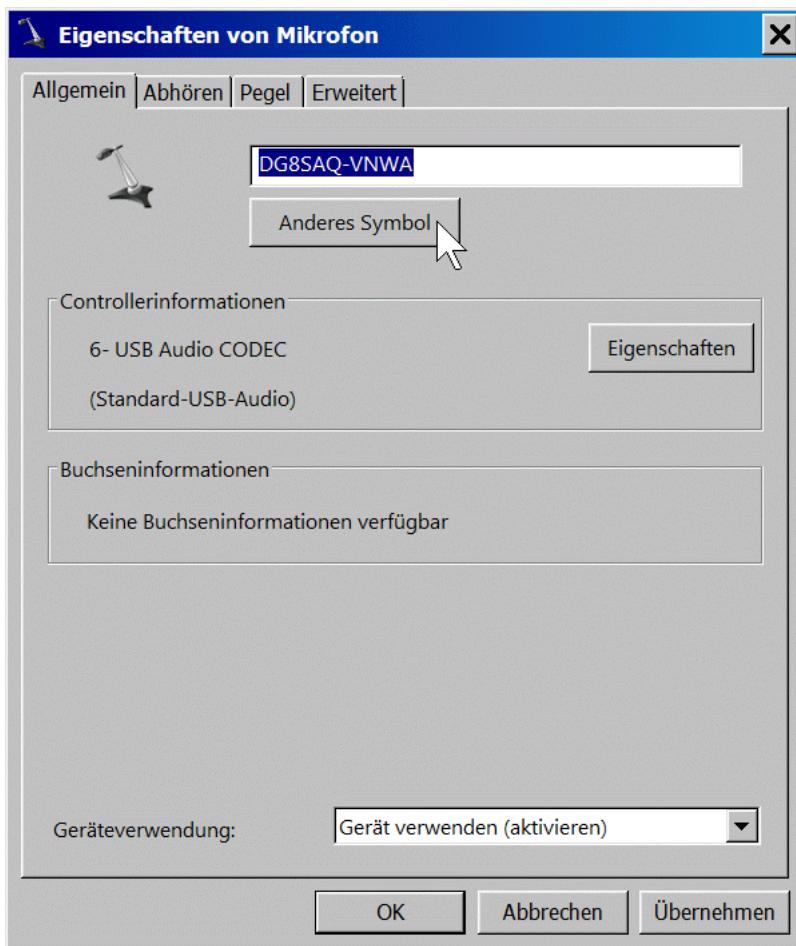
In contrast to all other supported operating systems, **Windows 7 and 8 allow to control the recording volume of the USB codec**. Note, that this is only a software control, which allows to multiply the sample data that the codec is sending by an amplification value. But apparently, numerical clipping will occur. By default, the volume setting is reportedly too sensitive, so reduce the volume appropriately by highlighting the microphone device of the USB codec, pressing the "properties" button and selecting the "level" tab.



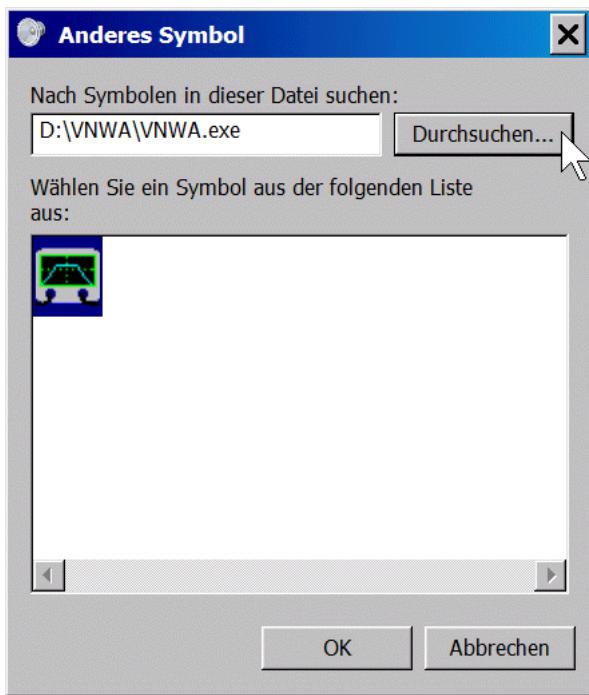
Levels between 2 and 4 are reported to work ok.

Not necessary but nice: Customizing the VNWA Audio Device in Windows 7 / 8

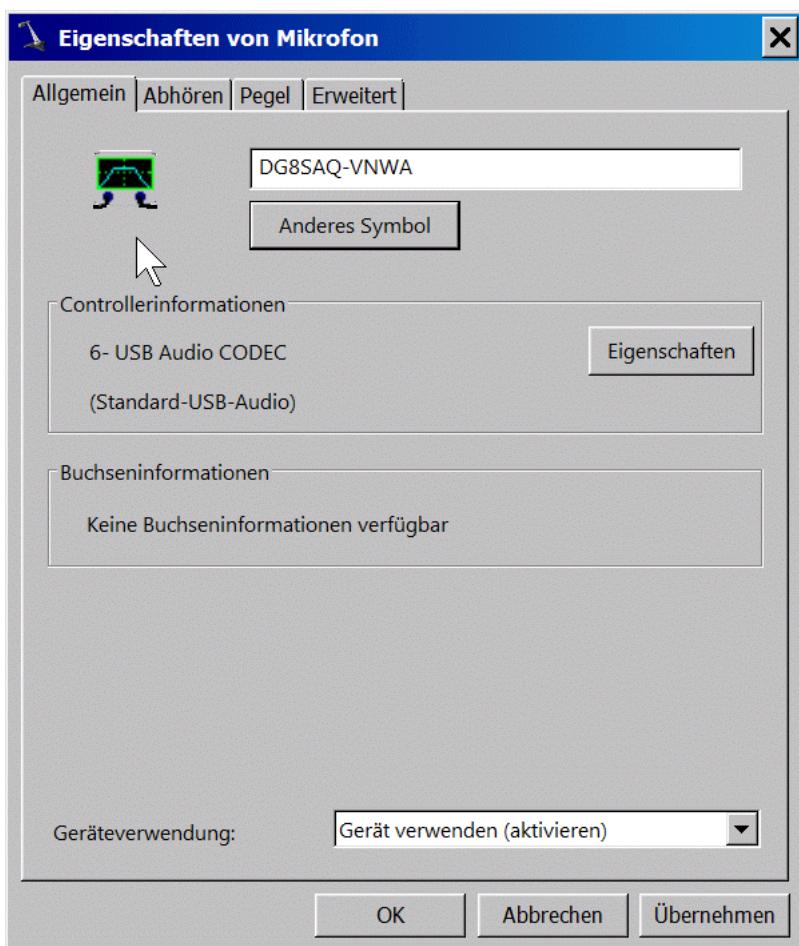
Windows 7 / 8 offers the possibility to change the name and symbol/icon of the USB audio device:



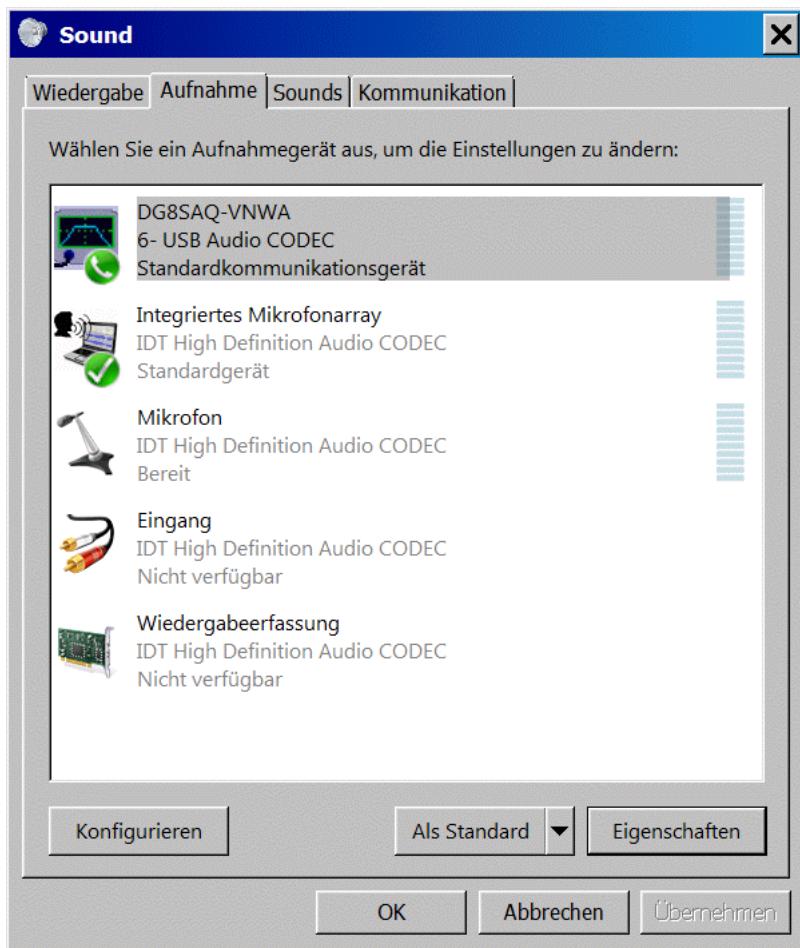
To change the name, simply replace the original name by something more descriptive like "DG8SAQ-VNWA" as shown in above example. The icon can be exchanged by pressing the button "other icon" (see mouse pointer in above screen shot). After pressing the button, a file selection menu opens. Use it to select the VNWA application file (VNWA.exe):



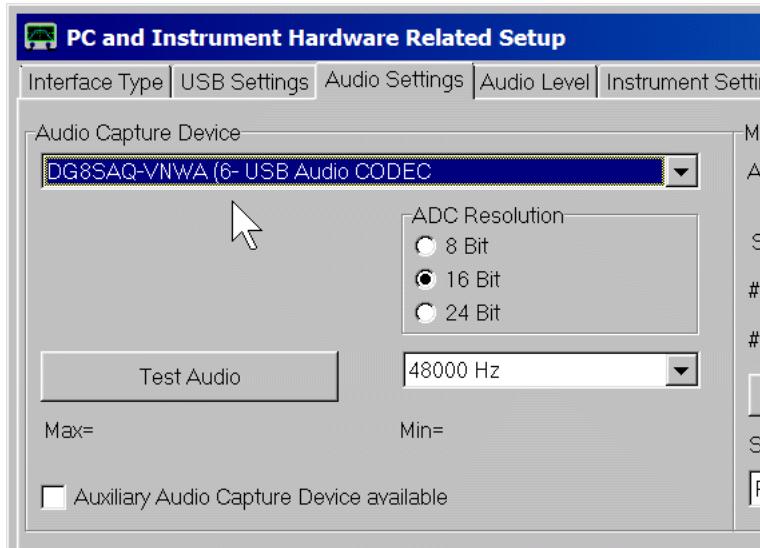
Confirm with "OK" and the USB audio input device will show with the VNWA icon:



Press the "Accept" button and then "OK" and the new name and icon are saved:



From now on you will see the VNWA USB codec with its new name in the VNWA audio setup:



►►► **Note:** The new name and icon is linked to the VNWA USB codec AND the specific USB port it is connected to. If you reconnect your VNWA to a different USB port, it will show with a different name and icon. Connecting it back to the original port, it will show the freshly installed name and icon, though.

=> **Always connect your VNWA to the same USB port. If you change the port, you will have to redo the VNWA audio settings AND the Windows7 audio settings!**

CONFIGURING THE SOFTWARE FOR USE WITH AN N2PK VNA

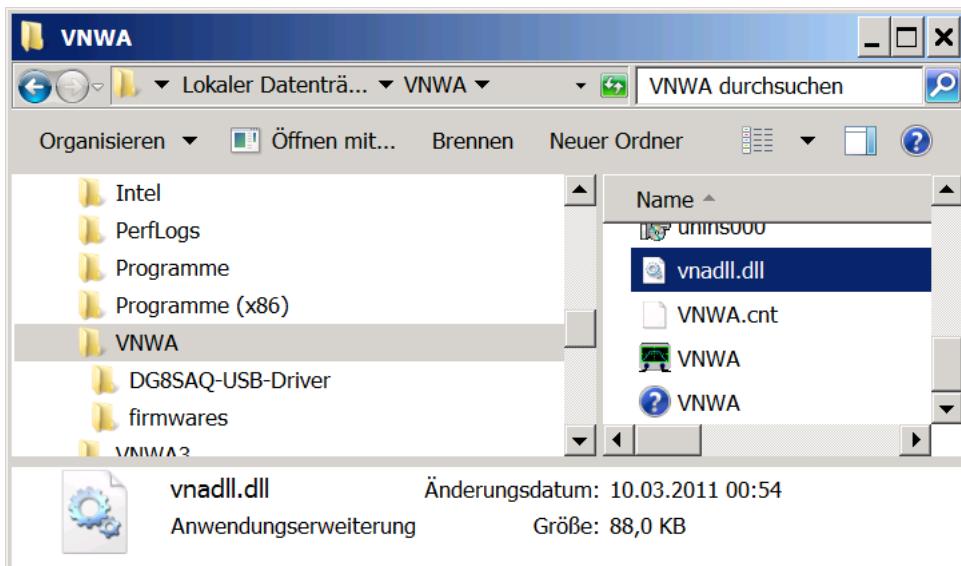
Activating N2PK Support with USB

Important:

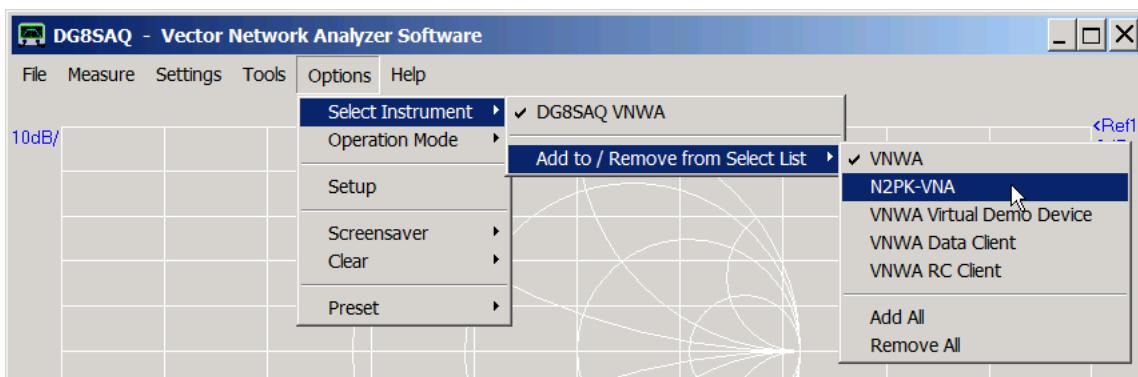
If you want to use the G8KBB USB interface mode, make sure that the files **vnadll.dll** and **delphivna.dll** are present in the VNWA software directory prior to first program start.

- **delphivna.dll** will automatically be provided when using the automatic software installer to install the VNWA software.
- **vnadll.dll** is maintained by Dave Roberts G8KBB and is subject to changes to support N2PK-VNA firmware updates. Always obtain the latest version of vnadll.dll from Dave's webpage <http://www.g8kbb.co.uk/>. Go to his download page and download the latest USB driver zip file. You'll find vnadll.dll in there.

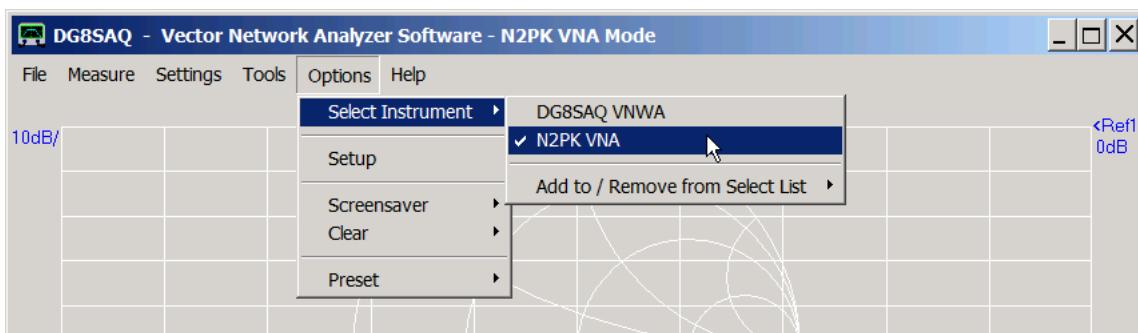
From VNWA software version 36.3 on, the simplest way to activate N2PK support is to copy the file **vnadll.dll** obtained from Dave Robert's webpage (see above) into the the VNWA installation directory:



The automatic software installer has already installed **delphivna.dll** into the same directory. Now, after launching the VNWA software, N2PK mode can be added to the select list ...



... and selected:



Alternative way to activate N2PK support

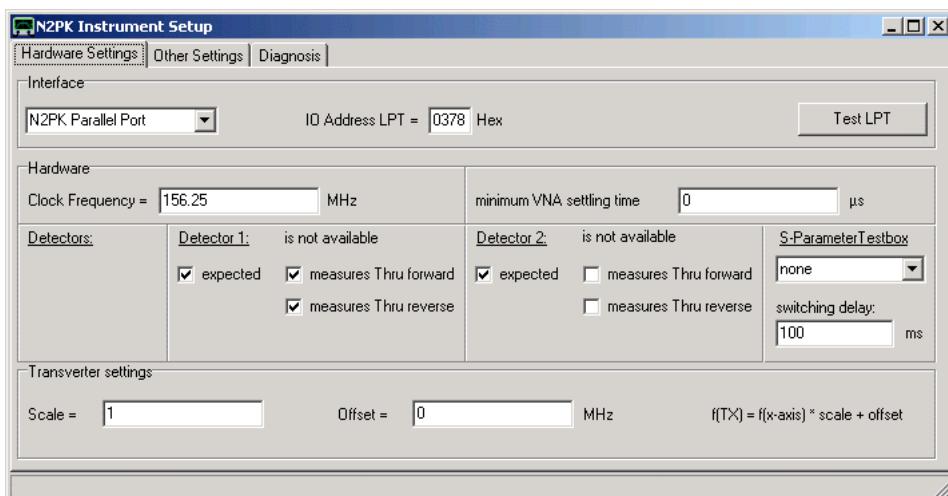
An alternative approach to activate N2PK support, is to create file named **N2PK.ini** with arbitrary contents in the program folder prior to program start.

You can do so by e.g. typing `echo > n2pk.ini` on the console prompt. This approach is only necessary if you run your N2PK VNA with LPT interface and you don't want to provide the USB controller `vnadll.dll`.

Activating N2PK Mode

Connect your N2PK VNA to your PC via USB or parallel port interface and power it up. To activate N2PK mode, select from the main menu "Options" , "Select Instrument" and "N2PK VNA". The software will search for the N2PK hardware, but likely it will not find it as we haven't provided setup data yet. If the software claims that it hasn't found detectors, close the notification. After having done so, the N2PK setup window will automatically open:

N2PK Setup



Enter the required data. If you use the LPT interface mode, make sure to enter the correct LPT address. Note, that LPT mode is not supported on 64 bit Windows systems.

In the "Other Settings" tab, you can specify a master calibration filename, which will be automatically loaded upon program start and which will be used to interpolate corrections in case no calibration has been performed.

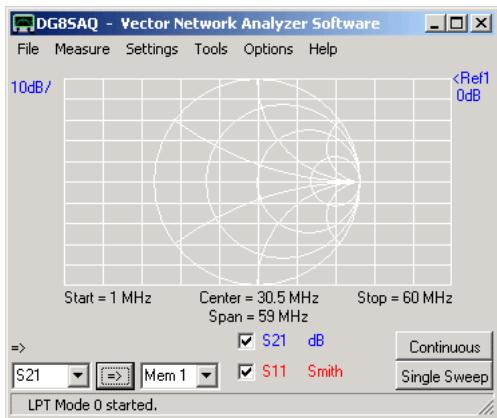
You can use the functions in the "Diagnosis" tab to check proper operation and communication between PC and N2PK hardware.

If done, close the setup window and close the main window by pressing the **X** buttons. Closing the main window will cause the entered data to be stored in ini-files, which you will now find in your VNWA programm directory.

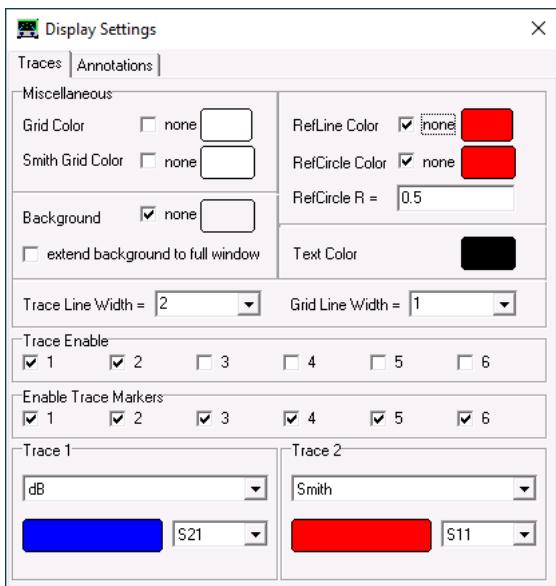
This setup is required only once at first program start. On the subsequent program starts, the data will be read from these ini-files and the setup menu will not be entered automatically any more. But you can still enter and modify the setup manually via the menu "Options-Setup".

DISPLAY SETTINGS AND MEASUREMENT

After having gone through the hardware setup procedure after first program start, and after having **restarted the VNWA software**, you should see the following screen, which I subsequently call the main graphics window:



Select the menu item "**Settings**"-"**Diagrams**"-"**Display**". You'll see the following window:



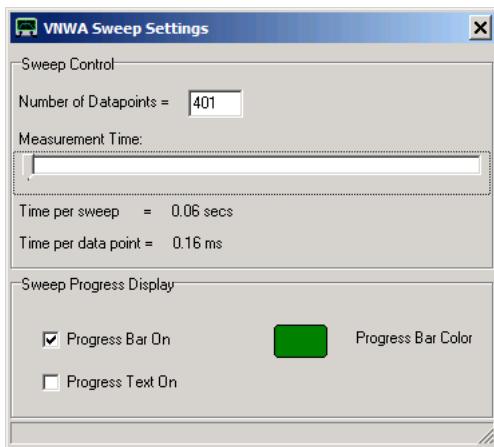
Here, you can customize colors, line widths and select, which traces (=curves, maximum 6) and types of traces to be displayed.

To save CPU time it is recommended to select Background "none", which leads to a grey windows type background color. I prefer grid and smith diagram grid colors to be white on a grey background. To change colors, left-click the color boxes and select a color of your choice.

You can display **two horizontal freely placable reference lines** if you like. If you don't need them, check "RefLine Color " "none".

You can select the traces you want to display with the "**Trace Enable**" checkboxes. When checking one of the traces, a trace information field pops up, where the memory space (S21, S11, S12, S22, Memory1...4) and the display type (dB, Smith...) can be selected.

When you are done with these settings, close the Display Settings window and open the menu item "**Settings**" - "**Sweep Settings**:



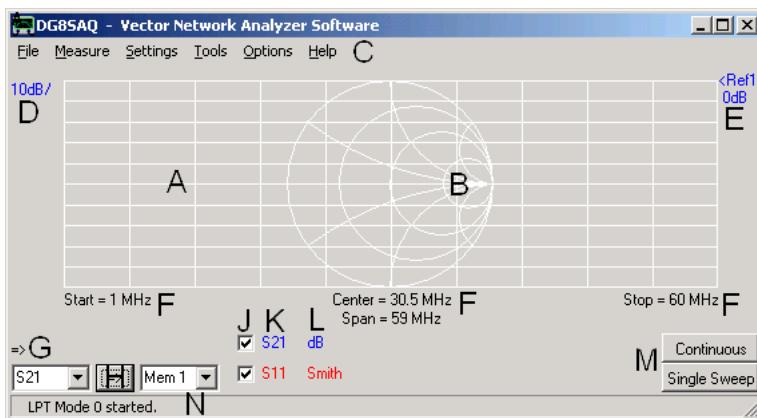
Select the **number of data points** (min 1, max 8192) and the **sampling time per data point** (min 0.2 ms, max 100 ms) for your measurement.

►►► **Note:** Sweep rates below 1ms / frequency point are only available in USB mode.

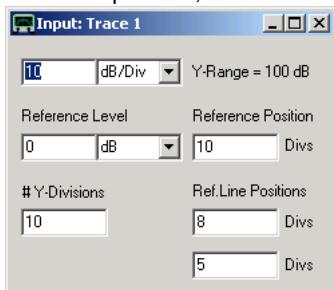
►►► **Note:** The sample rate slider will block on the left at the fastest possible sweep rate. If you want to sweep faster you need to increase the IF by decreasing the #samples per IF period and/or increasing the sample rate.

Select a **sweep progress display** option if you like. This is particularly usefull, when doing slow sweeps.

Close the Display Settings window when done. You should see the main graphics window again with your new colors.



A **dB vs. frequency grid (A)** and a **Smith chart (B)** are visible. Above, you find the **main menu (C)**. The top left blue **scale label (D)** indicates the scales per division for the to be blue trace. The right blue "reference label" (E) indicates the reference level and reference position of the to be blue trace. To change the scale, the reference level or the reference position, double-click either of the labels D or E, and modify the data in the window which is popping up:



Also, the number of horizontal grid lines and the position of the optional reference line can be changed here. Close the window when done.

An alternative way to change the scale/division is to hold the mouse pointer over the corresponding reference label (E, the shape of the pointer will change) and turn the mouse wheel.

Yet another possillity is to right-click the label.

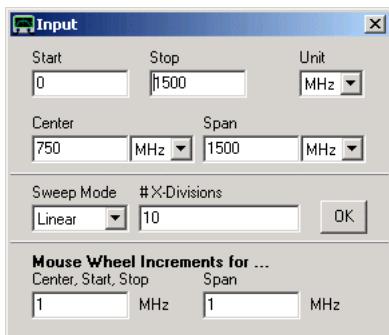
An alternative way to change the reference level of a trace is to move the mouse pointer to the according reference label of the trace on the right hand side of the grid (E, mouse pointer shape will change), then press the left mouse key and

move the mouse up or down while keeping the left mouse key pressed. This way, the traces can be moved up or down the screen with the mouse in realtime.

Yet an alternative is to right-click the reference label.

A very useful feature is the "**get scales from**" feature accessible on right-click on any trace label (D,E,L). It allows to copy all scale information from one trace to another, e.g. when you want to save S21 to Memory 1 and then display Memory 1 with the same scales as S21.

In order to set the measurement start and stop frequencies, double-click on one of the **frequency labels** below the grid (F). The following **frequency "Input"** window will pop up:

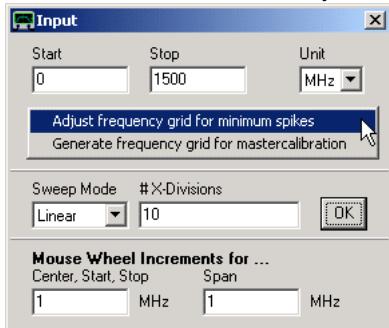


Enter start and stop frequency or alternatively center frequency and frequency span. The units selected here will also be displayed on the main screen. The center unit will also be used for markers.

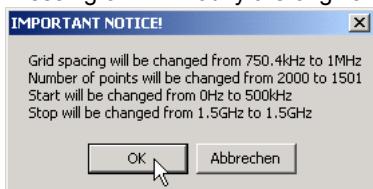
Also, the number of vertical grid lines and the sweep mode (linear, logarithmic, listed sweep) can be changed here.

Note, that the start, stop, center frequency and the frequency span can also be changed with the mouse wheel by holding the mouse pointer over the corresponding main window frequency label below the grid (F).and turning the mouse wheel. The increment values can also be set in the frequency "Input" window.

Note, that while VNWA2 has a quite odd clock frequency of 35.9 MHz, the VNWA3 clock frequency is exactly 12 MHz. Since, very sensitive measurements will reveal some degree of undesired interference at multiples of the clock frequency, it might be desirable to modify the frequency grid such that these frequencies are being maximally avoided. This can be done automatically for the linear sweep mode by right-clicking the "linear" combo box:

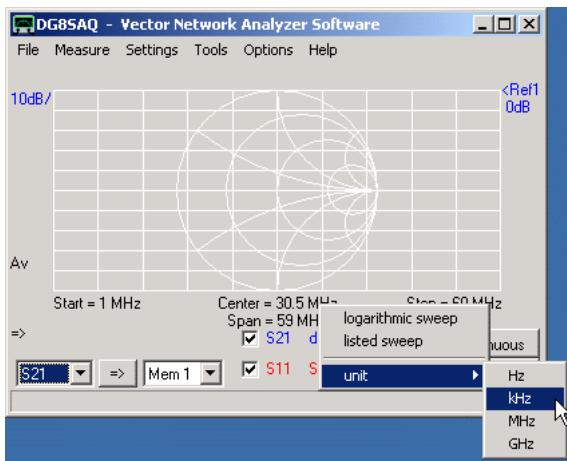


Pressing ok will modify the original frequency grid:



Close the frequency input window when done.

Note, that you can modify the displayed frequency units at labels K (the center unit will also affect the marker unit!) by right-clicking one of the frequency labels:



Start and Stop labels will always show the same unit.

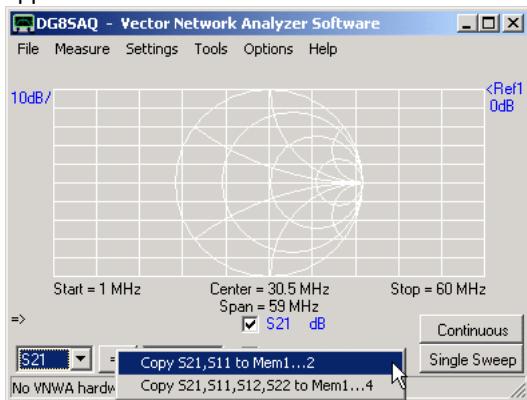
If you have calibrated your instrument, it is now ready for sweeping (which means measuring). To do so, press the button **"Single Sweep"** (M) if you only want to acquire a single frequency sweep or the button **"Continuous"** (F) for continuous sweeping. Both sweeps can be interrupted by pressing the same button again.

Warning: Your measurement results will be quite meaningless unless you have calibrated the instrument!

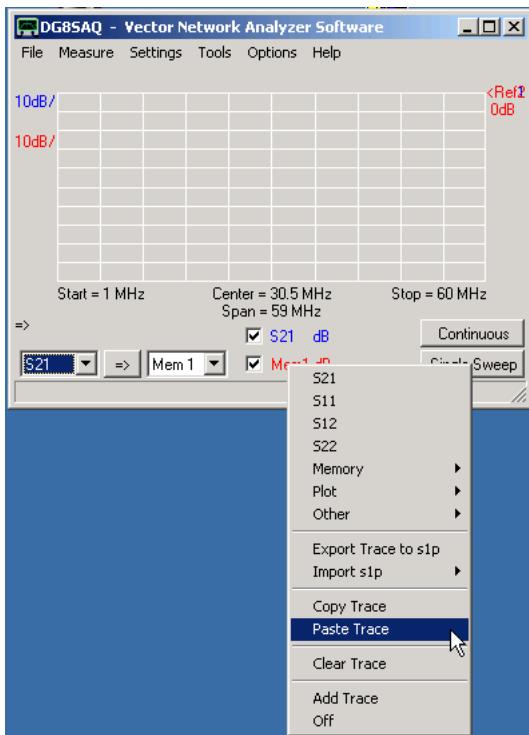
Each trace can be switched on or off by checking/unchecking the checkboxes below the grid (J).

You can also change the displayed **memory space** (**S21, S11, S12, S22, Mem1...Mem4**) and the **display type** (**dB, Smith, mag, phase...**) by right-clicking the labels next to the checkboxes (K, L). Alternatively, double-click the label K or L.

Trace data can be copied from one memory space to another (e.g. S21 to Memory 1) with the button H. The combo box to the left of the button selects the source, the one on the right selects the target memory space for the copy process. Several memory spaces can be copied simultaneously by right-clicking the copy button H. A popup menu will appear in this case:



It is also possible to **copy data** from one trace to another using the **trace right-click menus copy and paste**:



Note, that this facility does not use the Windows clipboard, thus it cannot be used to copy data from one VNWA instance to another but only from one trace to another inside the VNWA application. Also note, that it is obviously not possible to paste data to custom, optimizer and spline traces as these are not defined by data fields.

Markers can be added by right-clicking into the grid field or into the smith chart. Marker positions can be changed by dragging the markers with the mouse (pointing the mouse onto the marker triangle, pressing the left mouse key and keeping it pressed while moving the mouse).

The "File" menu allows to **save or print the screen**, to **save or load S-parameter files** in Touchstone format (s1p, s2p) or **save/load calibrations**.

S-parameter files can also be loaded most conveniently by dragging them from any file browser and dropping them onto the VNWA main window.

MEMORY SPACES

General

Data inside the VNWA software is organized in distinct **memory spaces**, which generally contain only S-parameters, but **no frequency information**. All memory spaces (except the "Plot" spaces) share the same frequency frame data given by the x-axis of the main diagram in unzoomed mode.

Available memory spaces:

S21 = measured data S21
S11 = measured data S11
S12 = measured data S12
S22 = measured data S22
Mem1 = data space Mem1 data
Mem2 = data space Mem2 data
...
Mem10 = data space Mem10 data
Plot1 = data space Plot1 data
Plot2 = data space Plot2 data
...
Plot10 = data space Plot10 data
s_11...s_33 = measured 3-port data
Cust1...Cust6 = custom calculated data

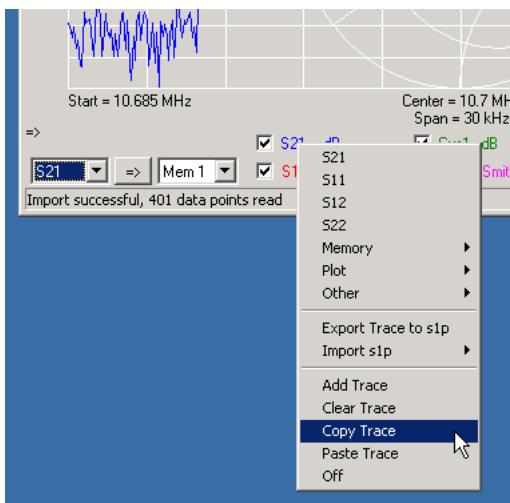
►►► **Note:** If the frequency frame data is changed, the memory space data remains unchanged (except for the "Plot" data spaces, which contain original frequency information), thus, you will see the same traces with changed frequency axes, which generally invalidates data.

►►► **Note:** If the number of frequency points is changed, the **memory space data will be erased** (except for the "Plot" data spaces, which contain original frequency information).

►►► **Note:** Memory space data can be exported to a file or imported from a file by means of the "File"->"Export Data" or "File"->"Import Data" menu.

►►► **Note:** Data can be copied from almost any memory space to almost any other memory space by means of the "=> button".

►►► **Note:** Data can also be copied and pasted from any memory space to almost any other memory space by invoking the right-click menus "**Copy Trace**" and "**Paste Trace**".



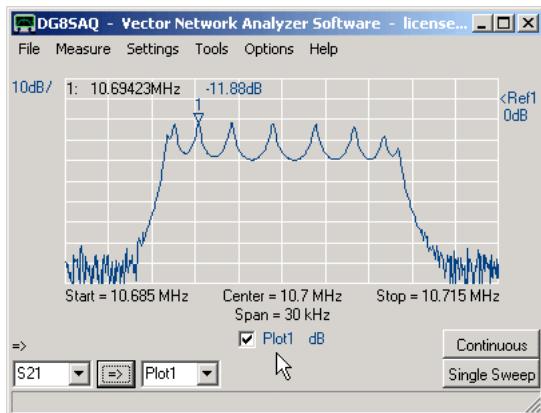
Special features of the Plot memory spaces

The memory spaces Plot1...Plot4 each contain an individual frequency grid memory. While importing data into e.g. Mem1 causes the frequency grid for all traces (except Plot* traces) to be modified, importing data into the Plot* memory

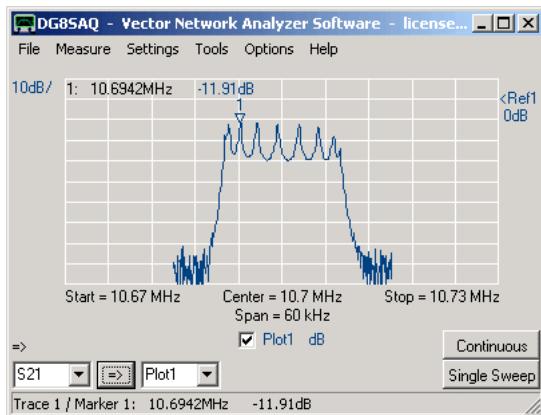
spaces leaves the main frequency grid untouched. The Plot* data remains valid upon changing the main frequency grid (frequencies, lin/log scale) or the number of frequency data points. Note, that marker data for the Plot* memory spaces is interpolated to the main frequency grid.

Example:

1. Copy any data to Plot1 and display it:



2. Increase the frequency span and the number of data points:



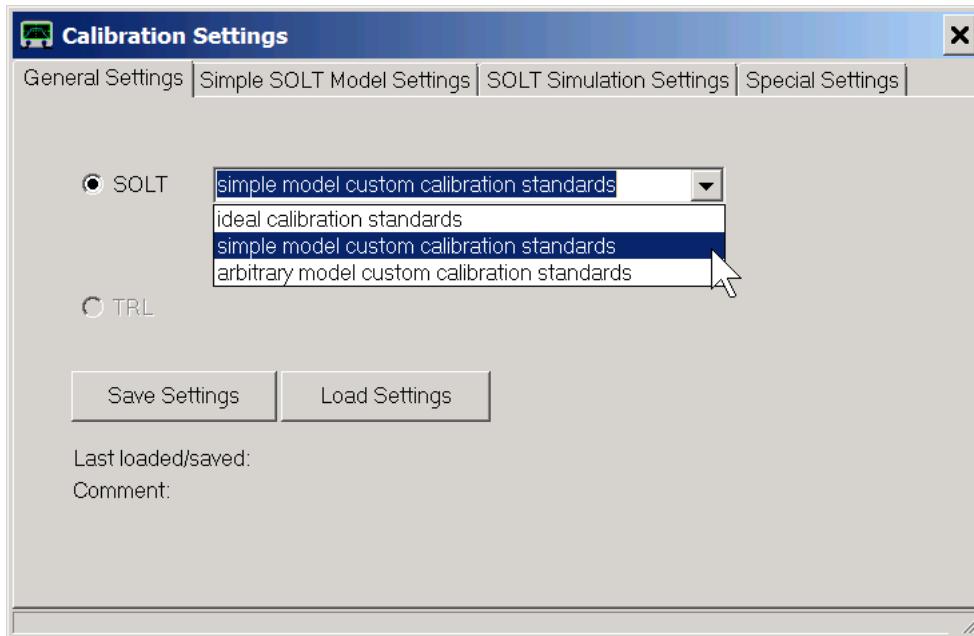
You will still see correct data plotted into the valid frequency span. Note, that marker levels have slightly changed due to the frequency grid interpolation.

INSTRUMENT CALIBRATION

- | **Calibration Standard Setup**
- | **Performing a Calibration for a Two Port S-Parameter Measurement**
- | **Master Calibration**

CALIBRATION STANDARD SETUP

Todate, the VNWA software allows for SOLT (Short, Open, Load, Thru) calibration. Before actually performing an instrument calibration, you need to specify the RF properties of the calibration standards you are going to use. To do so, select from the VNWA main menu "Settings" the submenu "Calibration Kit Settings". The following window will open:

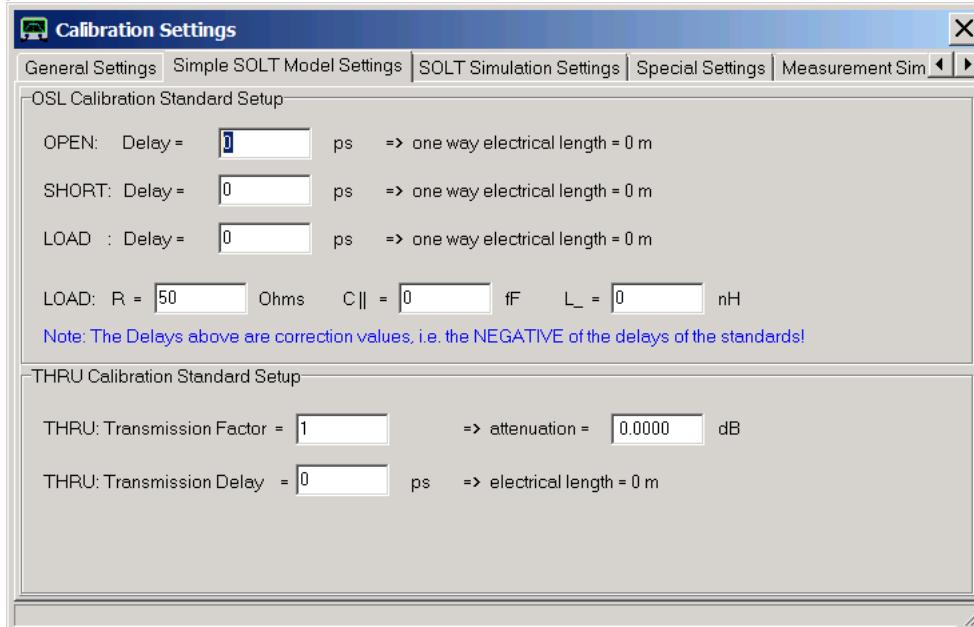


If you do not know anything about your cal standards, select "ideal calibration standards". In this case, the load standard is assumed to be 50 Ohms, the open and short standards are assumed to be ideal open and short circuits of zero length.

Simple Calibration Standard Model

For high precision measurements in the VHF and UHF range, this is generally not good enough. In this case you need to obtain a more precise model of your cal standards.

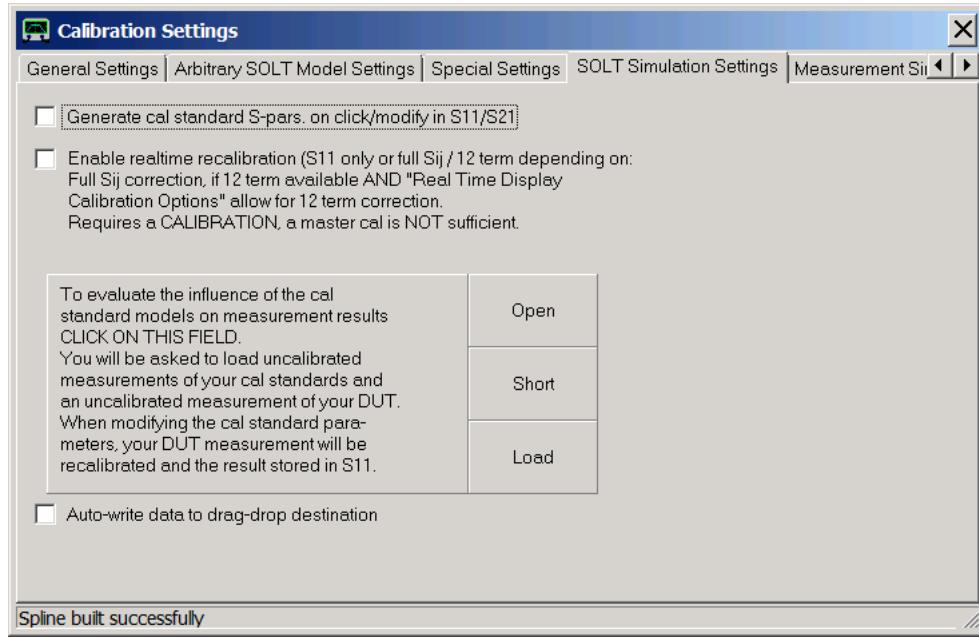
If you select "simple model custom calibration standards" and open the "Simple SOLT Model Settings" tab, this is what you'll see:



Enter the electrical delays of your open and short standards as well as the DC resistance and shunt capacitance of your load standard.

Kurt Poulsen has compiled calibration kit data for the calibration standards available through SDRKits which can be downloaded from his web page <http://www.hamcom.dk/VNWA/>.

You can accurately measure the DC resistance of your load standard with a 4 wire Kelvin probe. In order to obtain the other parameters for standards not covered by Kurt Poulsen's documents, either refer to the manufacturer's data sheets or measure them with a reference VNWA. For comparison between the measured reflection coefficients from a reference VNWA and the model data, you can simulate the model calibration standard reflection coefficients by activating the checkbox "Generate cal standard S-parameters on click/modify in S11/S21" in the "SOLT Simulation Settings" tab:



S11 data will automatically be updated whenever you modify the calibration standard model parameters. E.g. if you display the measured reflection coefficient of your Open in memory space Mem1 and compare it to the simulated reflection coefficient in memory space S11, you can modify the parameter "Open delay" (it can also be modified with the mouse wheel), until both curves match.

►►► **Hint:** You will only see a difference in delays, if you look at the phase.

►►► **Hint:**

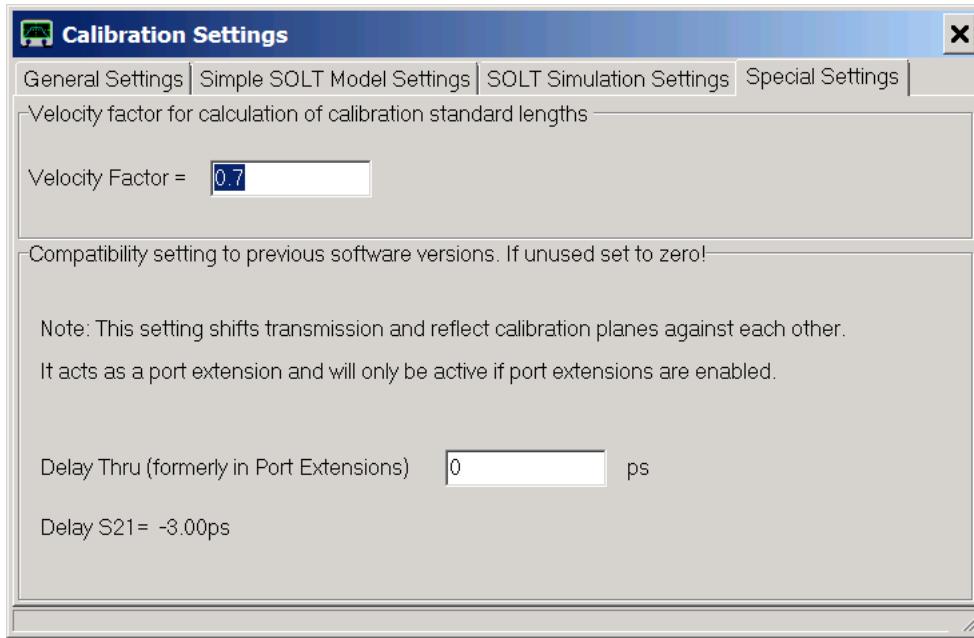
If you do not have access to a reference VNWA, then you might use the realtime recalibration feature (greyed in above figure). To do so, you must previously have performed a SOL reflect calibration with your unknown calibration standards and you need to have measured the reflection coefficient of an about 30cm long open ended or shorted high quality 50 Ohms coaxial transmission line, preferably an air line or of semi-rigid type. It must be measured on your VNWA which is calibrated with your unknown cal standards. Make sure, the measured line reflection coefficient is displayed in S11. If a valid calibration is available, then you are allowed to activate the "**enable realtime recalibration**" checkbox. When activated and if you click on one of the cal standard parameter fields and turn the mouse wheel, then the parameter will change and the calibration will be recalculated in realtime. You will see this by a changing S11 trace. This way you can actually tune the cal standard parameters for lowest port mismatch ripple of your transmission line measurement in amplitude (dB) and phase progression (-continuous phase/f). Make sure you keep one delay (e.g. that of the Open standard) at a fixed value (e.g. =0). This will define the position of the calibration plane. See an example on page Extracting calibration kit parameters.

►►► **Hint:**

The feature in the big field in above figure may serve the same purpose, but it utilizes s1p-measurement data files instead of actual measurements. Instead of performing an actual calibration, you can calibrate the system by loading uncalibrated *.s1p files of your calibration standards. To do so, you can either click on the corresponding fields or drag and drop the *.s1p files to the corresponding fields (Open, Short, Load). After having imported a SOL set of cal standard measurements, an instrument calibration will be calculated automatically. Then, uncalibrated DUT measurement data can be corrected with the calculated calibration by dragging and dropping the corresponding *.s1p-file(s) to the big field. If the "Auto-write" checkbox is checked, then the corrected DUT measurement data will automatically be written as s1p-file(s) back to the data destination folder with modified file name(s).

►►► **Hint:** Calibration standard delays are entered and stored as time delays. But the software converts and displays

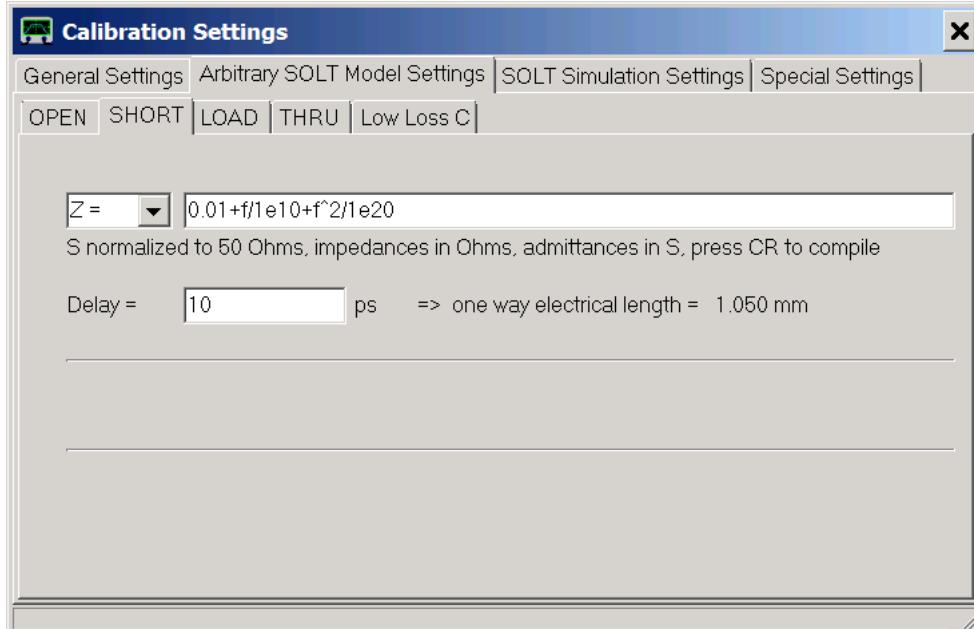
them for convenience as propagation lengths as well. To do so, the software needs to know which velocity factor to use. This velocity factor can be defined in the "Special Settings" tab. The predefined value of 0.7 is quite typical for PTFE dielectric SMA standards:



The **Delay Thru setting** in above screenshot should **always be set to zero** unless you exactly know what you are doing. It is only retained for compatibility with previous software versions. This is a port extension that acts differently on transmission and reflection.

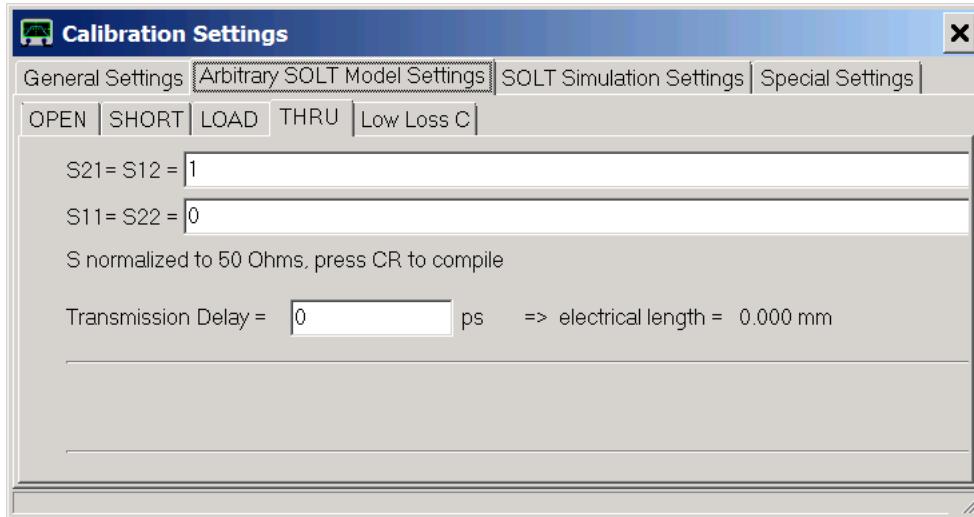
Arbitrary Calibration Standard Model

From VNWA software version 35 on the user may create his own calibration standard model by selecting "arbitrary model custom calibration standards" in the "General Settings" tab. In this case the cal standard model can be defined by a custom formula:



The user can specify a formula for the impedance Z, the admittance Y or the reflection coefficient S of the reflection standards. For convenience, a delay needn't be programmed into the custom formula but can be specified in the field below. The formula will be parsed upon pressing the return button. Parsing errors will be reported between the bright lines.

Similarly, the transmission and reflection coefficients of the thru standard can be specified by formulas:

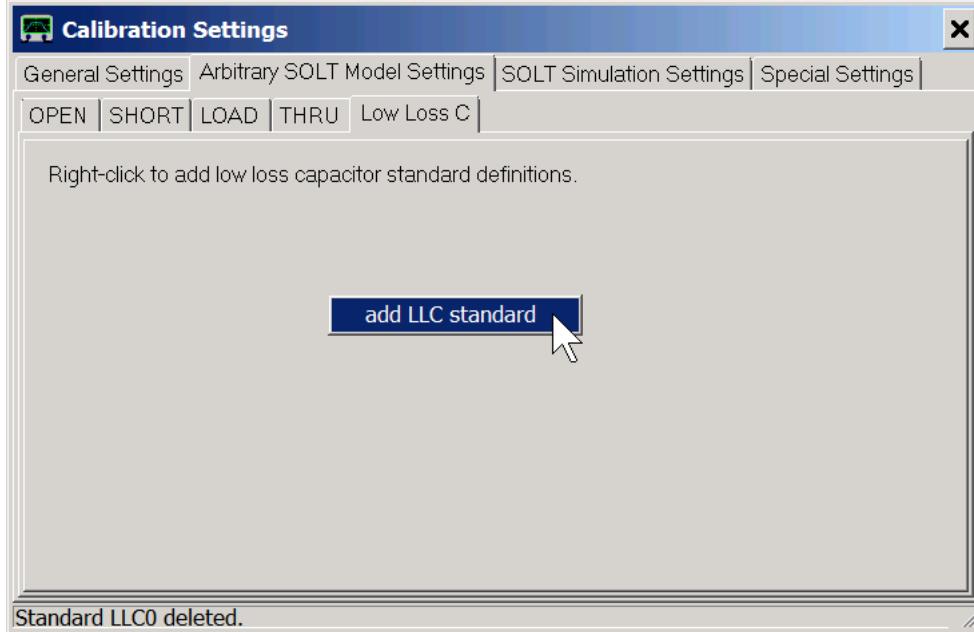


The only restriction is that the thru standard be reciprocal, i.e. $S_{21}=S_{12}$, and symmetric, i.e. $S_{11}=S_{22}$. Otherwise, just about anything with two connectors can be used as a thru standard provided its properties are known.

Low Loss Capacitor (LLC) Calibration Standards

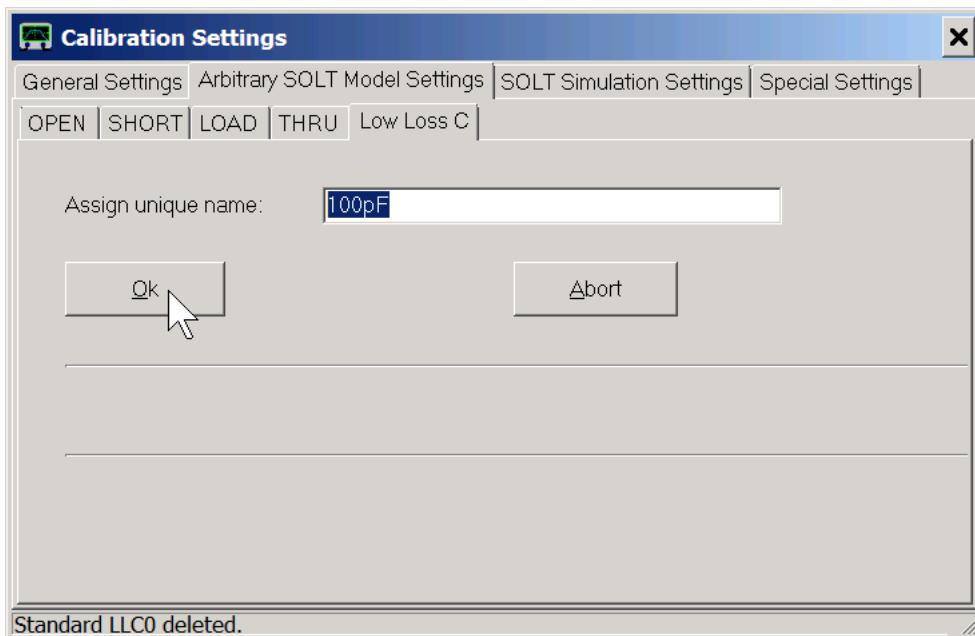
From VNWA software version 36.0 on, low loss capacitors can be used as optional additional calibration standards. These are user definable and thus only available within the arbitrary calibration standard model. As capacitors are strongly frequency dependent impedances, several capacitor calibration standards with different capacitance values are needed to cover a wider frequency range.

A new LLC standard can be defined by right-clicking onto the **Low Loss C**-tab:

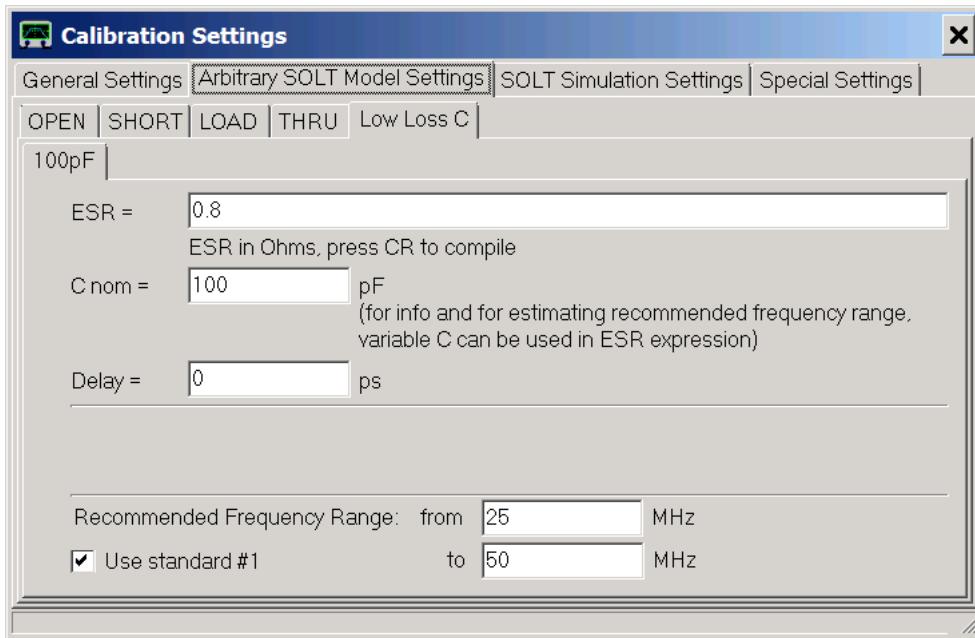


The popup menu also allows to delete or rename an existing LLC standard.

Upon selecting "add LLC standard", you are asked to enter a unique name for the new standard, e.g. its capacitance value:



After confirming with **Ok**, a new tab with the previously assigned name appears, where the calibration standard definitions can be entered:

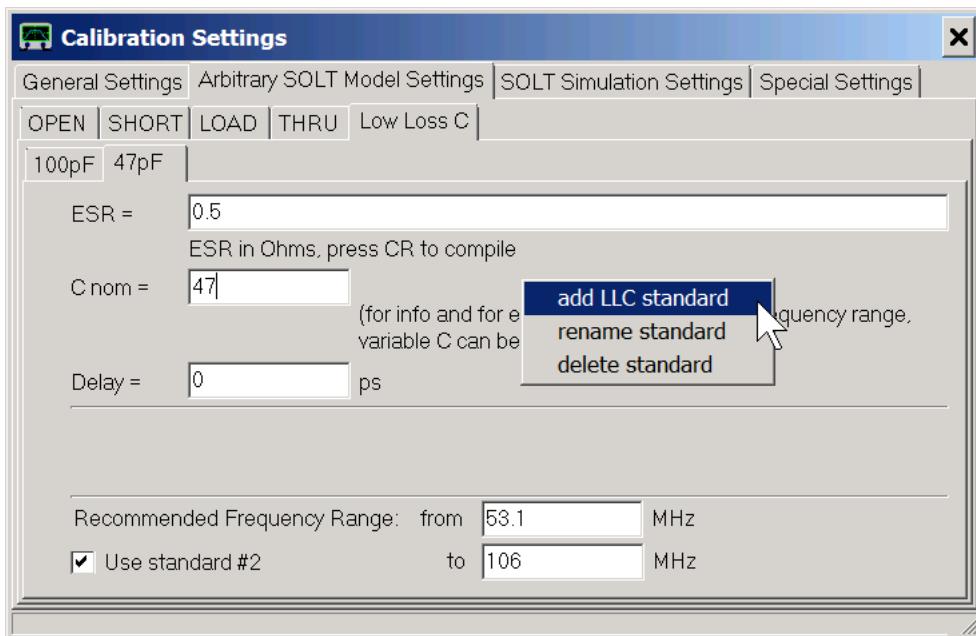


User definable settings:

- **ESR:** Effective Series Resistance of the capacitor. This may be frequency dependent and must be modeled. Variables f (frequency), w (angular frequency) and C (capacitance) may be used in the ESR expression, likewise all functions supported by the custom trace compiler.
- **C nom:** Nominal capacitance of the standard. This does not need to be accurate. It is only used to estimate the standard's frequency range of applicability, the **recommended frequency range**, which is displayed on the bottom of the tab.
- **Delay:** The time delay of the capacitor position relative to the calibration plane.
- **Use standard #...:** If checked, the standard will show in the calibration menu. If unchecked, the standard remains invisible during calibration. See page Low Loss Capacitor Calibration for details.

►►► **Note:** The recommended frequency range boundaries cannot be edited. These are automatically calculated from the entered Cnom value.

This way, up to 8 LLC standards can be defined:



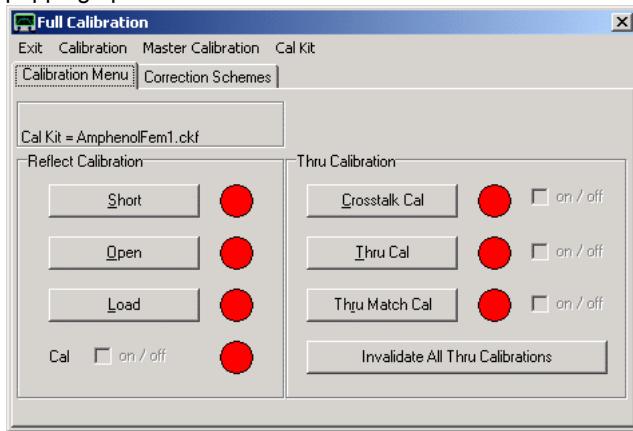
PERFORMING A CALIBRATION FOR A TWO PORT S-PARAMETER MEASUREMENT

When measuring S-parameters of a two-port device (S_{11} , S_{21} , S_{12} , S_{22}), care has to be taken, that calibration planes for the reflect (S_{11} , S_{22}) measurement and for the through (S_{21} , S_{12}) measurement are identical, i.e. the phase values for all 4 S-parameters are consistent.

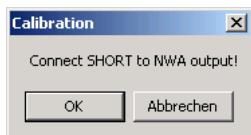
In the following, I describe the procedure, I normally use to calibrate my VNWA. This is not the only possible way to perform a proper calibration, see Kurt Poulsens documents at <http://www.hamcom.dk/VNWA/>. A different way to do a 2 port calibration using female Amphenol calibration standards provided by SDRKits is described on page "Practical Example on Performing a 2-Port Measurement".

►► **Important Note:** Before performing a calibration measurement, make sure, that the **instrument sweep rate is set to the lowest value** you intent to use the calibration with, otherwise your later measurement results will be dominated by the noise of your calibration measurements.

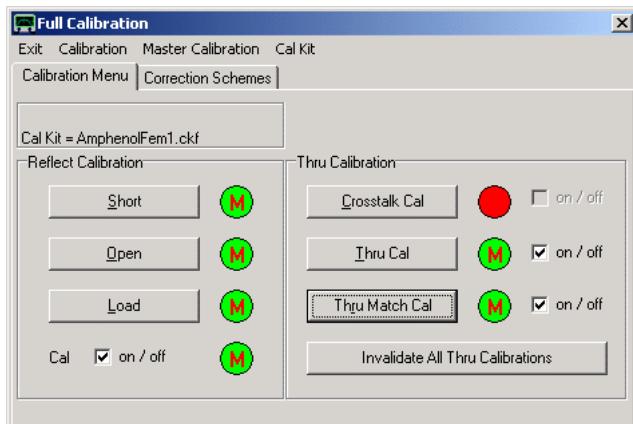
In order to initiate a two-port calibration, click the menu "**Measure**" and "**Calibrate**". You'll see the following window popping up:



When you press the buttons next to the red lights, you'll see a window popping up which tells you, which cal standard to connect, e.g. when pressing on "Short", you'll see this window:

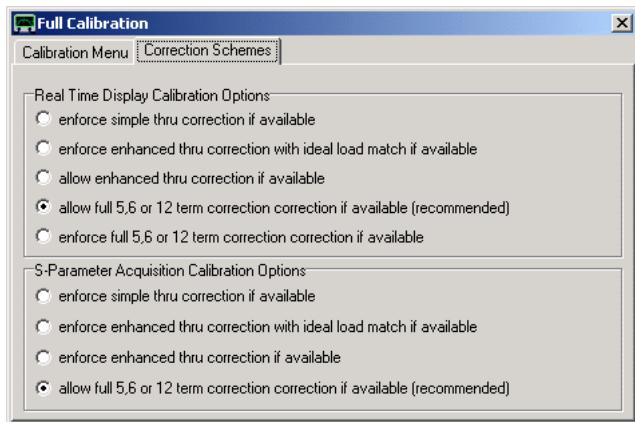


Now, connect the calibration standards, which you are asked for, and press ok when done. Output is synonymous for TX port and input for RX port. If you press "abort", no measurement will be performed. You can repeat any calibration measurement at any time. You can calibrate in any order you like. After a successful calibration sweep, the corresponding red lamp turns green indicating, that valid calibration data is available:



Note the red "M"s inside the green lamps. These indicate that no master calibration is available yet. By clicking on the green lamp, you can invalidate a calibration measurement and the green lamp turns red again. You can also switch corrections off and on again without invalidating the calibration measurements using the checkboxes next to the lamps.

The radio menus in the "Correction Schemes" tab allow to experiment with various degrees of mathematical corrections:



The recommended settings are the ones shown above.

In the following I will exactly describe my cal standards and how the calibration is to be performed.

Thru Calibration:

Starting point are the two coaxial cables comming from the VNWA TX port (output, left) and going to the VNWA RX port (input, right) with two SMA male connectors.:



In case you run the "**Crosstalk Cal**", just leave the two cables separated from each other as shown above.

►►► **Note:** Normally a crosstalk calibration can (and should) be omitted, as the VNWA's crosstalk level is below the noise floor. In this case, a crosstalk calibration only increases the noise level by 3dB. Use the crosstalk calibration to eliminate crosstalk of test adapters if necessary.

For the "**Thru Cal**" you need to connect the two cable ends. Obviously, you need an SMA female to female adapter as can be seen in the following picture:



Note that the adapter introduces some phase shift because of its finite electrical length that has to be accounted for in the further reflection calibration.

The "**Thru-Match**" calibration step requires the same thru standard as the thru calibration. While the thru calibration step measures transmission, the thru match calibration step measures reflection of the RX port in order to numerically compensate deviations from a perfect 50 Ohms RX port match.

Reflection calibration:

One possible way to deal with the additional phase shift of the thru adapter is to leave it in the system when performing the reflection calibration. The following picture shows my open "standard", which is the bare female SMA side of the thru adapter:



"Open" calibration standard

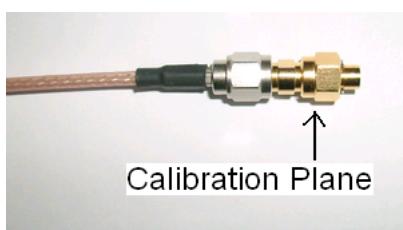
This nails the reference plane to the place where the RX cable connector was sitting during the thru calibration and ensures zero phase difference between S11 and S21.

An alternative calibration scheme using the calibration standards available thru SDR-Kits can be found on page Practical Example on Performing a 2-Port Measurement.

The thru adapter is also in place when measuring the "Short" and "Load" standards, which are both commercial products with male SMA connectors.



"Short" calibration standard



"Load" calibration standard

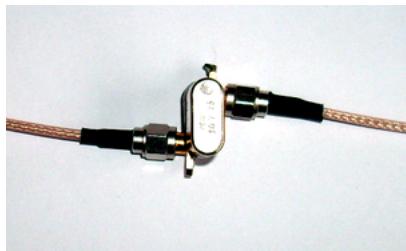
The "Short" standard is an SMA male connector with a built-in short circuit. The "Load" standard is an SMA male connector with a built-in 50 Ohms resistor.

Actually, for open, short and load standards, the calibration planes will still not be exactly at the same position as internal builds differ slightly. These slight errors can be corrected for by entering proper calibration standard parameters if known.

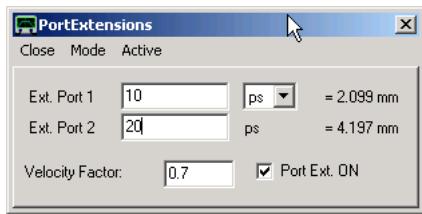
Finally, you want to remove the Thru adapter so you can connect the VNWA cables to your test object (a monolithic crystal filter in below picture)
in forward direction to measure S11 and S21



and reverse direction to measure S12 and S22.



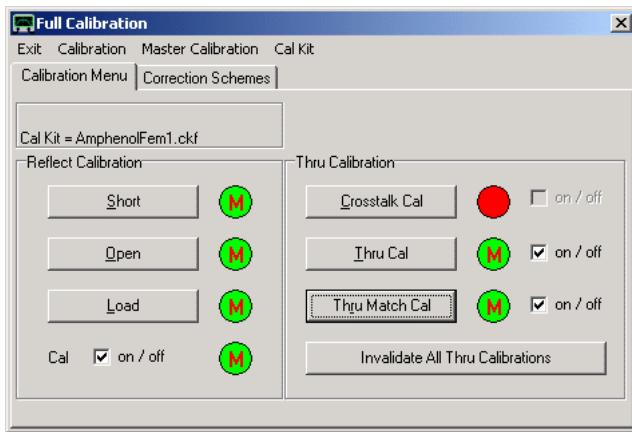
The only remaining effect to be corrected is, that your reference plane still sits at the position where the end of the Thru adapter was, which would be well inside your test object now. This can be compensated in the port extension menu invoked by "Measure"->"Port Extensions":



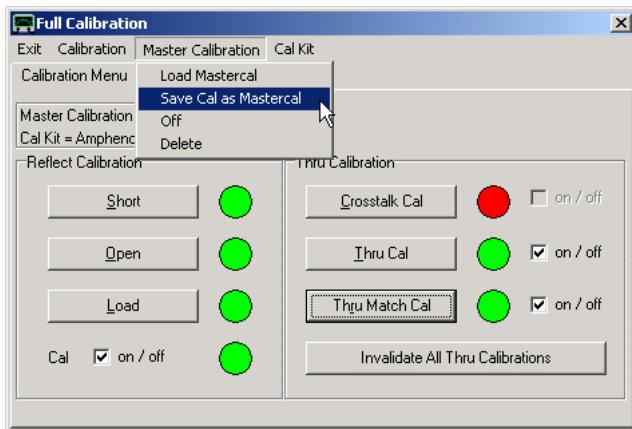
You can move calibration planes for forward and reverse measurements independently, by entering delays into the "Ext. Port 1" and "Ext. Port 2" fields. This might be useful, when you want to correct for different input and output connector lengths of your test object or if you want to move the calibration planes beyond the connectors to the solder pads of your component under test. A positive port extension will move the calibration plane away from the VNWA. Port extensions will be applied only if the "Ext. on" checkbox is checked. You can monitor the application by a slight rotation of your previously measured data in the Smith Chart.

MASTER CALIBRATION

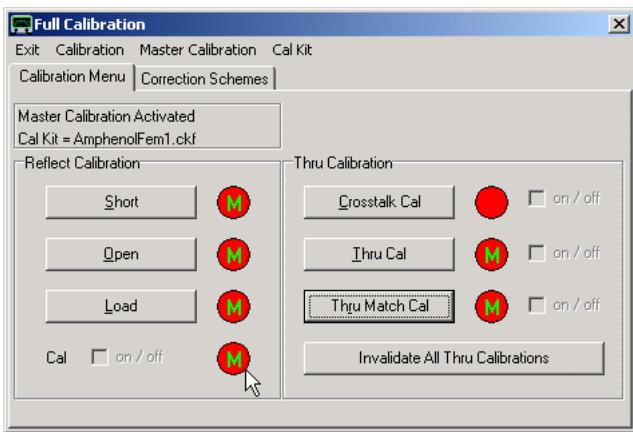
As long as you haven't performed a calibration, the VNWA will yield quite arbitrary and useless results. Once you have decided, what frequency range to sweep and how many data points to collect, it's time to do the calibration. **When you modify the number of data points or the frequency span, the calibration will be invalidated and is thus lost.** To avoid the necessity for repeated calibrations for standard moderate precision measurements, the master calibration feature has been implemented. The idea is as follows: Calibrate the instrument once with a large number of points (e.g. 8192) and the maximum span (1 kHz...1.3 GHz):



Then make this calibration the master calibration by saving it as a master calibration. You are asked for a file name to store the calibration for later reload after program restart. Successful activation of a master calibration is indicated by the red "M"s inside the green lamps turning green and thus vanish:



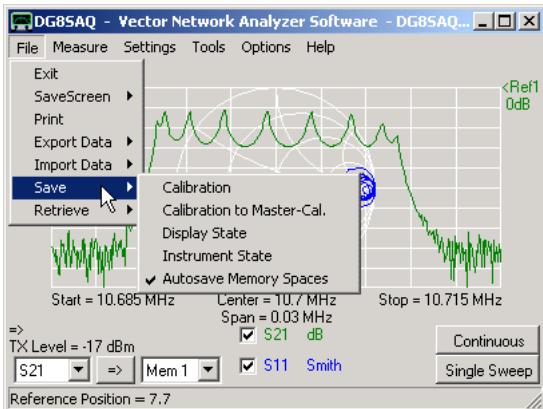
When you change the number of points or the frequency span, and the calibration is thus invalidated, the master calibration data is used to interpolate new calibration data. In this case, the instrument will still deliver reasonable results, even though it is not calibrated. This can also be seen from the calibration window which shows green "M"s inside red lamps indicating no calibration but an activated master calibration:



An active master calibration is also indicated by the **Master Calibration Activated** info in the top left rectangular info field above.

►►► **Hint:** You can have more than one master calibration by selecting different file names. You can reload an existing master calibration file using the Master calibration Menu above or via the main menu "Setup" and "Misc-Settings" tab.

Note, that you can also transform a loaded calibration into a master calibration via main menu "**File-Save-Calibration to Master-Cal.**".



You can load a calibration directly as master calibration via main menu "**File-Retrieve-Mastercalibration**" as well as via "Options-Setup" Misc. Settings tab, button "**Browse and Load Master Cal.**"

LOW LOSS CAPACITOR CALIBRATION

| Why low loss capacitor (LLC) calibration?

Theoretically, a SOL-calibration does remove all linear errors that can occur in an impedance measurement with the VNWA.

In real life, error correction is only as accurate as the calibration kit models.

Small changes in the models may yield big changes in the measurement results, particularly, when measuring extreme impedances, i.e. near the edge of the Smith chart.

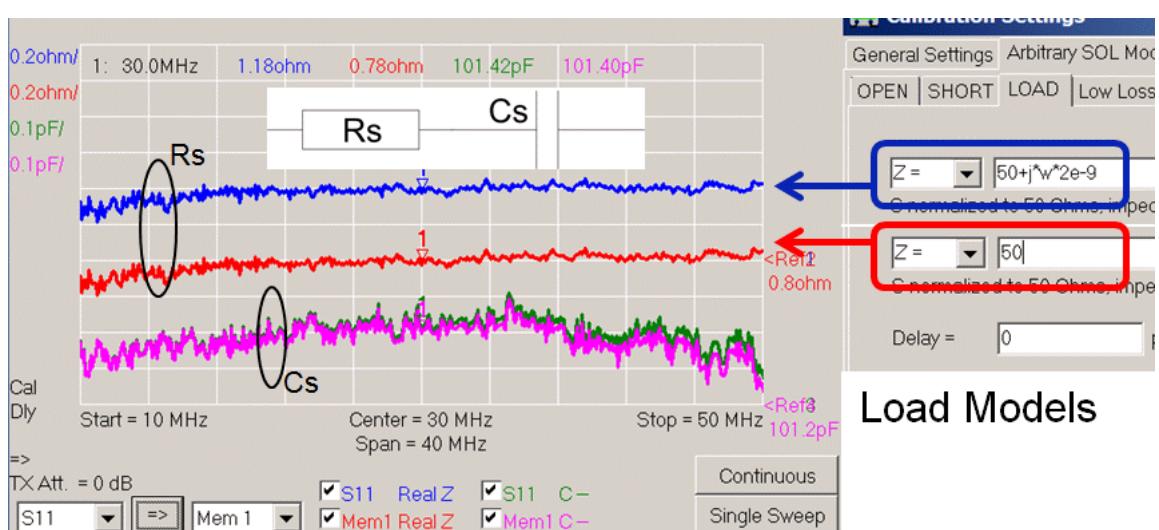
Accurately measuring Q-values of capacitors, inductors and resonators is a task particularly sensitive to the calibration standard models.

Problem:

The following picture shows an impedance measurement of a standard SMD 100pF capacitor. The measurement data has been corrected using **two different Load models**:

red: ideal load model (i.e. $Z=50$ Ohms)

blue: load model consists of 50 Ohms and 2nH connected in series



The DUT capacitor can be modeled as a series connection of capacitor C_s and resistor R_s .

Note, that while the measured capacitance C_s is quite independent of the load model, **the measured R_s changes by 50%**, even though a 2nH inductor is only a very minor correction to our load in our frequency range ($j*0.1$ Ohms... $j*0.5$ Ohms). And the effect persists even to the lowest frequencies.

=> Measured **Q-values depend strongly on the imaginary part of the load standard model impedance** for SOL calibration! So, for accurate Q-measurements you have to have this absolutely right.

BUT... The load standard's imaginary part of impedance is usually the one parameter of all calibration standards that is least accurately known, while the load's absolute value of impedance can be measured quite accurately at DC with a Kelvin probe.

Solution LLC-calibration:

Accepting that there will always be poor knowledge of the imaginary part of Z_{Load} , an additional calibration standard is needed to overcome this problem.

As the missing knowledge strongly influences loss measurements on e.g. capacitors, a capacitor with well known loss is best suited as additional calibration standard.

Best suited would be a capacitor with no loss at all (e.g. Effective Series Resistance ESR=0), second best is a low loss capacitor (LLC). When ESR is small, then the ESR uncertainty is automatically small as well.

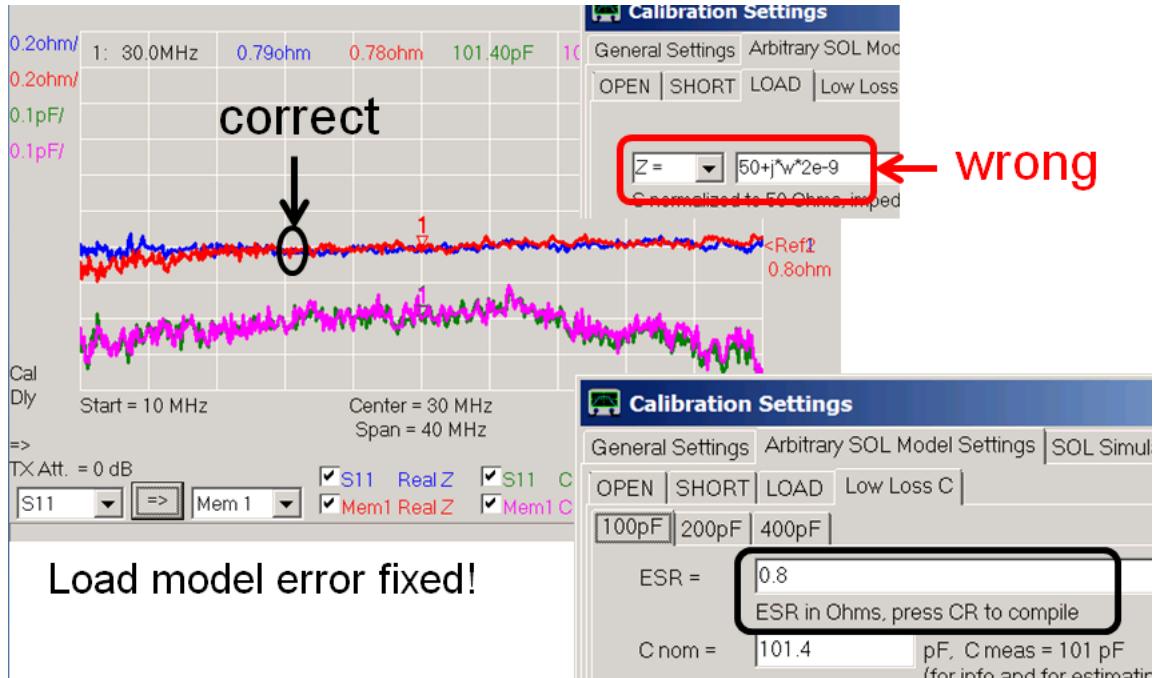
This implies if the Q of the capacitor is much higher than the Q of the DUT, then ESR might even be set to 0, as in this case it hardly influences the measurement result.

Since the magnitude of Z_{Load} is assumed to be known, the exact capacitance value of the calibration capacitor isn't required to be known. It is sufficient to know the loss, i.e. the effective series resistance ESR.

Note, that even though the exact capacitance value need not be known, the LLC capacitance still influences the usable frequency range. A capacitor is equivalent to an Open at very low frequencies and to a Short at very high frequencies. Thus, it does not deliver additional information at these extremes over the Open and Short standard. In order to cover a

wide frequency range, a number of LLC standards with different capacitances is required.

The following picture demonstrates how an additional LLC calibration fixes a wrong Load standard model. The same 100pF SMD capacitor as above was used as test object and as LLC standard:



The traces stored in Mem1 (pink, red) were determined with the assumed correct ideal load model ($Z=50$ Ohms). The S11 traces (blue, green) were obtained with the wrong load model, but the calibration was extended by a LLC calibration measurement. After LLC calibration the S11 measurement reproduces the supposedly exact ESR=0.8 Ohms as entered into the LLC model, even though the load model is wrong.

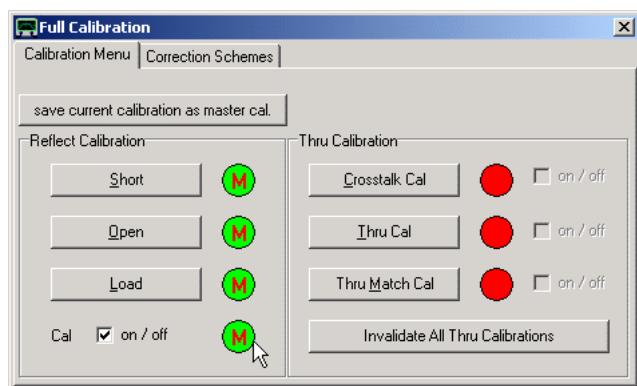
How to properly perform an LLC calibration:

In the following, a procedure to LLC-calibrate the VNA for a one-port measurement is described, assuming that the ESR of the LLC standard is known, but its delay is not.

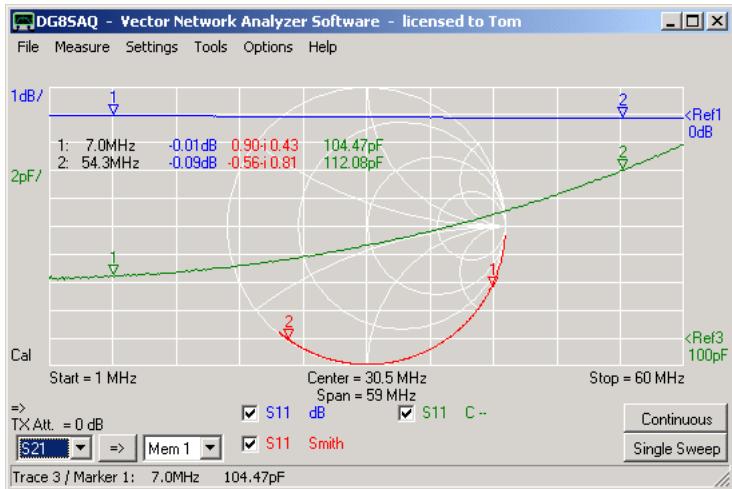
Note, that LLC calibration can only be used with the "arbitrary model" for calibration standards.

So, the first step is to determine the delay of the LLC standard.

To do so, first perform a normal SOL-calibration:

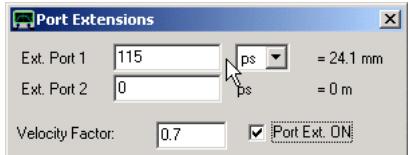


Next, measure your LLC standard:

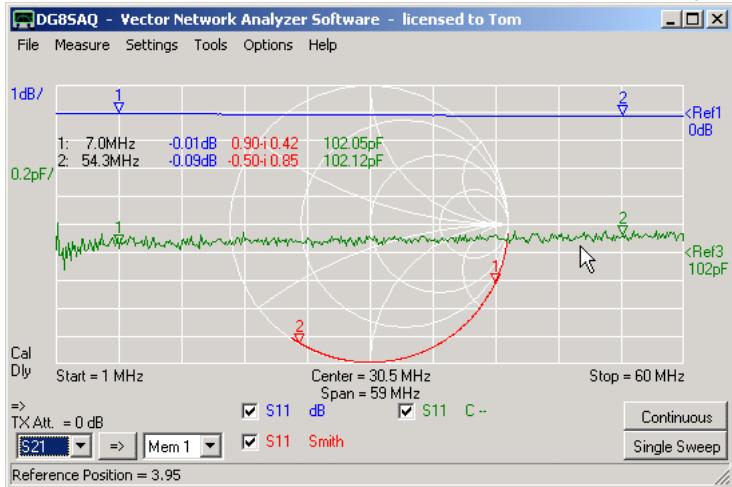


Generally, the capacitor will not be located in the SOL calibration plane. This leads to a frequency dependence of the measured capacitance value as seen above (green trace).

Next, we use a port extension to determine, how far away the capacitor is from the calibration plane.
Activate port extensions and tune the Port 1 extension ...



... until the capacitance becomes almost independent of frequency:

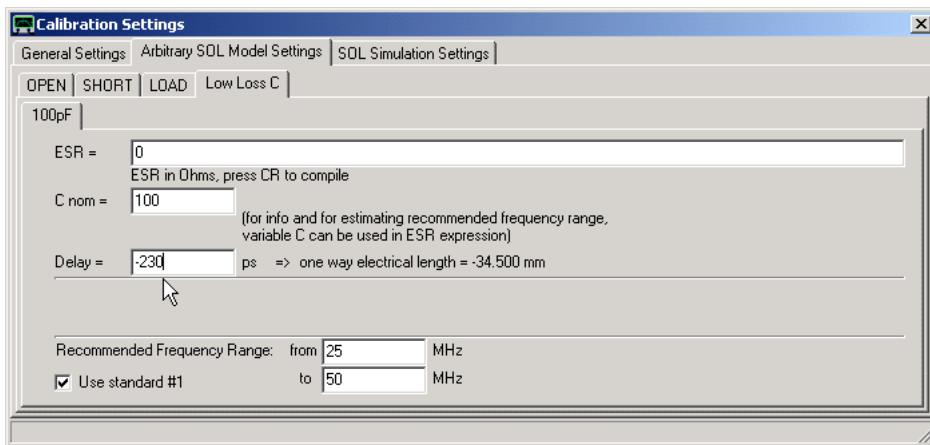


Note, that for a port 1 extension of 115ps the green capacitancy trace almost becomes flat.
Note, that that the scale for the green trace above is 0.2pF/ while it was 2pF/ in the previous plot.

Next, we create an LLC standard using the calibration settings menu.

The just determined port extension of our LLC standard is the one way delay from the calibration plane to the capacitor.
As the wave travels forth and back, the wave delay is twice the port extension, i.e. 230ps.

Now, we want to virtually move the capacitor back into the calibration plane. Therefore, we enter the negative of this delay, i.e. -230ps, into the LLC standard delay setting:



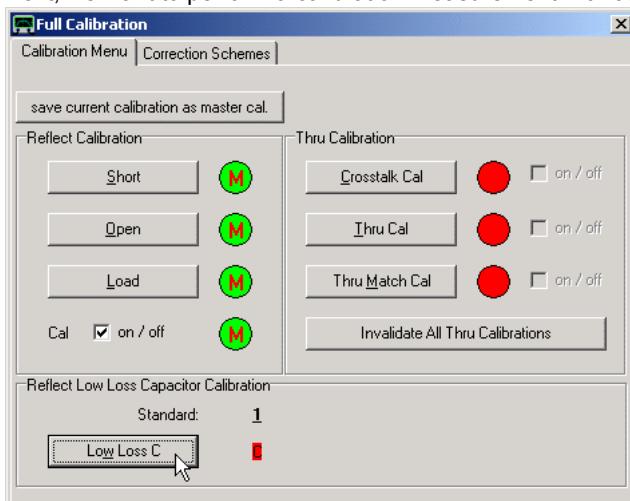
Note, that it is sufficient to enter an approximate nominal capacitance value. In the line ESR you should enter an expression modeling the frequency dependent ESR (effective series resistance) of your capacitor as closely as possible. Note, that you can use mathematical formulas for modeling. The syntax is identical with the custom trace compiler syntax, e.g. variable **f** denotes the frequency.

Example of an ESR that rises linearly in frequency from 0.1 Ohms to 10.1 Ohms at 1MHz:
ESR = 0.1+10*f/1e6

Either consult the capacitor data sheet or use a transmission measurement to determine the ESR without using the SOL calibration standards.

For a first test we use ESR=0, which is a good initial choice for very low loss capacitors.

Next, we want to perform a calibration measurement with our LLC standard:

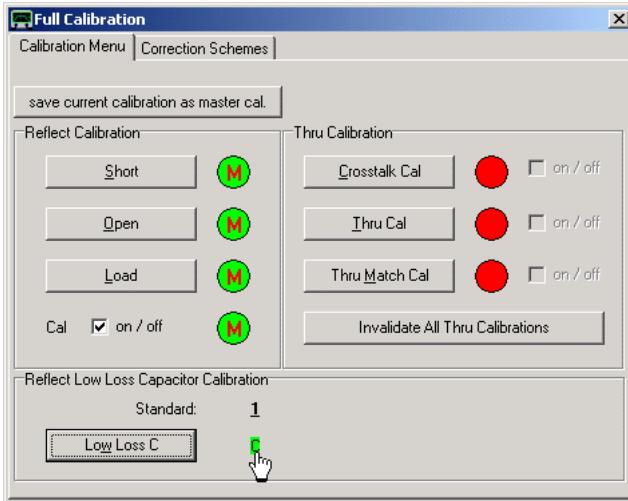


After clicking on the "Low Loss C"-button, you are asked to connect the standard. As you can define up to 8 different LLC standards, the standard's name is shown. Be sure to connect the right one.

The calibration sweep is performed for the underlined standard. If you want to use a different LLC standard, click the corresponding number and it will become underlined, i.e. active.

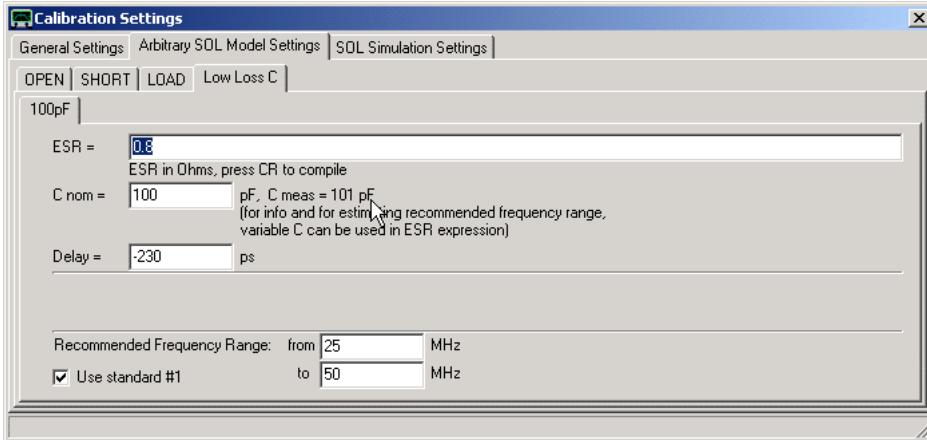


After the cal measurement, the LLC cal label turns from red to green indicating an active LLC calibration:



Note, that clicking onto the green LLC label will delete the calibration measurement and turn the label red again.

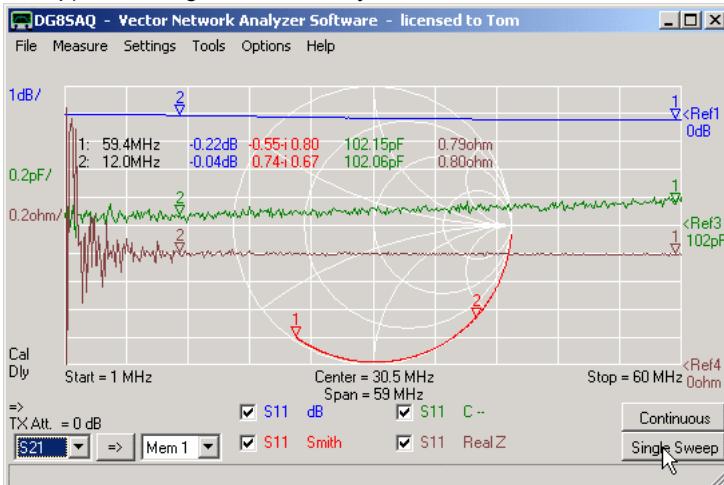
After the calibration measurement, open the calibration settings again:



Note, that next to the nominal capacitance value there should be a fairly accurate indication of the real capacitance measured during the calibration measurement. This capacitance value is determined from the last frequency point, so it may be off from the average capacitance, but not far. If this value is far off, then either the delay is wrong or you have picked the wrong standard.

We now enter the correct ESR for this capacitor of 0.8 Ohms (not exactly low loss, better use a high Q type).

Next, measure your LLC calibration standard as it was a DUT (i.e. not doing a calibration measurement but a normal sweep) still having the found delay switched on:



Note, that the LLC calibration step changes the measured series resistance of the capacitor to the value specified in the LLC standard setup, thus the LLC correction works.

►►► **Note:** LLC correction will work somewhat beyond the recommended frequency range, but it will completely fail, if the frequency is too far ABOVE the recommended frequency range. In this case, the software falls back to the standard SOLT correction ignoring the LLC calibration measurement.

►►► **Note:** If several LLC standards are defined and calibrated, then the software automatically use the standard best fitting the current frequency point. So, within a single sweep several LLC standards may be used for correction depending on the data point frequency.

►►► **Note:** The LLC correction will NEVER be saved into a master calibration! Master calibrations cannot contain LLC corrections. When saving a calibration containing LLC corrections into a master calibration, the LLC part will simply be omitted, but the SOLT corrections will properly be saved.

MAGI-CAL: AUTOMATIC CALIBRATION

Introduction

Normally, the VNWA calibration process involves a lot of connect/disconnect cycles of the various calibration standards. This is time consuming and prone to handling errors.

Therefore, the Magi-Cal ® device has been developed. It allows to perform an automatic one port or two port VNWA calibration with a single connect cycle and pressing a single button.



The Magi-Cal ® device contains two sets of reflect calibration standards for either rf port and a thru calibration standard potentially connecting port 1 to port 2. The standards can be connected internally to the two ports via USB control. An indicator LED displays the currently active standard, e.g. if the LED is lit red, then both ports present a short. Upon connecting the Magi-Cal ® to a computer USB socket, the LED indicates functionality of the device by briefly flashing up in the sequence red-orange-green-OFF.

Properties:

- automatic VNWA two port SOLT calibration device.
- for one-port and full two-port calibrations.
- frequency range: dc...1GHz, with reduced accuracy to 1.3GHz.
- supports all VNAs that are operated with the DG8SAQ VNWA software, e.g. VNWA2, VNWA3, N2PK-VNA, FA-VA5
- plug-and-play, supports plug/unplug at any time.
- requires VNWA software version 36.7.8 or above.
- uses VNWA LibUSB driver, no separate driver installation required.
- So far, no Linux/WINE support!

Connecting the device



As shown above, the Magi-Cal ® device is connected to the VNWA test leads such that port 1 connects to the VNWA TX port and port 2 connects to the VNWA RX port. If an S-parameter test set is connected to the VNWA, Magi-Cal ® can also perform a full two port calibration. In this case Magi-Cal ® port 1 must be connected to test set port 1 and Magi-Cal ® port 2 must be connected to test set port 2. Both, Magi-Cal ® and the VNWA must be connected to USB sockets of the same computer.

With this setup, Magi-Cal ® will generate a valid SOLT calibration with calibration planes coinciding with the connector mating planes, just like with using regular SMA calibration standards:



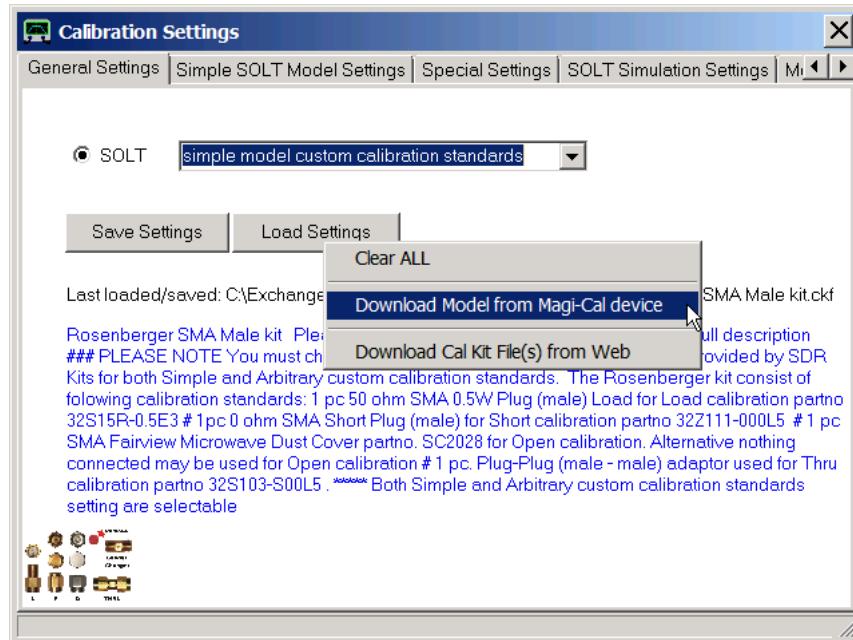
Usage

Preparation:

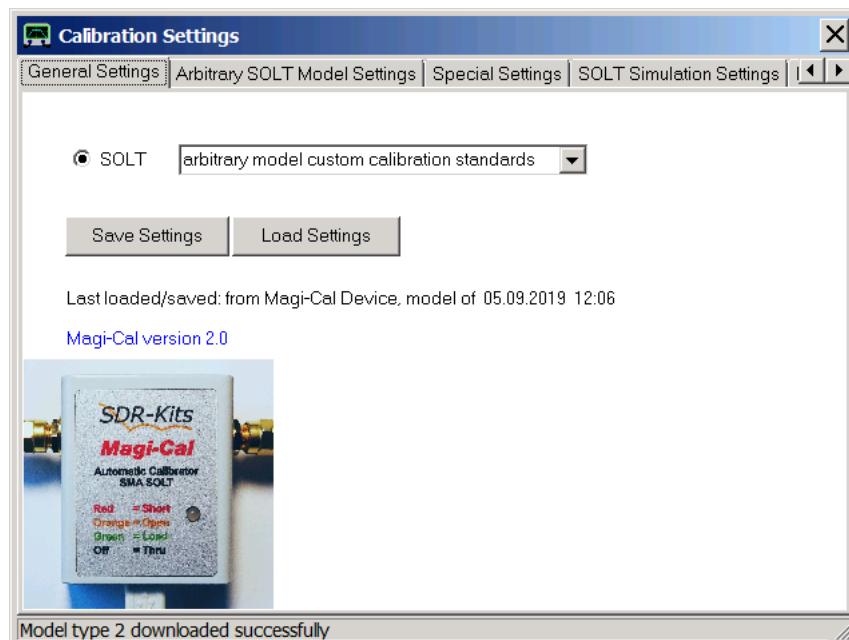
The Magi-Cal ® internal calibration standards' rf properties are far from ideal. Therefore, sophisticated device models have been developed to describe their behavior over the 1GHz frequency range. Moreover, every single Magi-Cal ® device has been characterized by SDR-Kits and the model parameters have been stored inside the Magi-Cal ® hardware itself. Thus, the device can be carried from one VNWA to another and always achieve a valid calibration.

The first step prior to calibration is to download the Magi-Cal ® model parameters from the actual Magi-Cal ® device into the VNWA software. This step has to be performed if prior to usage a calibration kit different from the actual Magi-Cal ® has been used/loaded. Once, the Magi-Cal ® model has been loaded, it remains valid until a different calibration kit model is being loaded.

The model download is achieved via the Calibration Kit menu, which is invoked via *Settings-Calibration Kit* or via keyboard short cut *K*.



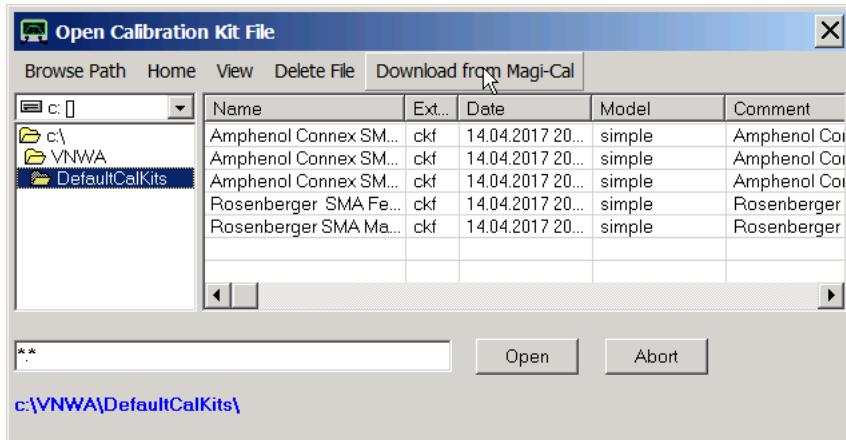
As seen above, the currently loaded calibration kit model is for a standard Rosenberger one. In order to change this to the connected Magi-Cal ® kit, **RIGHT-click** the *Load Settings* button. Beware, that this is not the normal Windows way to press a button, but this is where the **special load options** are found. If an operational Magi-Cal ® device is detected, the right-click menu will show the menu item *Download Model from Magi-Cal device*. Clicking it will initiate the download process:



Note, that Magi-Cal ® only works with the arbitrary model setting. Changing this setting will invalidate the model.

An alternative way to download the model from the Magi-Cal ® device is to left-click the *Load Settings* button the normal way, so the calibration kit file manager will open.

If an operational Magi-Cal ® device is being detected, the menu item *Download from Magi-Cal* will become visible. Clicking it will initiate the download process.



At the end, the *Calibration Kit File manager* will close automatically and the Calibration Kit menu will come back to focus, where the success of the operation can be observed.

►►► WARNING:

For a correct calibration, the Calibration Kit menu settings must match the actual calibration kit hardware.

Be aware, that the Calibration Kit menu settings can get changed by various operations, because they are also stored in any calibration file (*.cal) and in any instrument state file generated by the VNWA software.

Thus, **importing a calibration file or an instrument state** may change the Calibration Kit menu settings!

Prior to any calibration, check and make sure, that the Calibration Kit menu settings reflect the calibration kit hardware you are using!

Calibration:

Now, a SOL or SOLT calibration can be performed using the calibration menu which is opened via VNWA main menu *Measure-Calibrate* or using the keyboard shortcut C.

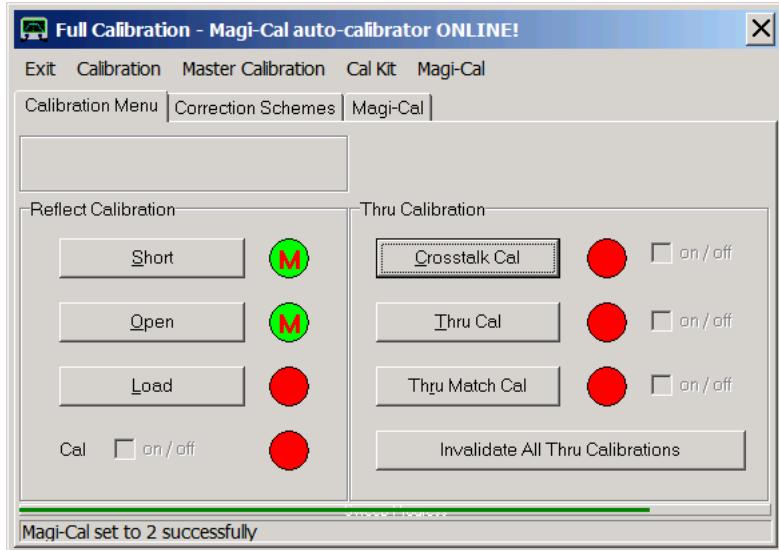
The automatic calibration process is started there via the *Magi-Cal* menu:



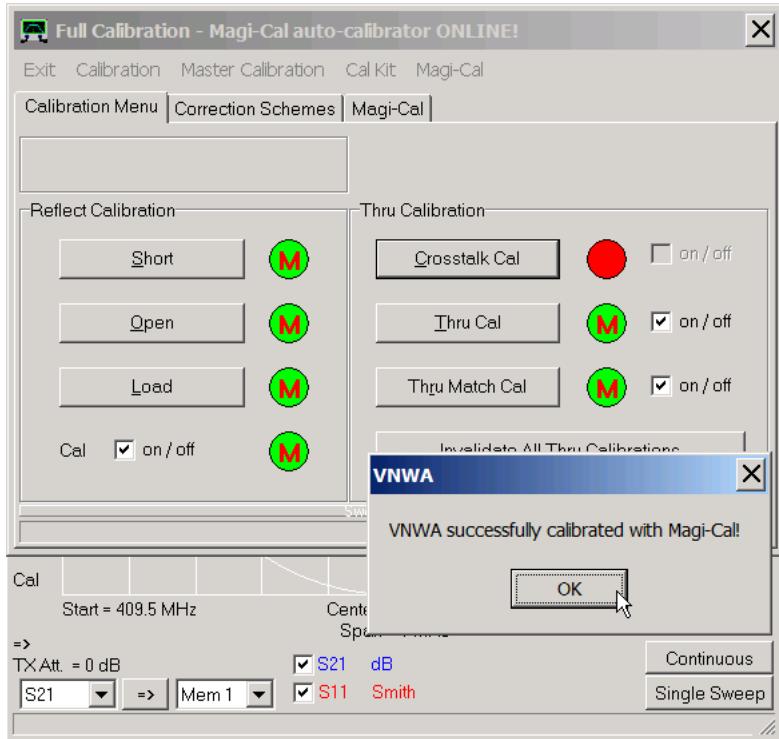
The user can select to automatically perform either a one port SOL or a two port SOLT calibration.

The calibration progress will be displayed by the sweep progress bar and the status of the red/green calibration

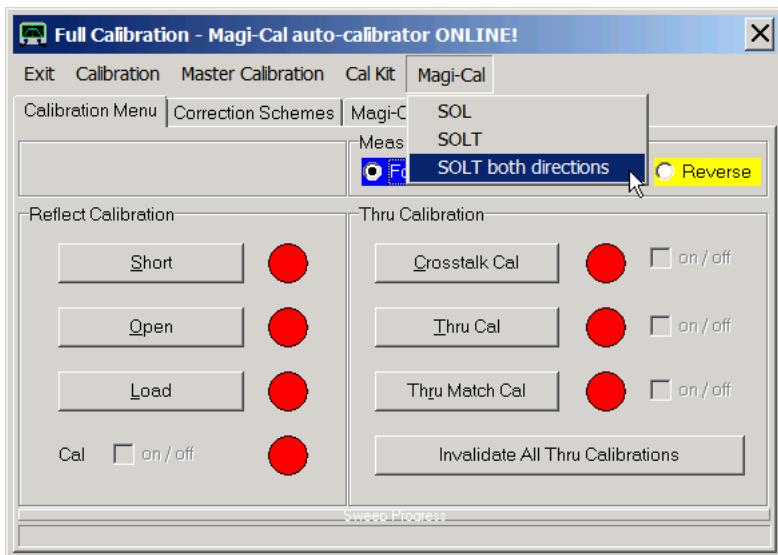
indicators:



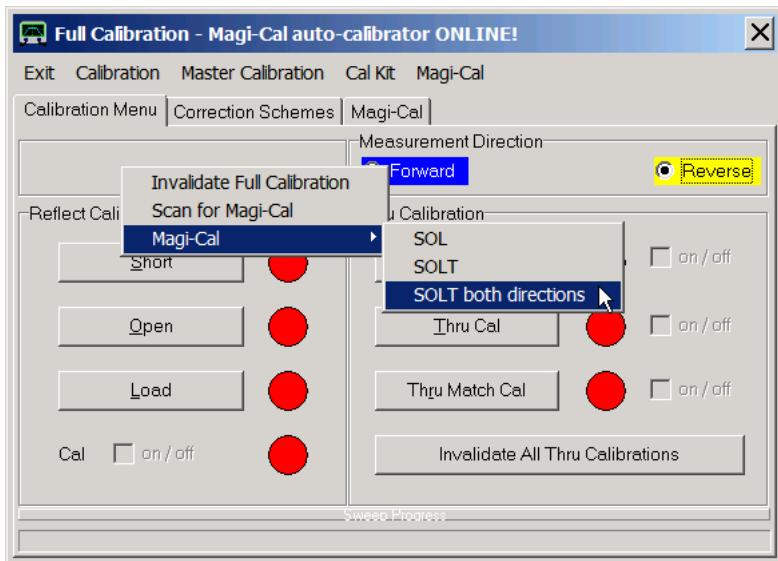
A success message will indicate the end of the calibration process:



If an automatic S-parameter test set is present, the Magi-Cal ® menu shows the additional menu item **SOLT both directions**, which allows to perform a full two port calibration automatically:



The Magi-Cal ® calibration functions can also be accessed by right-clicking the calibration menu window at any position:



An automatic full calibration can also be initiated with the keyboard shortcut **M**.

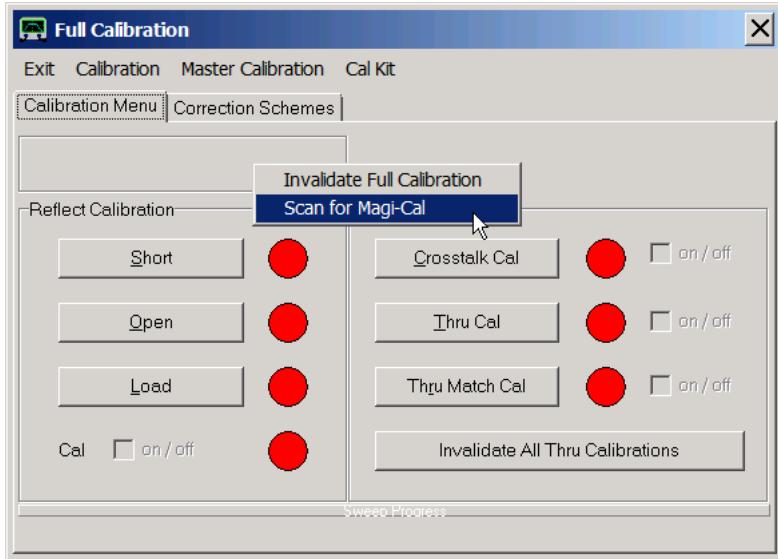
Note, that it is still possible to calibrate with each standard manually by pressing the according cal standard button. The corresponding standard will be switched online automatically.
Moreover, if an S-parameter test set is used, it is possible to automatically perform a SOL or SOLT calibration in one direction and leaving the opposite direction calibration untouched.
Also note, that a crosstalk calibration is not part of the Magi-Cal ® calibration process. If needed, remove the Magi-Cal ® device from the rf ports after automatic calibration and then perform the crosstalk calibration manually.

If a Magi-Cal ® device is detected by the VNWA software, the calibration menu will also show the Magi-Cal ® tab. This displays the device temperature and a stored reference temperature, neither of which is currently being used:



Troubleshooting

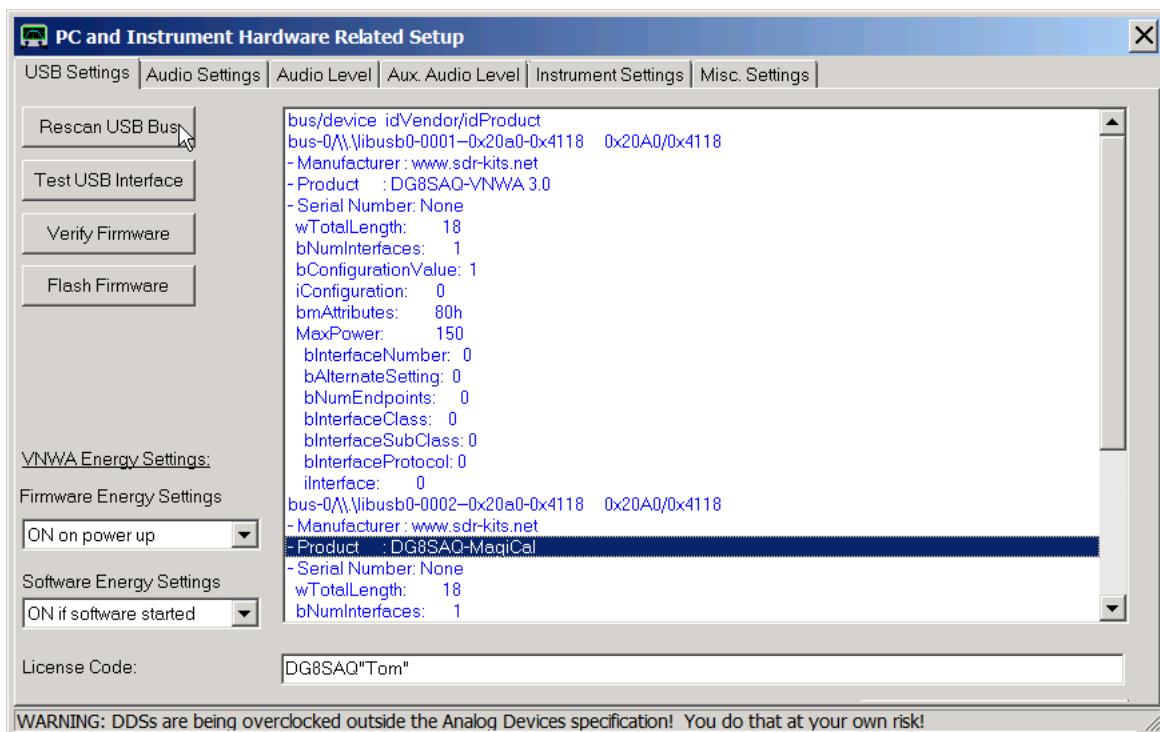
If the calibration menu window doesn't show the Magi-Cal ® items, scan for a Magi-Cal ® device by right-clicking the *calibration menu* window at any position and selecting the *Scan for Magi-Cal* menu item:



If the Magi-Cal ® items then won't show up, either the Magi-Cal ® software modules are missing or there is no Magi-Cal ® device present on the computer USB bus.

The Magi-Cal ® software modules can be installed by reinstalling the VNWA software using the automatic installer for software version 36.7.8 or higher.

You can check if a Magi-Cal ® device is present on the computer USB bus by using the VNWA setup menu. Pressing the *Rescan USB Bus* button in the *USB Settings* tab lists all LibUSB devices present on the computer:



Above screenshot shows both a VNWA3 and a Magi-Cal ® device (highlighted) being present.

MAGI-CAL AS PERMANENT CALIBRATOR

Introduction

The previous chapter explains, how the Magi-Cal automatic calibration device is used for calibrating the VNWA instead of using four individual short, open, load and thru calibration standards. In order to take a measurement of a DUT, the calibrator still needs to be removed and replaced by the DUT.

Now, in an automated monitoring system may arise the need to permanently keep the calibrator connected to the VNWA for repeated calibration and automatically switching in the DUT for a measurement while at the same time switching the calibrator off.

If the task is to perform only a reflect calibration and a subsequent reflect measurement (S11), both tasks can be performed with a single Magi-Cal device.

The idea is to use the Magi-Cal port 1 as calibrator, which stays connected to the VNWA TX port at all times. The DUT in turn is connected to Magi-Cal port 2. While the Magi-Cal Thru standard is switched in, the DUT is connected via the Magi-Cal Thru standard to the VNWA TX port. The basic setup is shown in below picture.



During the SOL-calibration, the DUT is automatically disconnected by the internal Magi-Cal thru switch going to off state, so the DUT won't interfere with the calibration process.

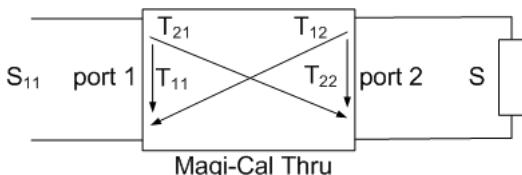
During measurement, the Magi-Cal is switched to Thru-mode, so the calibration standards are disconnected but the DUT is connected via the Magi-Cal Thru element to the VNWA TX port.

The only disadvantage of this approach is that the DUT is not connected directly to the calibration plane, which is the Magi-Cal port 1 position, but that the Magi-Cal thru element is connected in between and will thus distort the measurement result.

But, since the Magi-Cal is fully characterized (also the thru element), this effect can be compensated easily by use of mathematics. This compensation process is known as de-embedding, as the DUT can be thought of being embedded (in this case only on one side as it is a 1 port DUT) in a distorting network (here the Magi-Cal thru).

De-embedding the Thru element

As is shown in below schematic, our measurement setup consists of a calibrated VNWA measuring the reflection coefficient S_{11} of the combination of the Magi-Cal thru element characterized by its s-parameters ($T_{11}, T_{21}, T_{12}, T_{22}$) and the DUT with reflection coefficient S :



The measurement result S_{11} depends in the following way on the thru s-paramenters and the DUT reflection coefficient:

$$S_{11} = T_{11} + \frac{S \cdot T_{21} \cdot T_{12}}{1 - T_{22} \cdot S}$$

We do measure S_{11} , but we want to know the DUT reflection coefficient S . Thus, we have to solve above equation for S and obtain:

$$S = \frac{S_{11} - T_{11}}{S_{11} \cdot T_{22} - T_{11} \cdot T_{22} + T_{21} \cdot T_{12}}$$

So, we actually can obtain the DUT reflection coefficient from our measurement.

Examples

We have to define a custom trace for S with above equation:

The dialog shows the following fields:

- Expression:** $(s11-t11)/(s11*t22-t11*t22+t12*t21)$
- Global Subexpressions (available in all expressions, subexpressions may use other subexpressions from above):**

Name	Alias	Expression
Sub1 =	t11	= s_thrumatch
Sub2 =	t21	= s_thru
Sub3 =	t12	= s_thru
Sub4 =	t22	= s_thrumatch
- Aliases:**

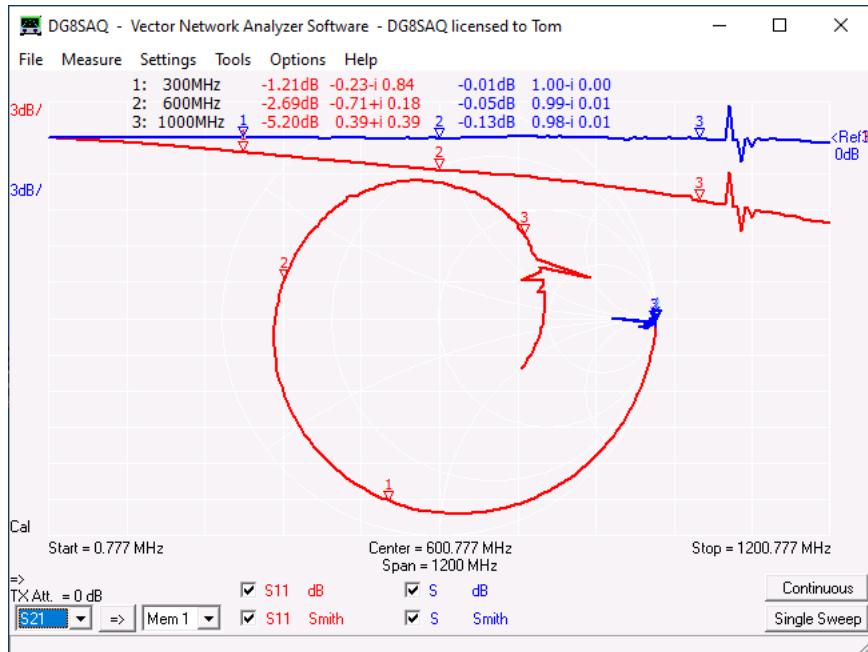
S21 =	S11 =	S12 =	S22 =	Caption:
Mem1 =	Mem2 =	Mem3 =	Mem4 =	Mem5 =
Mem6 =	Mem7 =	Mem8 =	Mem9 =	Mem10 =
- Buttons:** ok, Save, Load

Note, that the Magi-Cal Thru element is symmetric, thus $T_{11}=T_{22}=s_thrumatch$ and $S_{21}=S_{12}=s_thru$. Also note, that we have conveniently assigned the caption S for this trace.

Now, we take a measurement with Magi-Cal port 2 being unconnected:

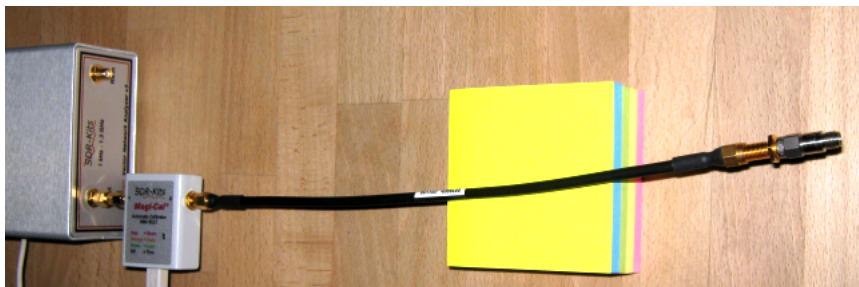


We obtain the following result:

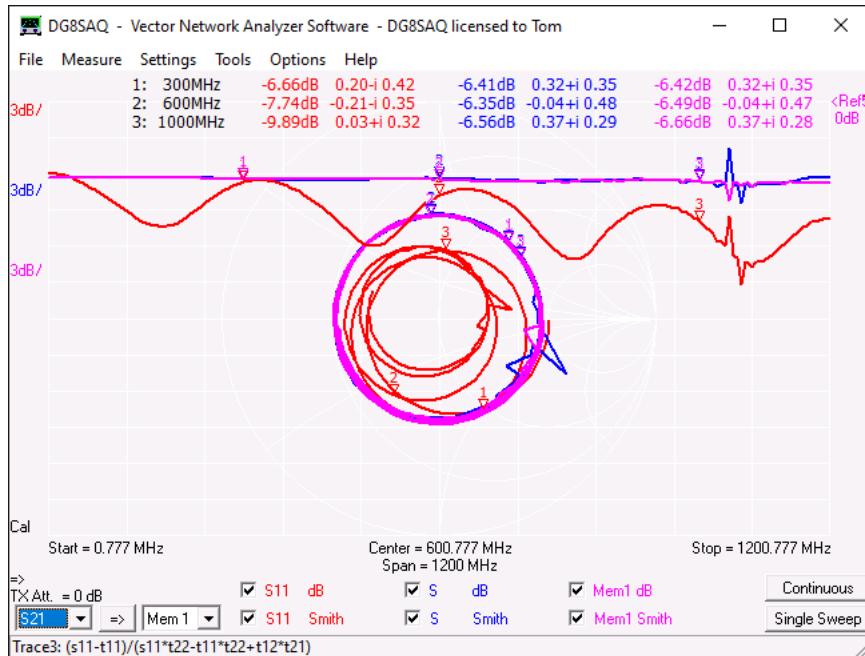


While S11 spirals in due to the increasing loss and phase shift of the Magi-Cal Thru, the mathematically corrected custom trace S shows a point at the open position (apart from some spurs at >1GHz). This proves that we have shifted the calibration plane to the Magi-Cal port 2 and that we have also corrected for the Magi-Cal Thru loss.

Next, we try with a DUT, which consists of a piece of transmission line which is terminated by a 3dB attenuator with its far end being left open. So, we expect this DUT to show a fairly flat return loss of 6dB since the signal travels the attenuator twice (forth and back).



The measurement yields...



Again, while the directly measured S11 (red) is clearly distorted by the thru element, the corrected trace S (blue) shows the expected behavior.

The pink Mem1 trace shows a conventional direct measurement of the DUT reflection coefficient for comparison. As can be seen, S and Mem1 match very well.

TCP/IP remote control example

The kind of measurement shown above is usually performed remotely and thus requires remote control of the measuring system.

VNWA offers a TCP/IP remote control interface. The following shows a TCP/IP example command sequence to perform such a calibration, measurement and evaluation. This requires VNWA software version 36.7.8.10 or newer.

TCP/IP command sequence:

```

command          (explanation)
...              (set up frequency grid and sweep speed here)
download-magi-cal-models (download magi-cal standard models)
calsweep open    (perform open calibration)
calsweep short   (perform short calibration)
calsweep load    (perform load calibration)
magical thru     (switch magical to thru mode)
sweep s11        (perform reflect measurement)
write-cust-expr (s11-s_thrumatch)/(s11*s_thrumatch-s_thrumatch^2+s_thru^2) d:\out.s1p (de-embed S11 and write result to local Touchstone file d:\out.s1p)

```

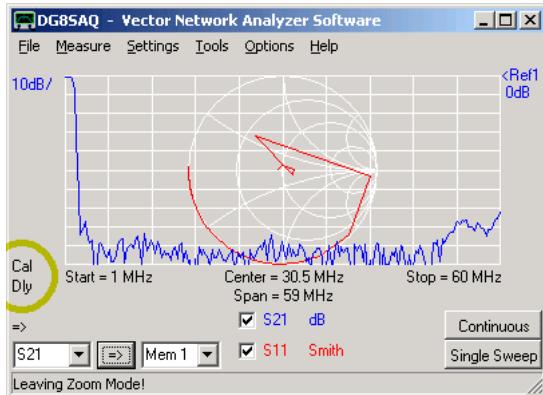
Note: Avoid any blank characters in the mathematical expression used in command write-cust-expr.

MEASUREMENT

Measurement:

Your instrument is now set up and calibrated and showing the main graphics window. It is ready for **sweeping**, which means measuring.

To do so, press the button "**Single Sweep**" if you only want to acquire a single frequency sweep or the button "**Continuous**" (F) for continuous sweeping. Both sweeps can be interrupted by pressing the same button again. The picture below shows the measurement result of a low pass filter.



Note the two labels which are circled yellow:

The **Cal**-label indicates a valid calibration. If that label text shows **MC** instead, there is no valid user calibration, but a Master Calibration is loaded. You can double-click the label to activate the calibration menu. Right-click to switch off the calibration. If the user only performs a thru calibration, the software combines the user thru calibration with the master reflect calibration (if available).

The "**Dly**" label indicates, that a port extension (delay) is being applied. Double-click it to enter the port extension menu or right-click to switch off port extensions.

Note the **arrow** below the yellow circle. It indicates the measurement direction:

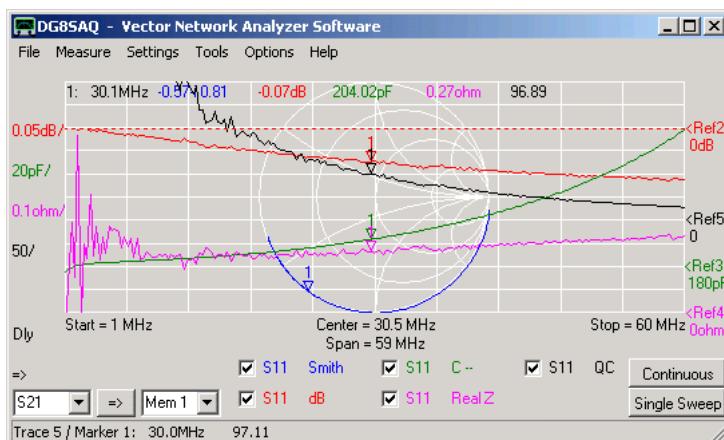
=> means forward measurement. Measurement data is stored in data spaces S11 and S21.

<=> means reverse measurement. Measurement data is stored in data spaces S12 and S22.

As the VNWA2.* hardware cannot switch the measurement direction, you need to turn around the DUT manually for a reverse measurement.

Display modes:

Data can be displayed in various ways. The software can recalculate S11 e.g. to **VSWR**, **impedances**, **capacitances**, **inductances**, **Q-values**... . The following screenshot shows a reflection measurement (S11) of a wired 180pF capacitor:



►► Hint: If you change the display type (e.g. from dB to VSWR) and you won't see your trace because of unfit scales,

the fastest way to see something is to perform an **Autoscale** operation. To do so, right-click the according scale-label and select autoscale. **Scale to Ref** is an alternative to automatically scale the graph. It keeps the scale per division untouched but aligns ref level to the maximum of the trace.

Markers

Up to 9 markers can be used. To add a marker, right-click onto the main grid at the frequency, where you want the marker to show up and select "**add normal marker**".

Available marker types:

- *normal marker*
- *maximum marker* (automatically jumps to the maximum of the first displayed trace)
- *minimum marker* (automatically jumps to the minimum of the first displayed trace)
- *absolute markers* (places markers at specifiable absolute levels)
- *delta marker* (displays the frequency distance and the vertical distance to the last normal marker)
- *bandwidth marker* (places a maximum marker, two markers down one vertical division below the maximum marker and a center marker between the bandwidth markers)

►►► **Hint:** To see a summary of marker positions or change the marker frequency, double-click onto a marker.

►►► **Hint:** You can also drag the markers around with the mouse.

►►► **Hint:** To **delete** one or all markers, right-click into the display grid and select "delete last marker" or "delete all markers".

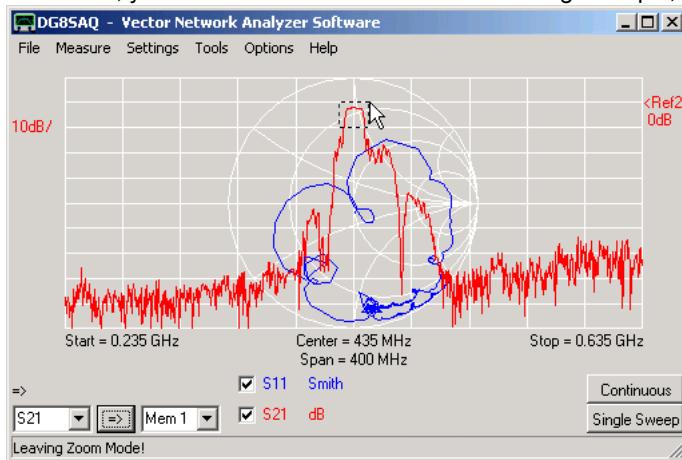
►►► **Hint:** You can enter an exact marker frequency after double-clicking onto the marker. The following window will open:



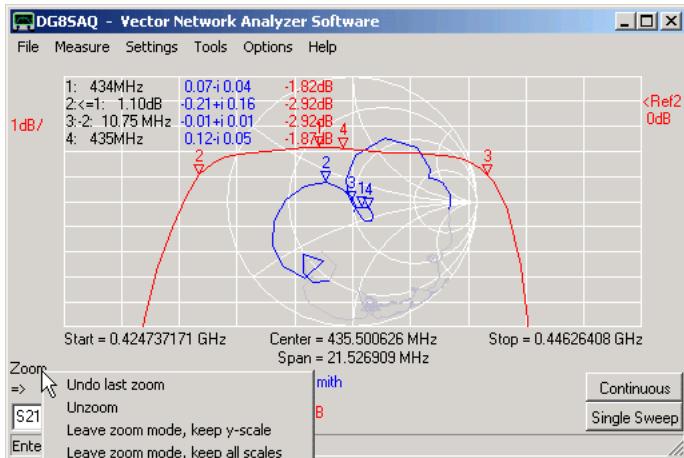
►►► **Hint:** In the above picture, you also see one of the two available horizontal dashed **reference lines**. You can move them up and down with the mouse. You can change their color by right-clicking on them.

Zoom

You can also zoom into a trace. To do so, left-click to one corner of an imagined box in the main grid, which you want to enlarge and draw the so called zoom-box with the mouse. While you keep the left mouse button pressed and you move the mouse, you will see the zoom-box. In the following example, we attempt to zoom into the measured filter's passband:



As soon as you release the mouse button, the zoom-box will be enlarged to completely fill the main grid.



You see the zoomed passband. Note, that I have added **bandwidth markers** in the above image as an example.

►►► **Note:** You can do multiple **consecutive zooms**.

►►► **Note:** When zooming, the measurement span and measured number of data points remains untouched. Only part of the data is displayed. When sweeping in zoomed state, you still sweep the whole unzoomed frequency range, but only part of it is displayed.

►►► **Note:** You cannot zoom inside the Smith chart. But as you can see in above screenshot, the Smith chart data outside the zoomed frequency range is greyed.

►►► **Note:** You can **unzoom** by right-clicking the **zoom label** near mouse pointer. If you right-click it, the above shown **unzoom menu** pops up.

You can

- ***Undo last zoom*** = return to the state before the last of consecutive zooms
- ***Unzoom*** = restore the original x- and y-scales before all consecutive zooms
- ***leave zoom-mode keep y-scale*** = restore the full frequency span but keep the zoomed vertical scale
- ***leave zoom-mode keep all scales*** = keep the zoomed frequency span and the zoomed vertical scale. The outside data is lost, the visible data is interpolated to the full number of data points grid. A sweep after this will only sweep the visible frequency span.
- ***Overlay Unzoomed*** = if selected, overlay an unzoomed and greyed version of the zoomed data with the zoomed range highlighted.

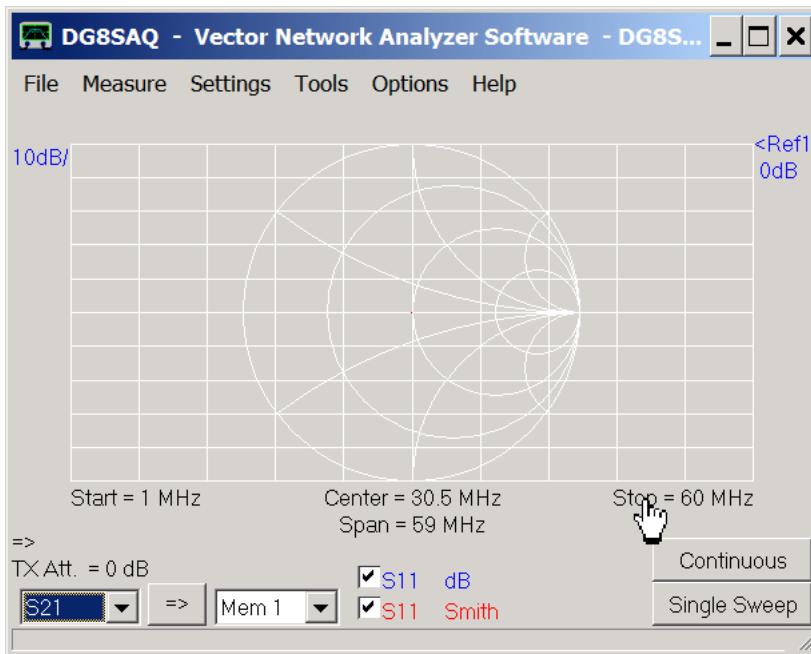
►►► **Hint:** You can also unzoom by right-clicking the main graphics grid.

EXAMPLE: A SIMPLE ONE PORT MEASUREMENT

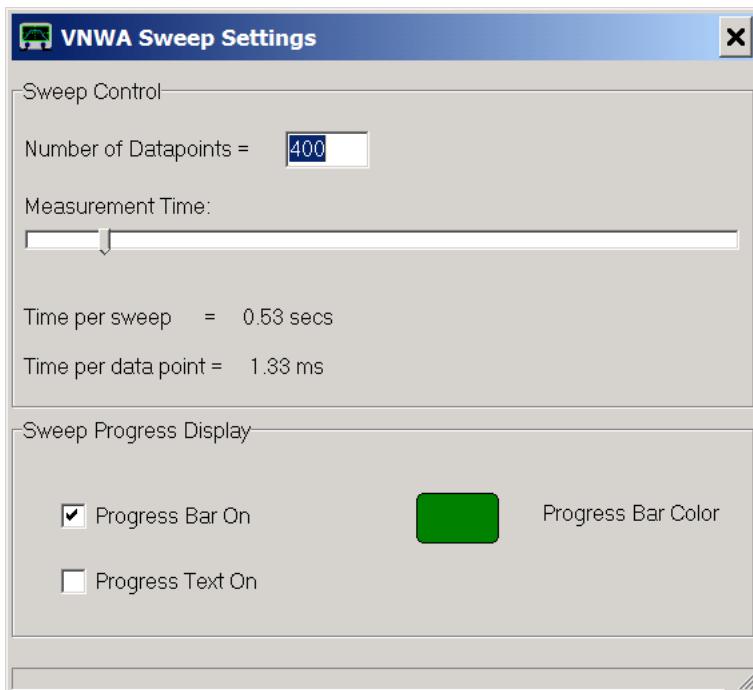
The following example demonstrates how the VNWA can be used to measure a little loop antenna for 13.5 MHz and extract its electrical properties.

1. Setting up

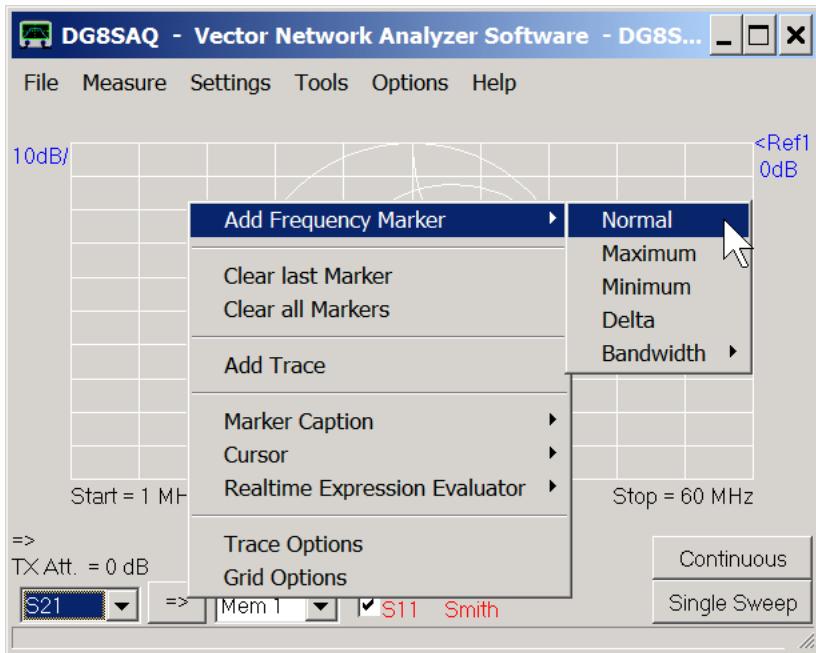
We set the instrument to the desired frequency range e.g. by double-clicking the "Stop" label or by simply pressing the f-key on the keyboard.



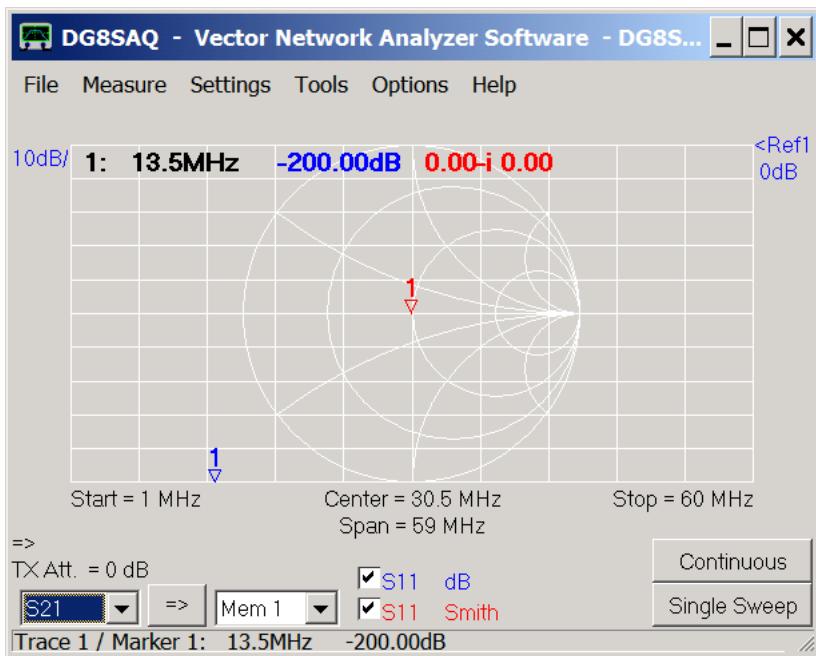
Also, select S11 for both traces as shown above. Select Settings-Sweep or press the s-key on the keyboard to choose reasonable sweep parameters:



Next we place a marker by right-clicking onto the main drawing grid:



Here it is:



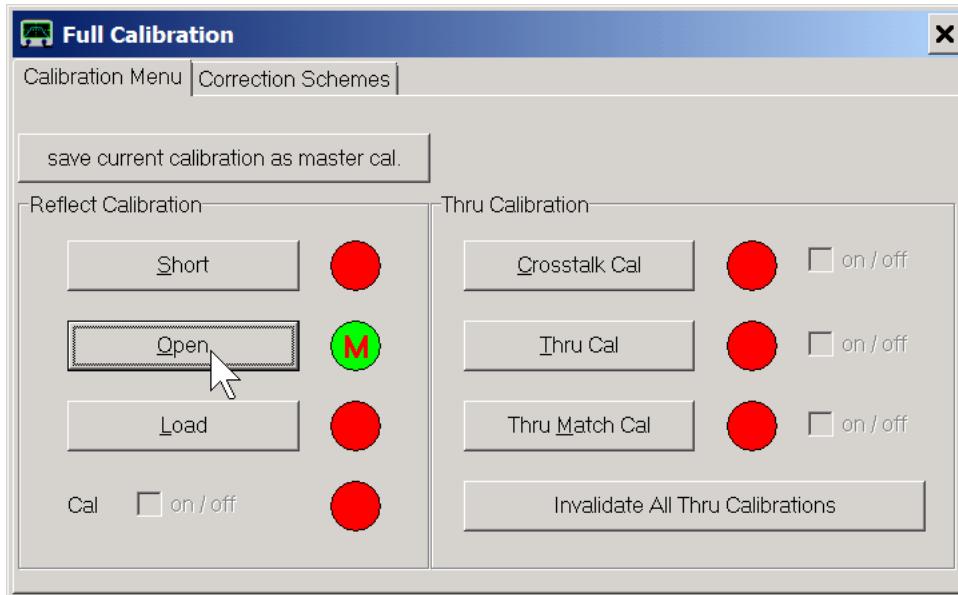
Now we are ready to calibrate the VNWA for a one-port measurement, also called reflection measurement.

2. Calibration

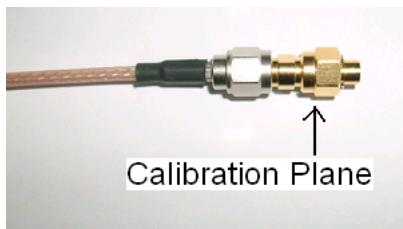
This example demonstrates how to calibrate a male test cable end with male calibration standards. We connect our test cable to the VNWA TX port and we connect our open calibration standard (in this case a female-female through connector) to the other end of the test cable:



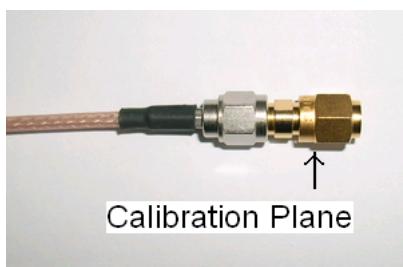
We select "Measure-Calibrate" or simply press the c-key on the keyboard, press the "Open" button, confirm that the open standard is connected and observe the instrument measuring the standard.
When the measurement is completed, the red Open light turns green:



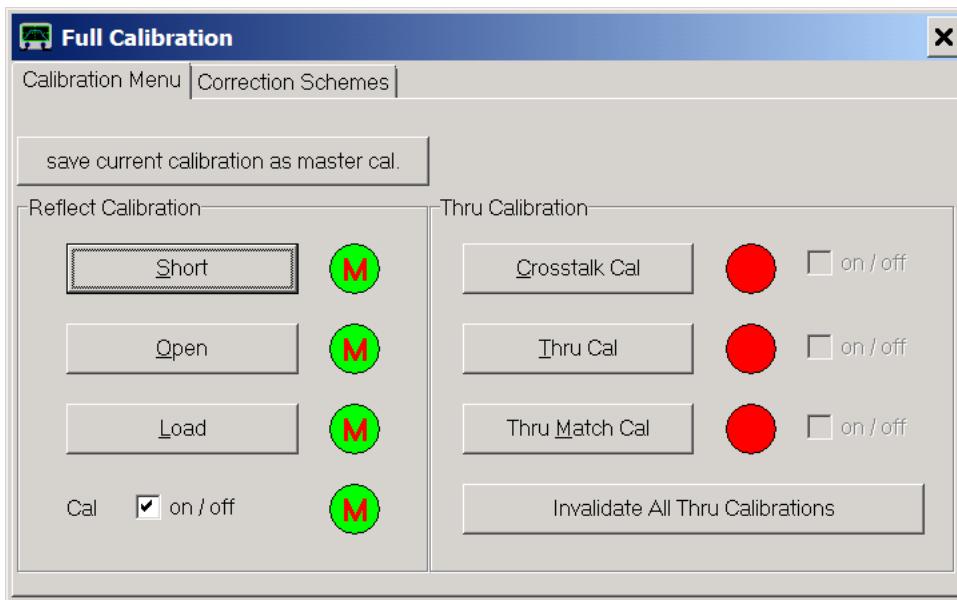
In similar fashion we measure the load calibration standard...



... and the short calibration standard:

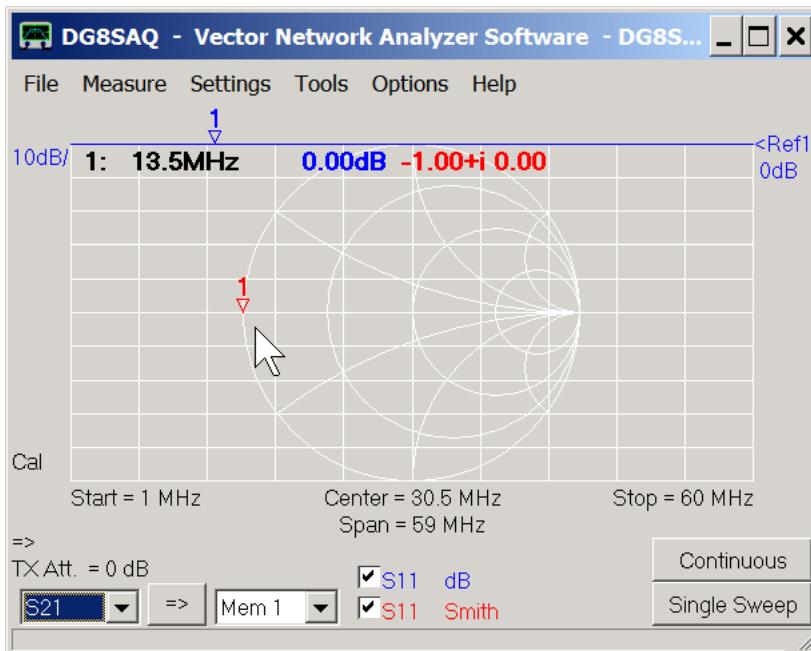


Finally, all reflect calibration lamps should be green:

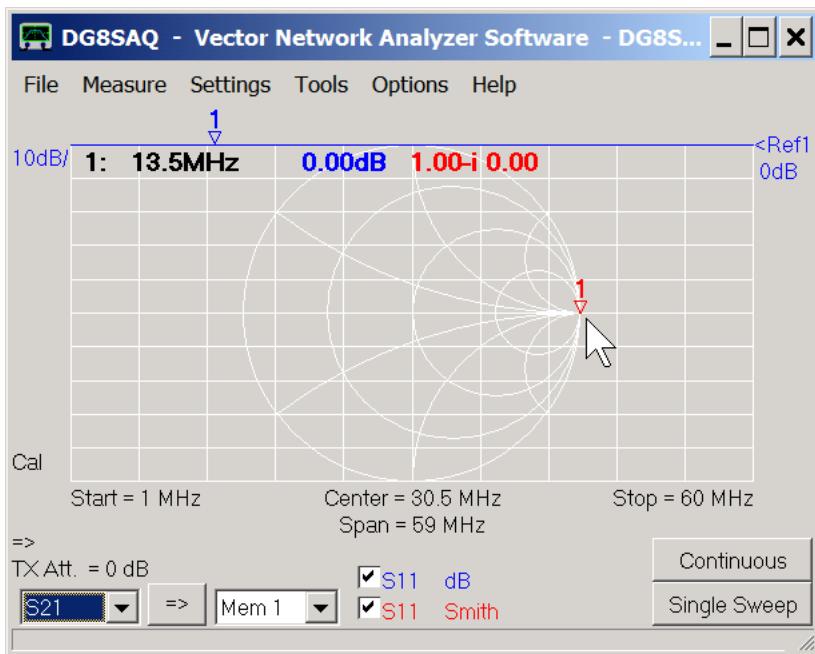


We close the calibration window and double-check the calibration by measuring the calibration standards. Observe the red Smith chart trace.

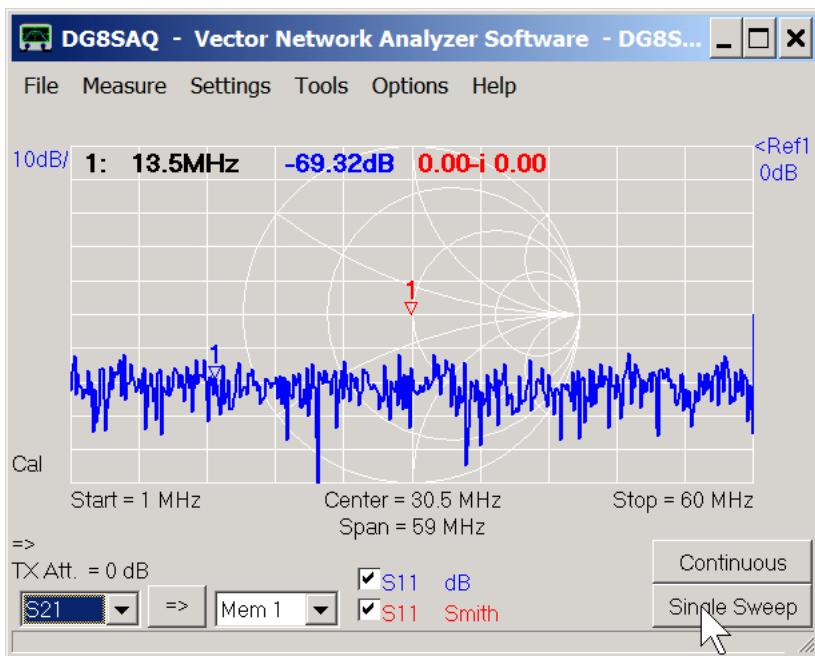
The short calibration standard should yield a dot at the left edge of the Smith chart (reflection coefficient -1, red marker value). Without the marker, the dot would be barely visible:



The open calibration standard should yield a dot at the right edge of the Smith chart (reflection coefficient +1):



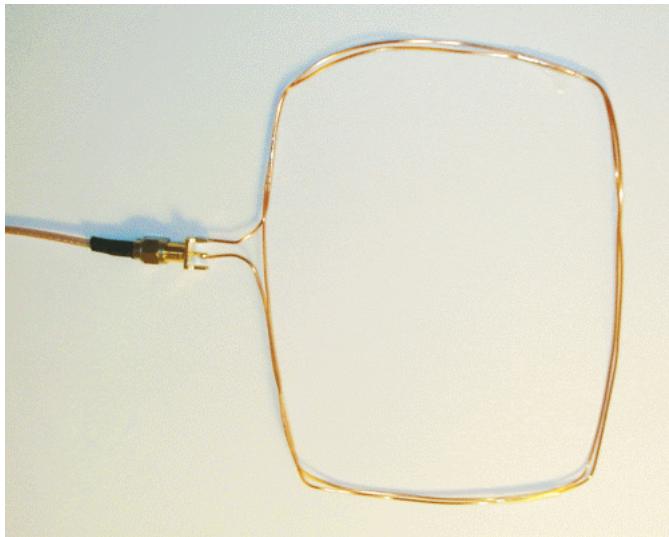
The load calibration standard should yield a dot at the center of the Smith chart (reflection coefficient 0):



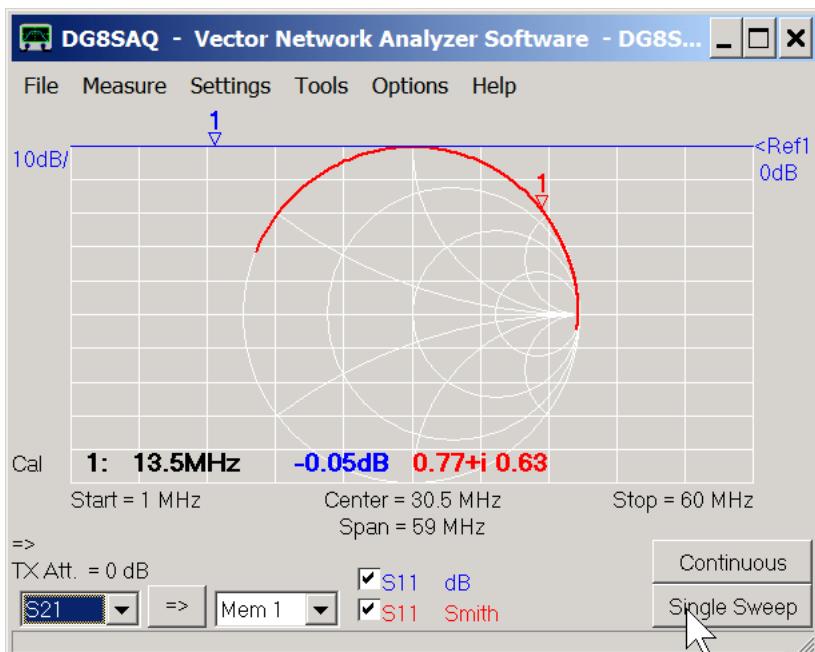
The dB scale trace shows that the measured reflection coefficient is not exactly zero but very very small due to noise.

3. Measurement

Next, we connect our loop antenna to be tested to the end of the test cable...



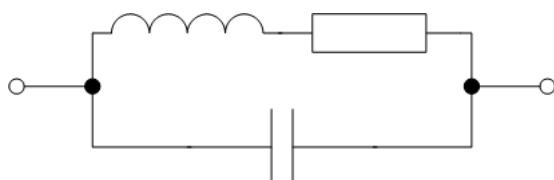
... and measure it:



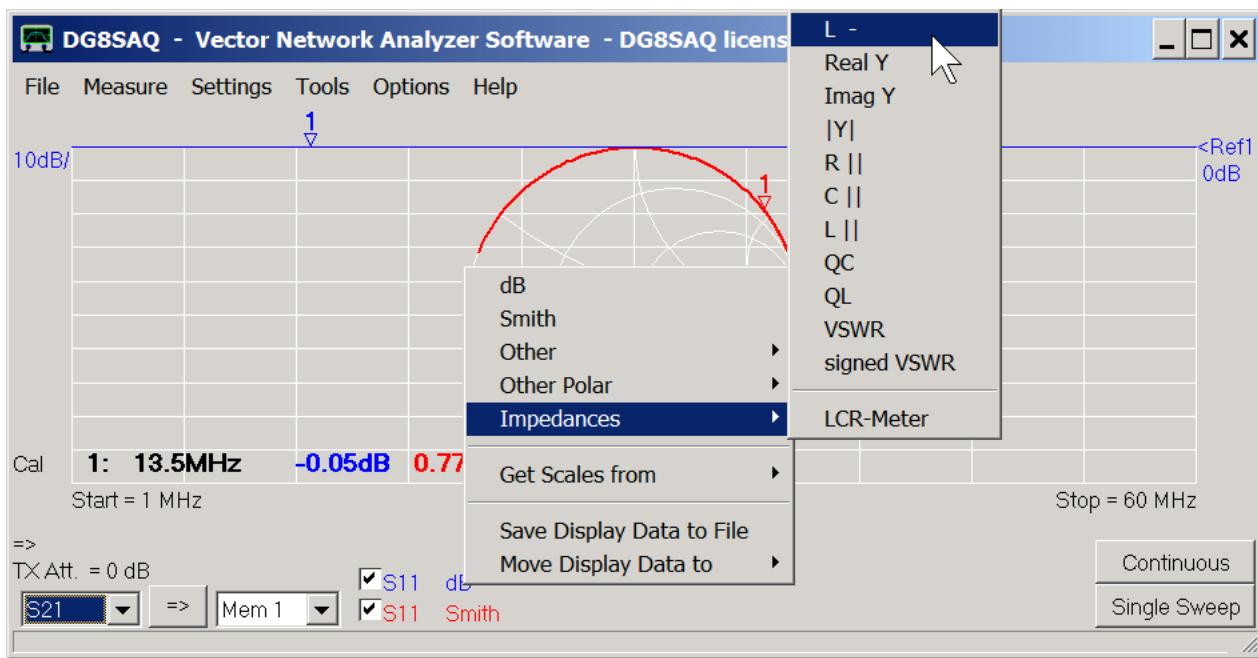
The antenna looks pretty much like an inductor with the red Smith chart trace circling mostly at the upper edge of the Smith chart. The blue trace shows that almost all power is reflected by the antenna (absolute value of reflection coefficient close to 1 = 0dB) if connected to the 50 Ohms source impedance of the VNWA.

4. Parameter extraction

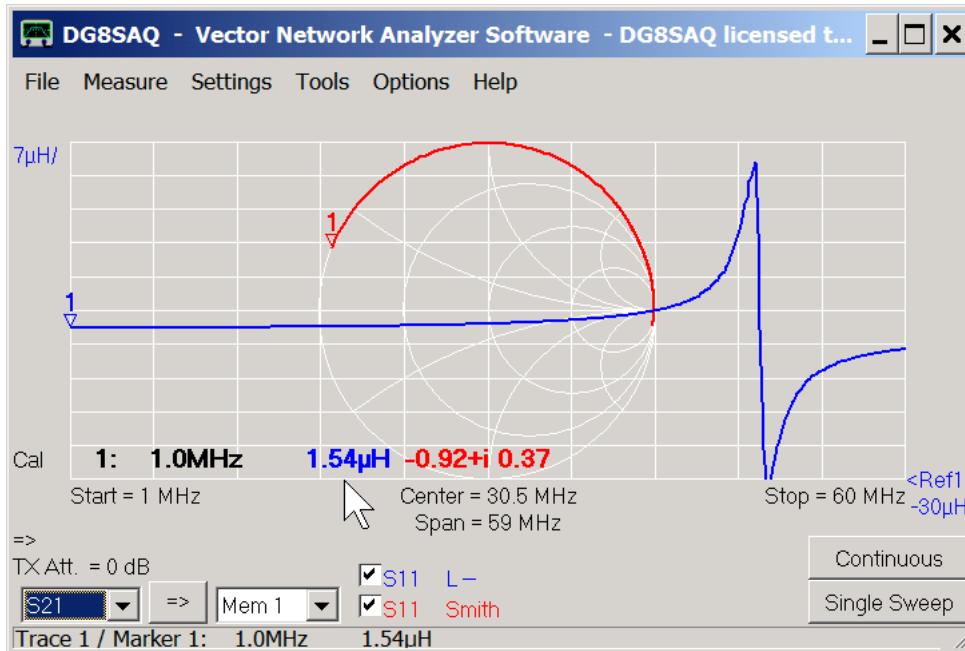
The antenna can be described quite accurately by the following equivalent circuit:



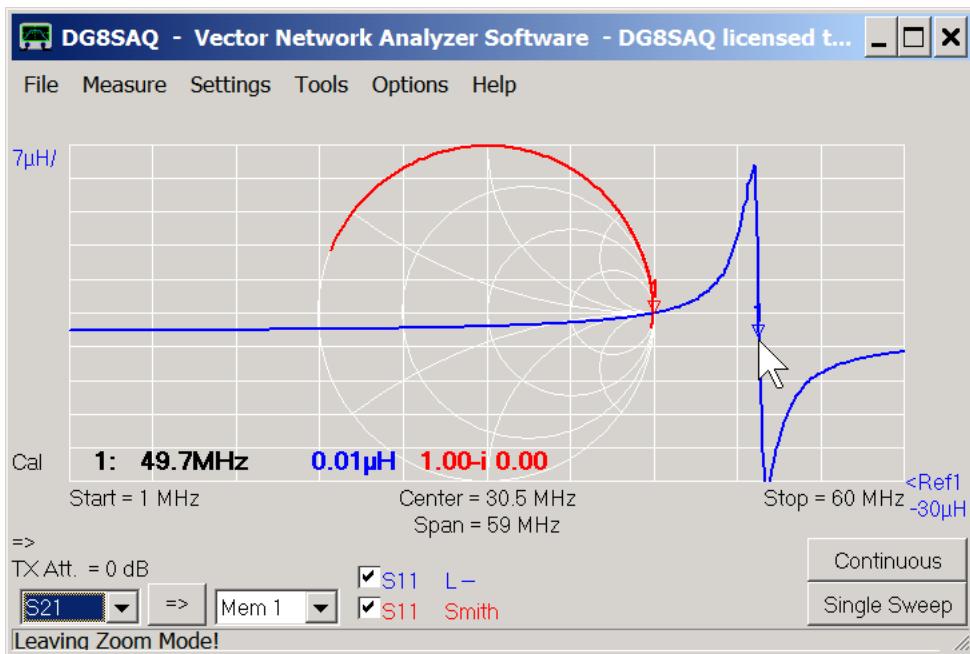
In order to obtain the inductance value, we switch trace 1 to display the series inductance calculated from the measured reflection coefficient S11 by right-clicking onto the dB label on the bottom:



The dominating contribution to the reflection coefficient at low frequencies comes from the inductance. We therefore move the marker with the mouse to the left and read off the inductance value at low frequency to yield $L=1.54\mu H$:



Next, we read off the parallel resonance frequency caused by the parallel capacitor by finding the frequency where the inductance value becomes zero between the resonance peaks to be $f=49.7MHz$.

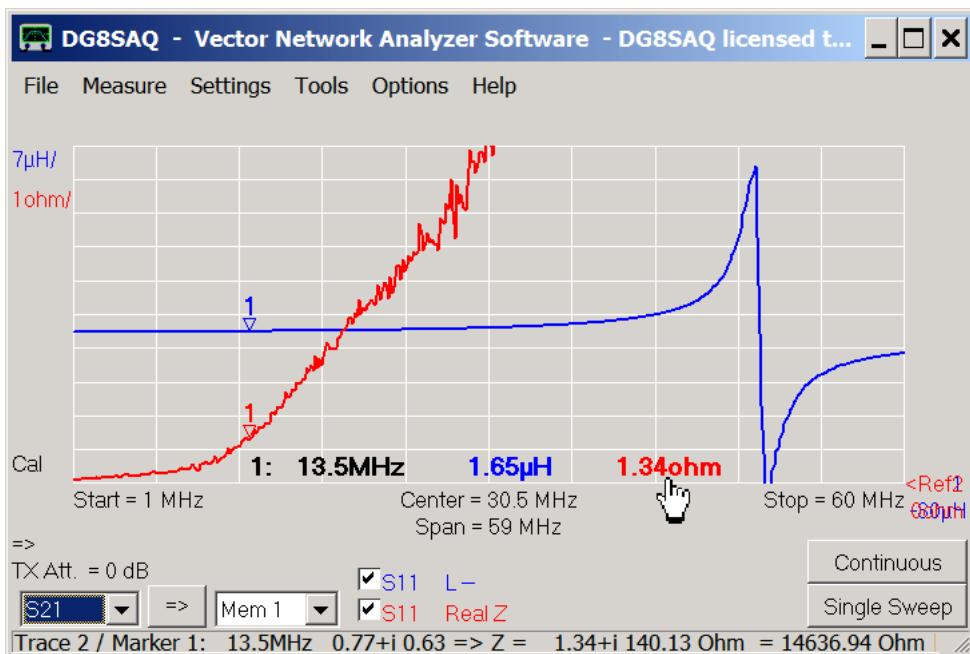


From this we can calculate the capacitance value using the parallel tank circuit formula ...

$$2\pi f = \frac{1}{\sqrt{LC}}$$

... to yield C=6.7pF.

Finally, we determine the real part of the antenna impedance at the nominal transmit frequency of 13.5MHz...



... to yield 1.34 Ohms. This impedance value is also shown in the bottom status line after right-clicking a marker. The first number pair is the reflection coefficient $S_{11}=0.77+i 0.63$. Its magnitude is close to 1, which means almost all power is being reflected. The last number is the first part of the (due to small window width truncated) number pair indicating the impedance in the parallel equivalent circuit, i.e. $14.6k\text{Ohms} \parallel -84\text{pF}$. A negative parallel reactance means the device is inductive. If you compensate the inductance by connecting 84pF in parallel, the resulting tank circuit will resonate at 13.5MHz and yield an impedance of 14.6kOhms in resonance.

These values can be used to calculate a matching network in order to match our antenna to e.g. 50 Ohms output impedance of a transmitter.

EXAMPLE: SIMPLE TWO PORT

The following chapter explains what a two port device is and how it is being described. Next, actual examples of how two port device measurements are being performed. Finally, the measured data is being analyzed in order to find out what we can do with it.

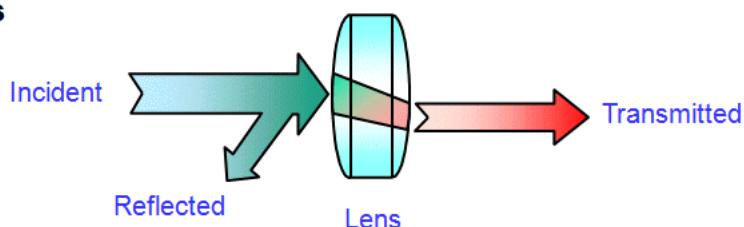
1. Principles of 2-Port device parameters and their measurement
2. Practical example on performing a 2-Port measurement
3. Interpretation and Usage of 2-Port measurements

PRINCIPLES OF 2-PORT DEVICE PARAMETERS AND THEIR MEASUREMENT

Electrical two port devices are devices with two ports or two connectors, where a signal can be sent into one port and be detected at the other port. Typical examples are attenuators, amplifiers and filters.

The description of electrical two port devices follows closely the description of optical devices like a lens. A lens is also a two port device, where a beam of light shines onto one side of the lens. Some of the incident light gets reflected back, some of the light entering the lens will be absorbed and converted to heat, while the rest will be transmitted from the other side:

Optics

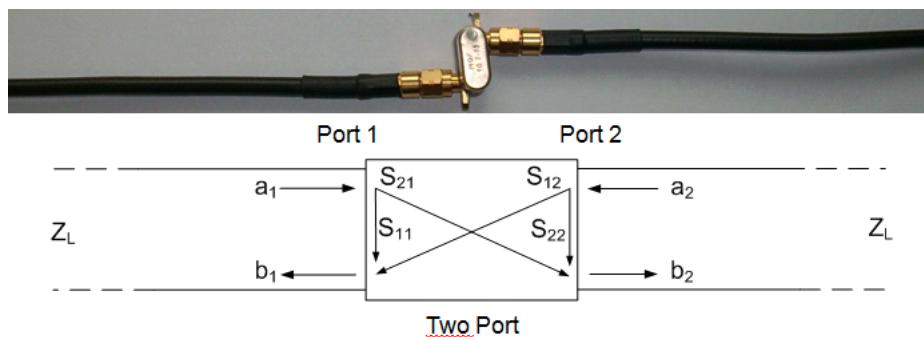


Electro-magnetics



An electrical two port device behaves quite similarly, when inserted between two transmission lines, where the left one is guiding an electrical wave into the device. Some of the electrical power might get reflected at the device input (reflection loss), some will be converted into heat inside the device (dissipation) and the rest will come out of the device at the right hand side.

As a lens can be used to shine light through it in both directions, also an electrical two port device can be operated in reverse direction. Thus, waves may enter the device from both sides (incident waves). Likewise, waves will leave the device from both sides (evanescent waves). Electrical two port devices are being characterized in terms of these incident waves denoted by **a** and evanescent waves denoted by **b**. The two ports are being numbered (1=left, 2=right) and the waves carry subscripts denoting on which side of the device they exist, e.g. a_1 is an incident wave (**a**) coming from the left (1).



The complex wave amplitudes a_1 , a_2 , b_1 and b_2 are complex numbers that carry both the wave amplitude and the wave phase information.

For linear devices there are linear relationships between these:

$$b_1 = S_{11} \cdot a_1 + S_{12} \cdot a_2$$

$$b_2 = S_{21} \cdot a_1 + S_{22} \cdot a_2$$

The complex numbers S_{11} , S_{12} , S_{21} and S_{22} are called scattering parameters or S-parameters. These fully

characterize the two port device for linear (=small signal) operation.
Now, we will investigate the meaning of these S-parameters.

Consider a situation where there is a wave a_1 being sent into the left side port 1 of our two port device, but nothing is sent into the right side port 2, i.e. $a_2=0$.

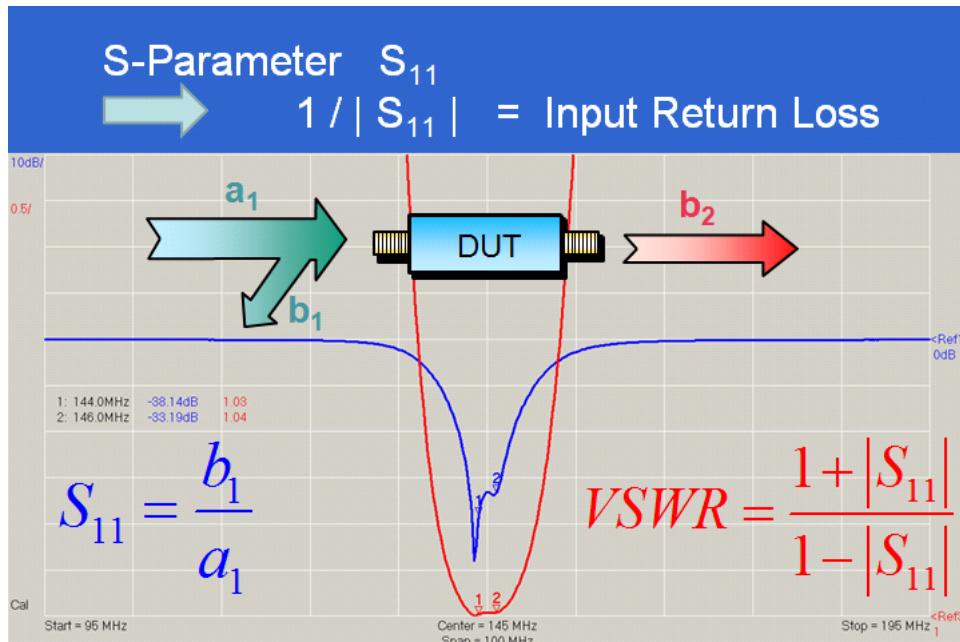
In this case the above equations simplify to:

$$b_1 = S_{11} \cdot a_1$$

$$b_2 = S_{21} \cdot a_1$$

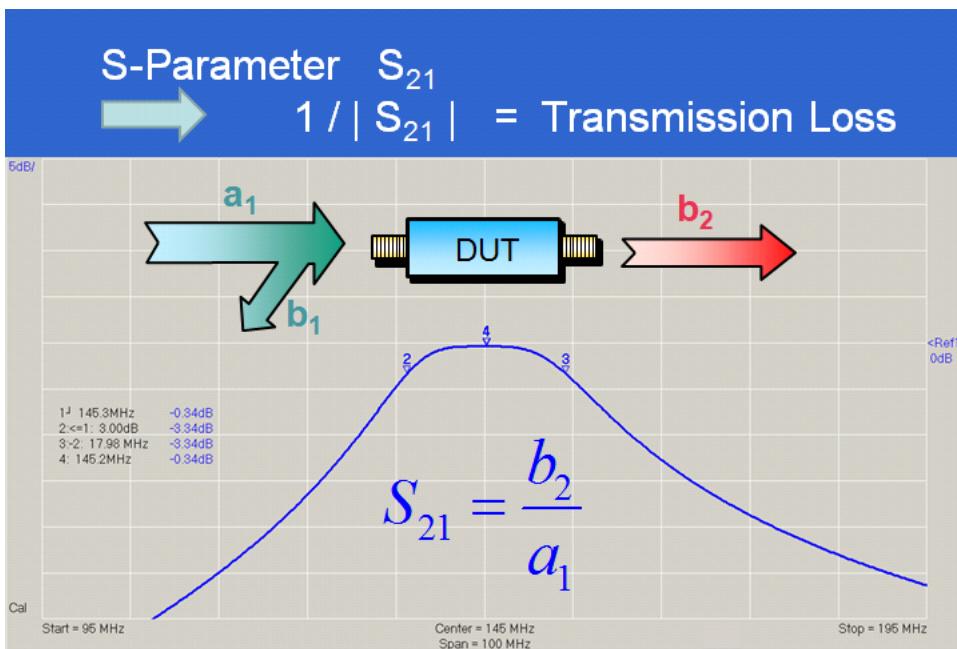
These simplified equations allow us to interpret the S-parameters as well as to understand how to measure them.

Let's look at the first equation. Solving it for S_{11} shows, that S_{11} is the ratio of reflected wave b_1 to incident wave a_1 at port 1:



The inverse of the magnitude of S_{11} is known as input return loss. I.e. a device showing a return loss of 10dB will yield $|S_{11}|=-10$ dB. This means, the reflected wave is 10dB weaker than the incident wave or 1/10th of the incident power is being reflected. But, since S_{11} also carries the phase information, also the input impedance of our DUT can be calculated from S_{11} .

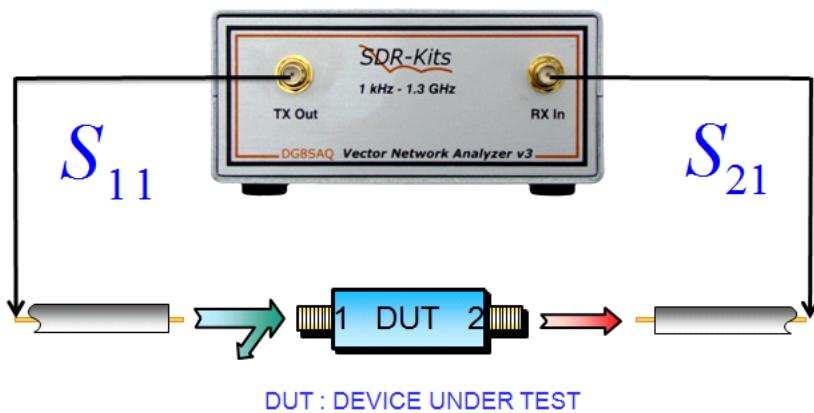
Solving the second equation for S_{21} shows, that S_{21} is the ratio of the transmitted wave b_2 to the incident wave a_1 :



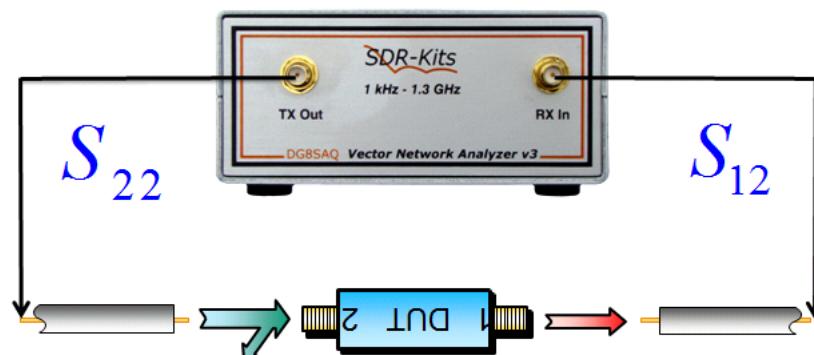
The inverse of $|S_{21}|$ is known as transmission loss. For example, a 10dB / 50 Ohms attenuator will show $S_{21}=-10\text{dB}$.

Now, it is clear that in order to measure S_{11} and S_{21} we must feed power into port 1 and detect the incident wave a_1 , the reflected wave b_1 and the transmitted wave b_2 .

This is exactly what the VNWA is doing:



If we want to know S_{12} and S_{22} , we can measure this in a similar fashion by sending power into port 2. As the VNWA in its standard version is unidirectional, i.e. only the TX port can deliver a signal, we have to interchange ports 1 and 2 of our test object, i.e. turn it by 180 degrees in order to measure S_{12} and S_{22} :



But why should we be interested in how our device performs in reverse direction, when we will only operate it in forward direction in the final application?

There are several answers to this.

Imagine, your test object is a low pass filter to be placed in between a power amplifier and an antenna. Clearly, only the power amplifier will feed power into the low pass filter and all that counts is how much power arrives at the antenna. So, S₂₁ being a measure for the transmission loss is crucial. But, as we do not want power to be reflected back into the power amplifier, we have to make sure that amplifier and filter are impedance matched. So, S₁₁ being a measure for the filter input impedance is crucial. Likewise, we want the antenna to be impedance matched to the filter output. Thus, S₂₂ being a measure for the filter output impedance is crucial, too, even though we won't actively feed power into the filter output. If there is no perfect match between filter output and antenna, which will lead to signal being reflected back into the filter, S₁₂ will become important, too.

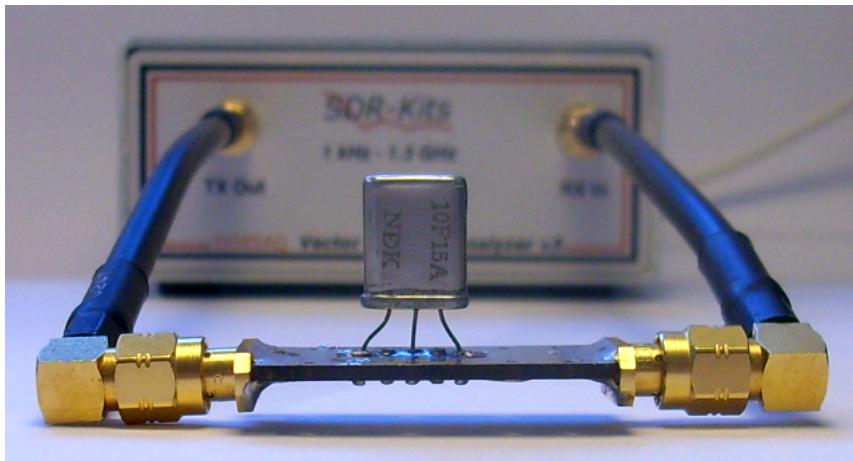
Another scenario is a filter, which is to operate in the final application in an impedance environment very different from the 50 Ohms environment of the VNWA like a high impedance crystal filter. The VNWA can only measure the filter S-parameters in its 50 Ohms environment. Once, the full set of S-parameters is known, it is possible to use these to simulate and optimize the filter performance in any final application with an arbitrary impedance environment. Such an example will be shown in page "Interpretation and Usage of 2-Port Measurements".

The following page "Practical Example on Performing a 2-Port Measurement" will present a step by step example on how the full set of S-parameters is being measured.

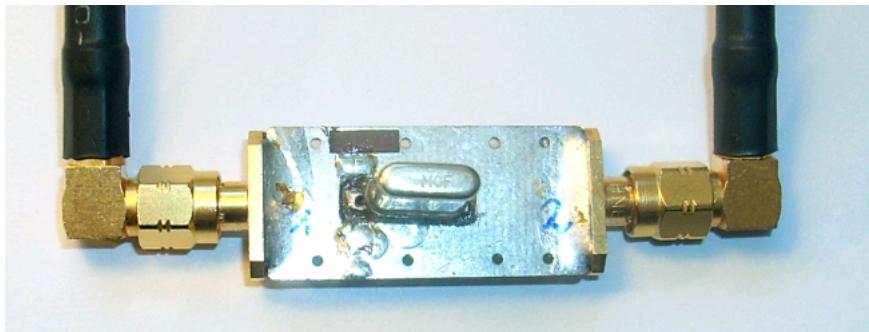
PRACTICAL EXAMPLE ON PERFORMING A 2-PORT MEASUREMENT

The following example demonstrates how the VNWA can be used to measure two port S-parameters of a narrow band 10.7 MHz two pole crystal filter.

We connect the filter using test cables and a test board between the VNWA TX port and RX port:



It is recommended that you number your test object ports to make very sure that you always measure in the proper direction:

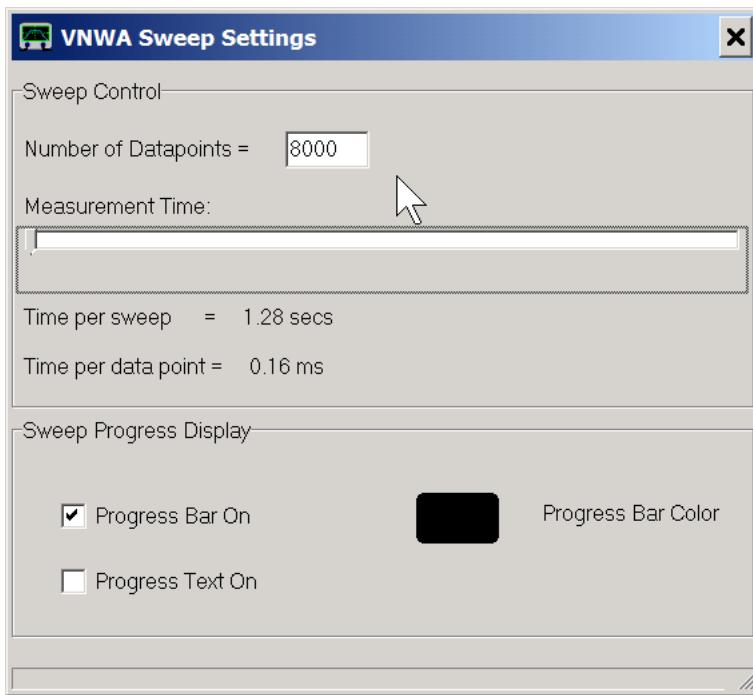


| 1. Setting up

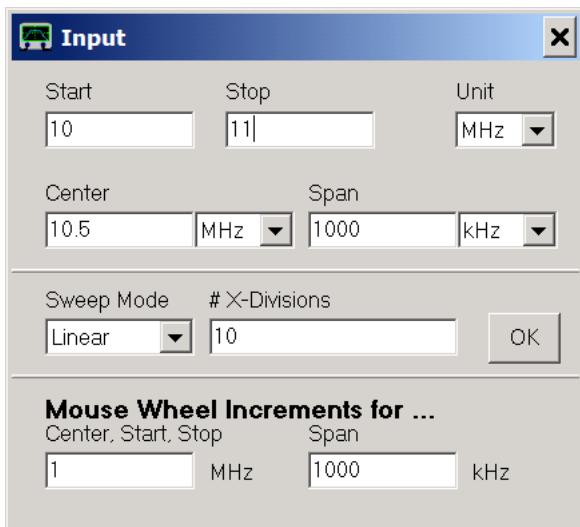
First, we need to set a frequency range reasonably covering our test object.

To find it, we will do a first transmission measurement without calibration over a relatively large frequency span.

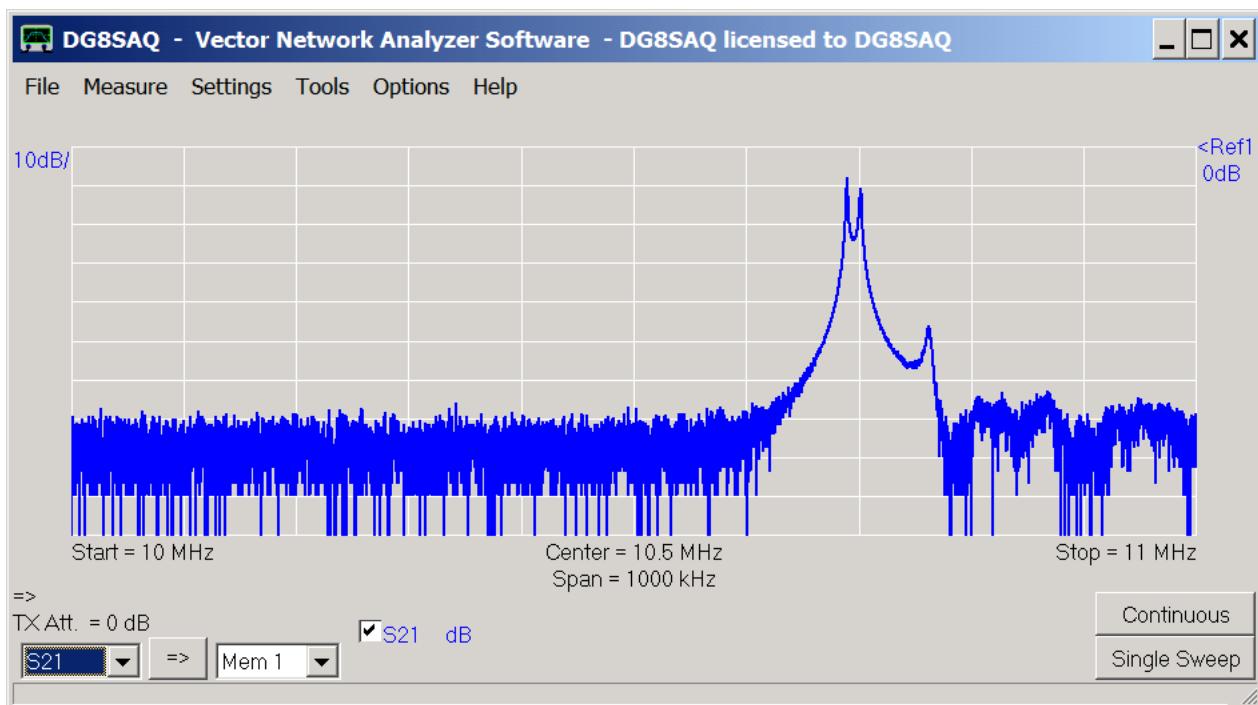
To make sure that we capture all interesting features, we will set a large number of data points (here 8000) and a the fastest sweep time (0.16ms per data point) by pressing the s-key or by selecting Settings-Sweep:



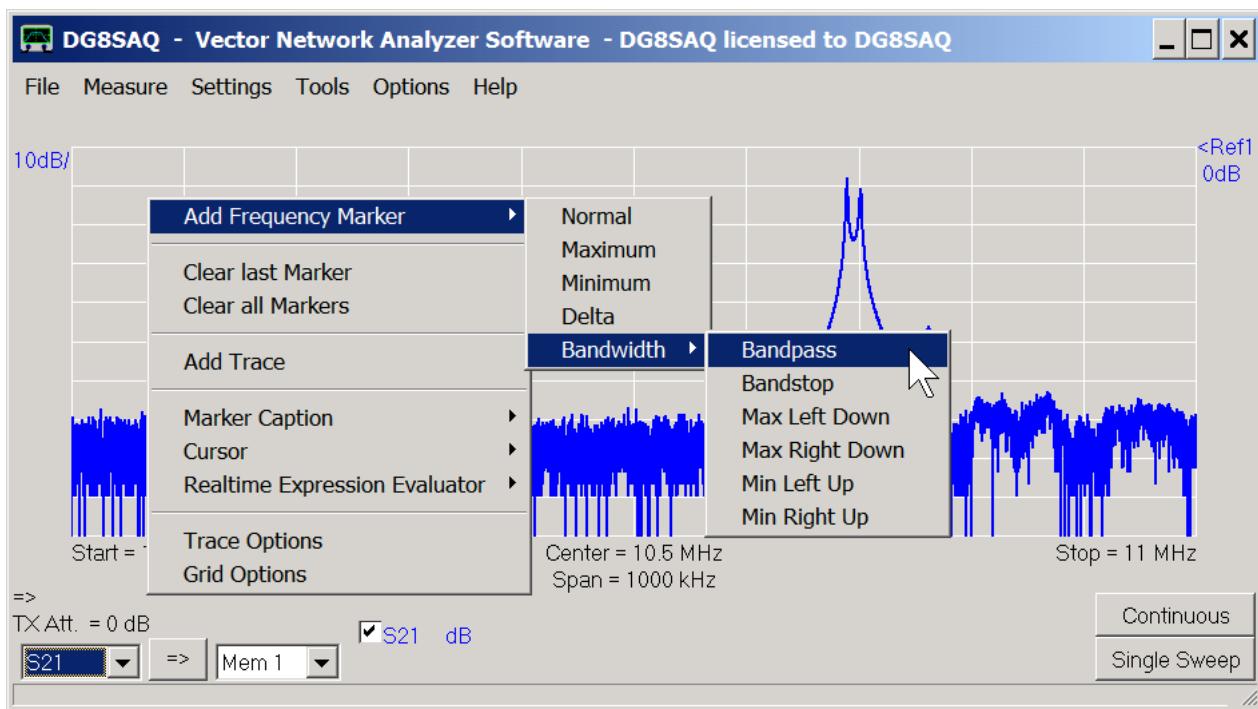
Next, we set the instrument to the desired frequency range e.g. by double-clicking the "Stop" label or by simply pressing the f-key on the keyboard.



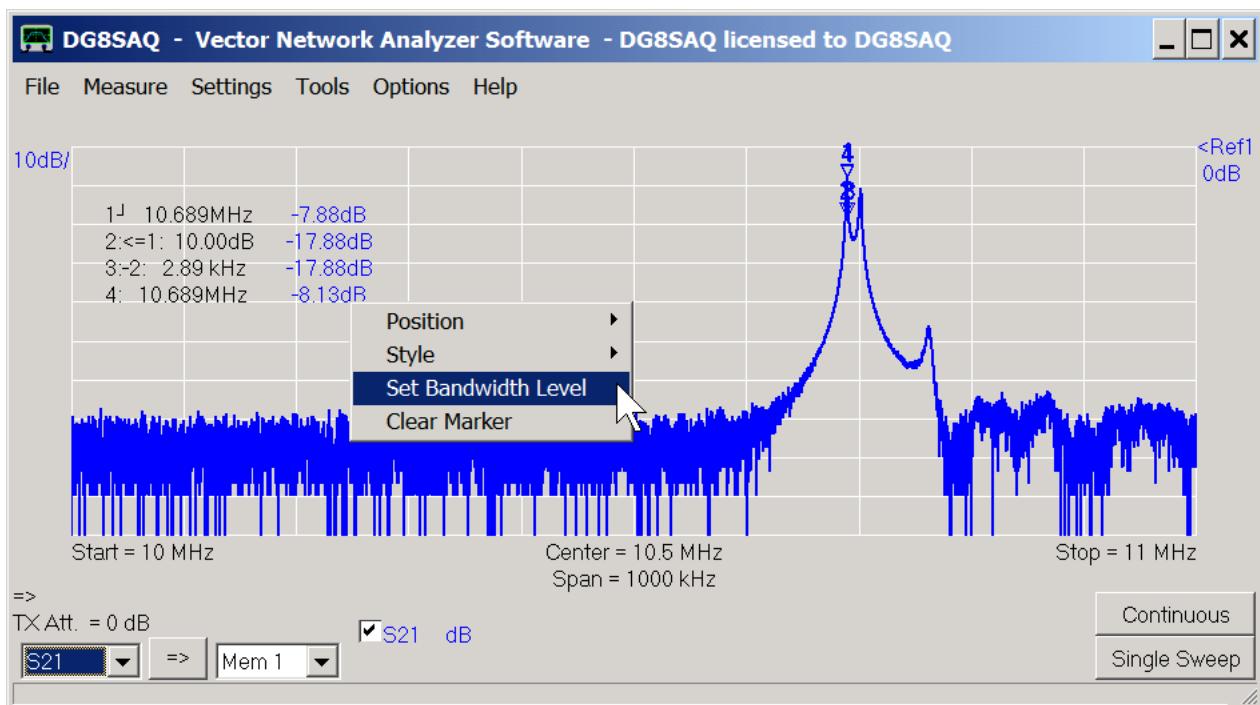
We select S21 (=transmission) for a trace as shown below and do a single sweep by pressing the "Single Sweep" button:



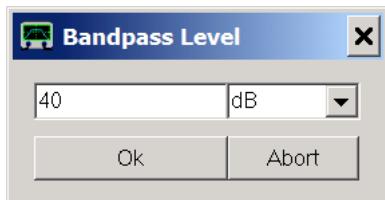
The span (1MHz) is much too high and the center frequency is off. We place a bandwidth marker by right-clicking onto the main grid in order to determine the optimum center frequency and span:



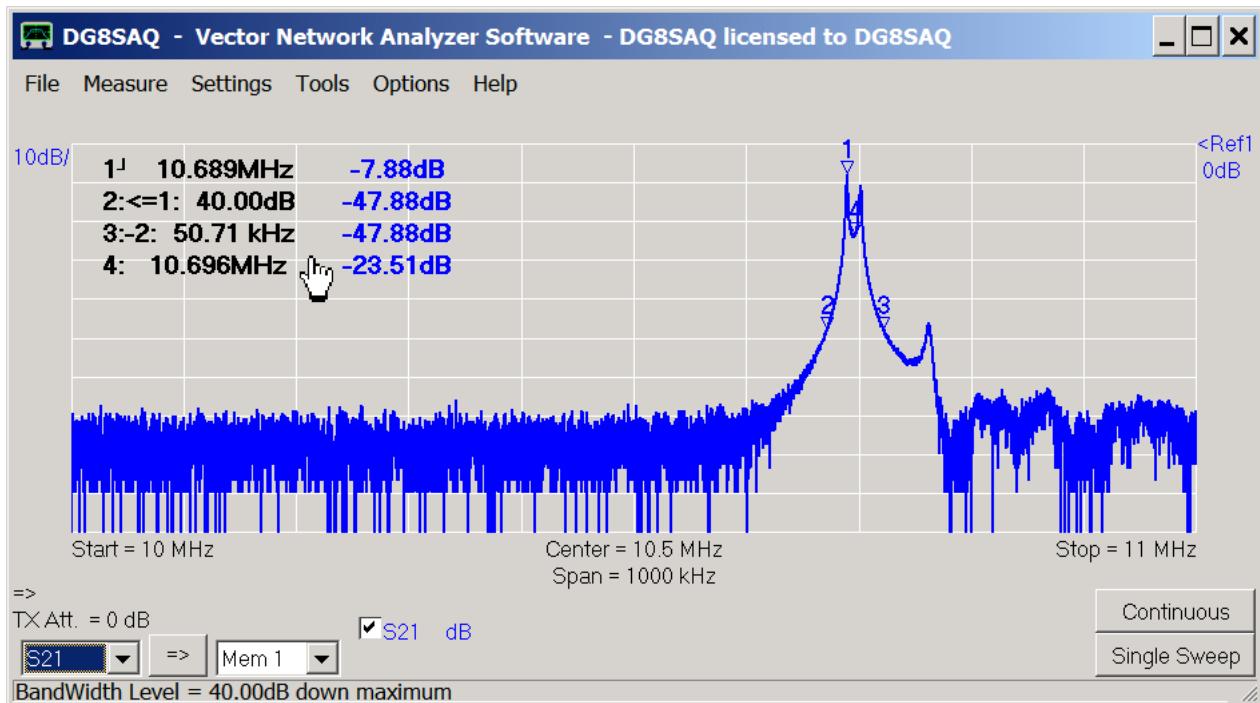
We want a higher bandwidth level, so we right-click on marker4 caption to change it...



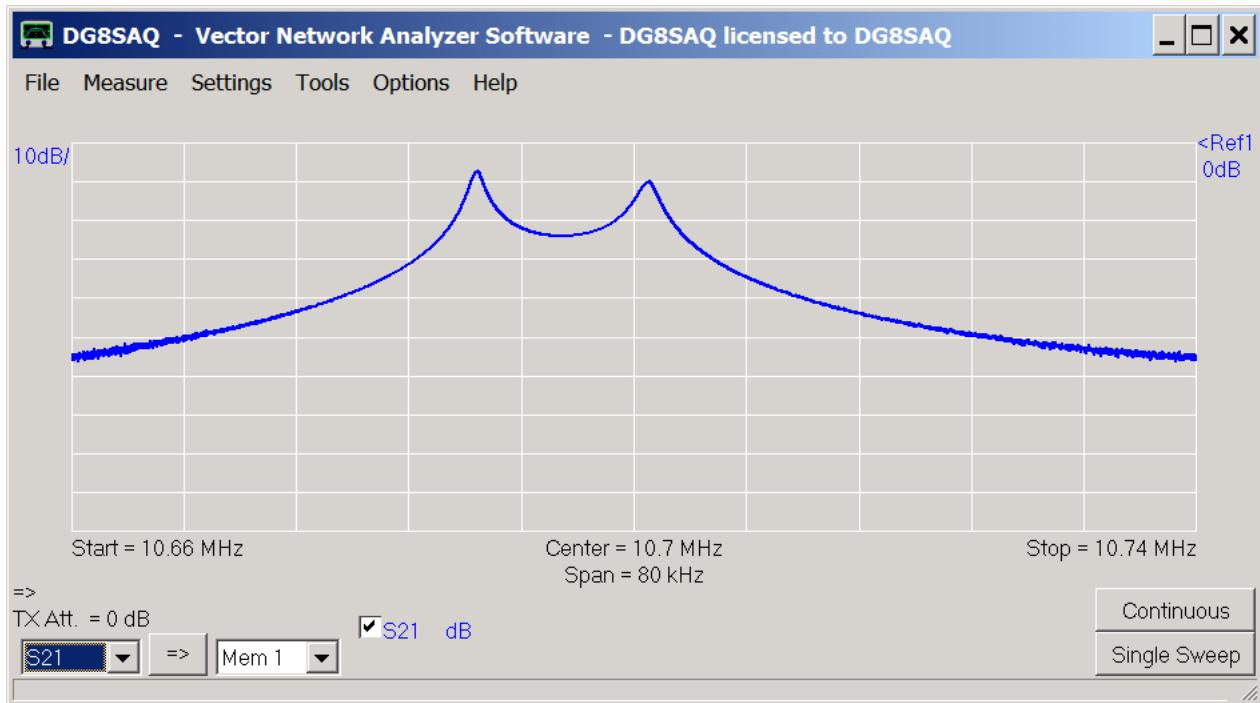
... to 40dB:



Now, we can read off the desired center frequency and span:

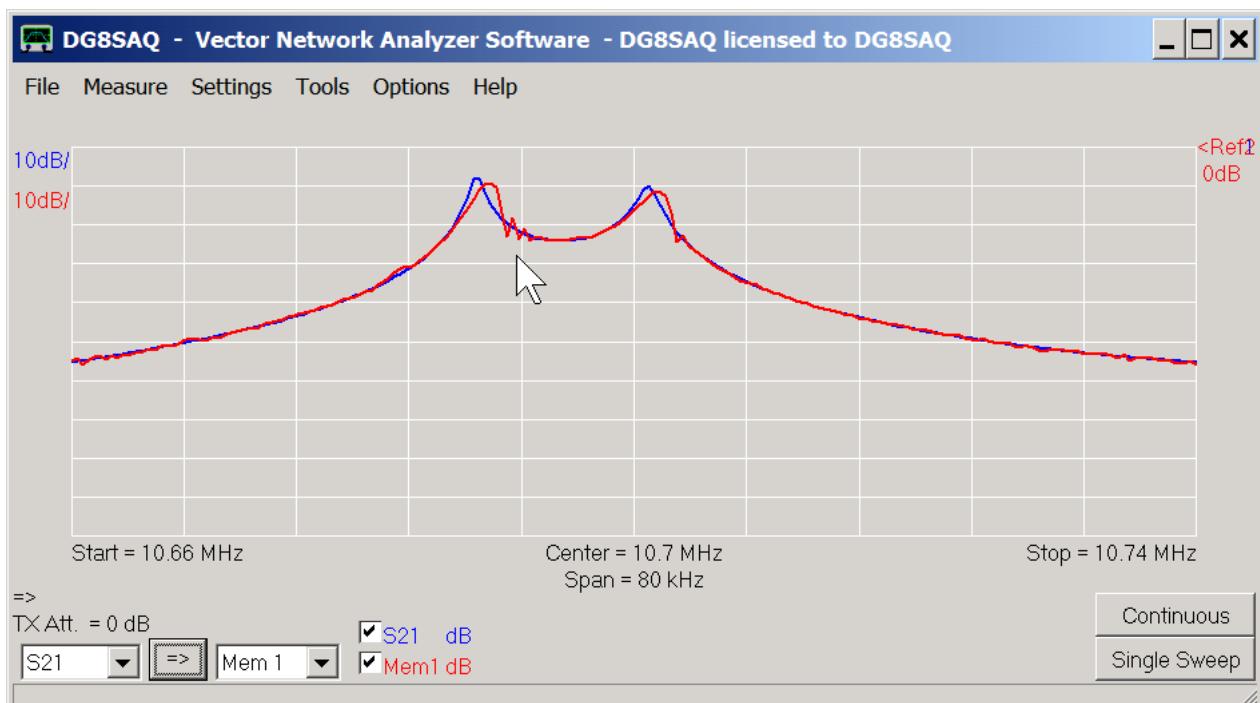


We decide to use a center frequency of 10.7MHz, a span of 80kHz:



This way, we have a good view of the passband of the filter (between the two maxima) and we also see some of the nearby stopband. Note, that the filter passband is not flat, because the filter is not impedance-matched to the VNWA impedance level of 50 Ohms.

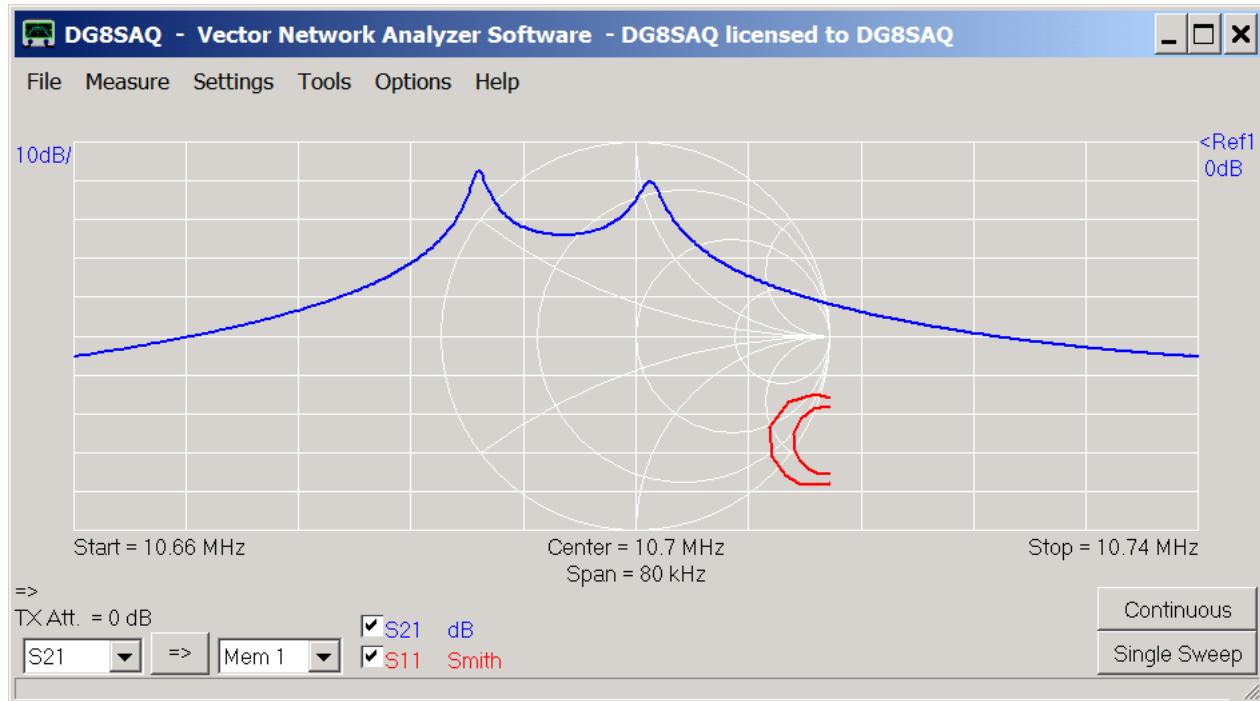
Now, we reduce the number of frequency points to a reasonably low number (500 is quite sufficient) and in turn increase the measurement time such that the whole sweep takes about 4 seconds. Note, that quartz filters have relatively high Q values. Therefore, they need time to settle to a new frequency stimulus. Thus, it is crucial to sweep them slowly. Play with sweeping speed and see how the passband changes. In below view (200 points only), the red trace is swept within 30ms, while the blue trace was taken within 4s:



Clearly, the red trace was taken at a too high sweep rate as the maxima are shifted to the right and the filter is ringing after the maxima (marker arrow).

Finally, we activate a second trace for S11 by pressing the + key on the keyboard, we select it to display a Smith chart

and we sweep again:

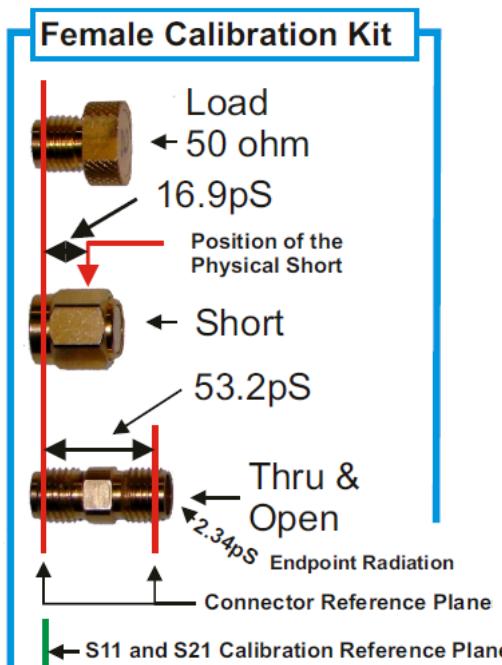


The just measured S11 reflection data is partly outside the Smith chart because our VNWA is not yet calibrated. The data is not valid yet.

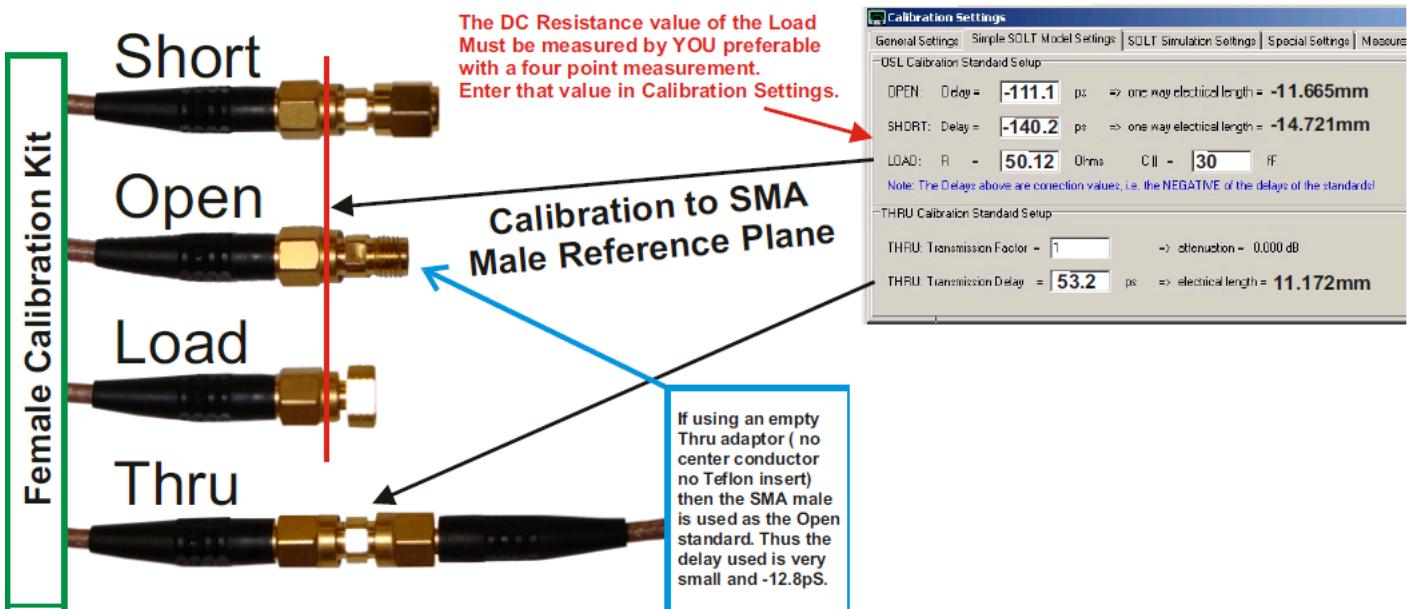
Now we are ready to calibrate the VNWA for a two port measurement, which consists of transmission measurements (S21,S12) and a reflection measurements (S11,S22).

2. Calibration

We want to perform a calibration using a female Amphenol calibration kit available through SDRKits:



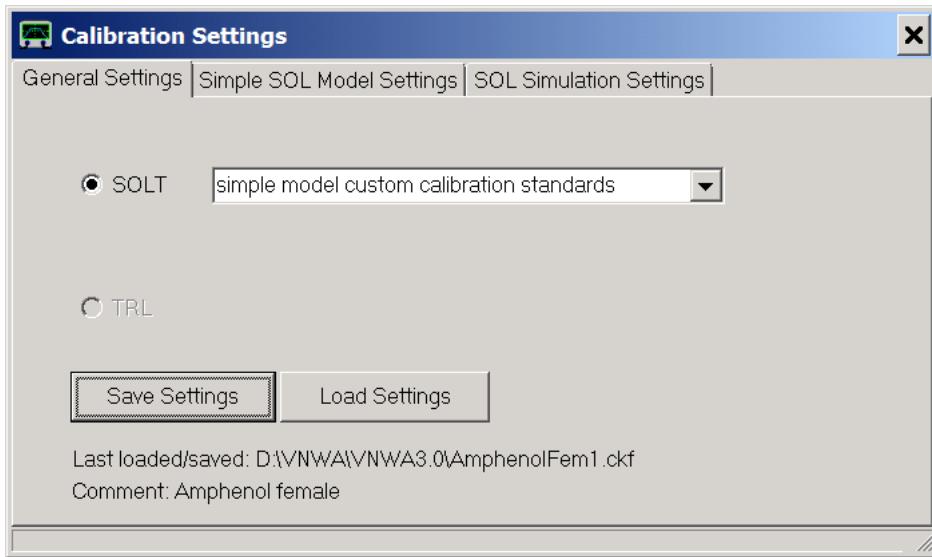
We want to perform a calibration which places the calibration plane exactly into the male test cable connector reference plane:



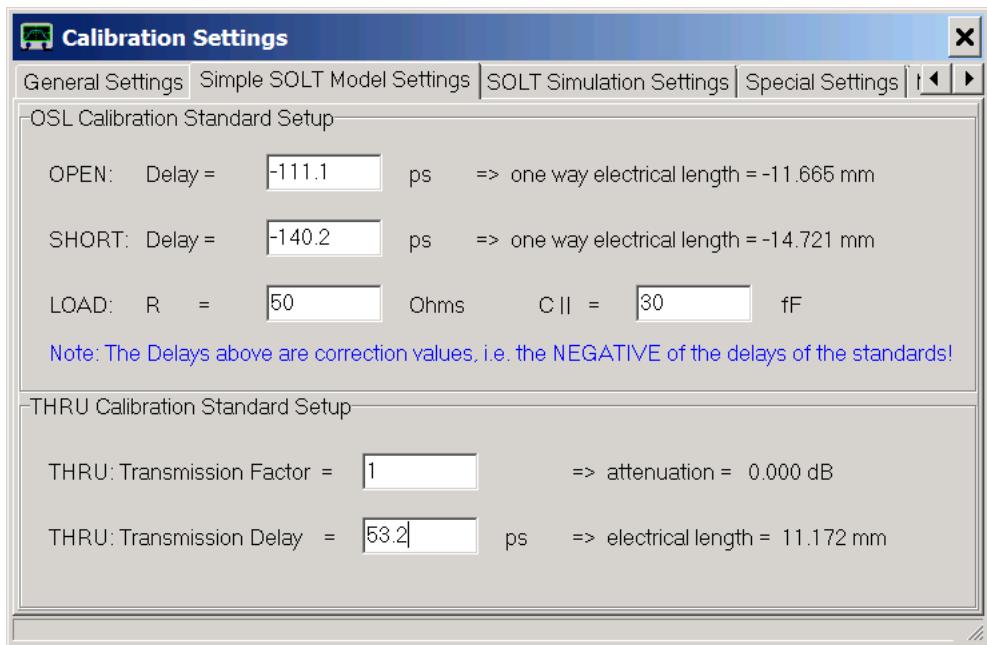
The above calibration kit data was compiled by Kurt Poulsen. Kurt updates this data for all kinds of calibration standards and models on his web page <http://www.hamcom.dk/VNWA/>.

Please check there to obtain the latest and most accurate data for your calibration standards.

We open the calibration kit settings window by pressing the k-key on the keyboard, select "simple model custom calibration standards"...

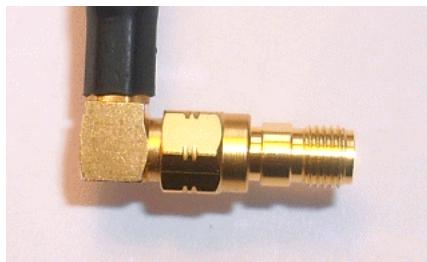


... and copy the calibration kit parameters from Kurt Poulsens document:

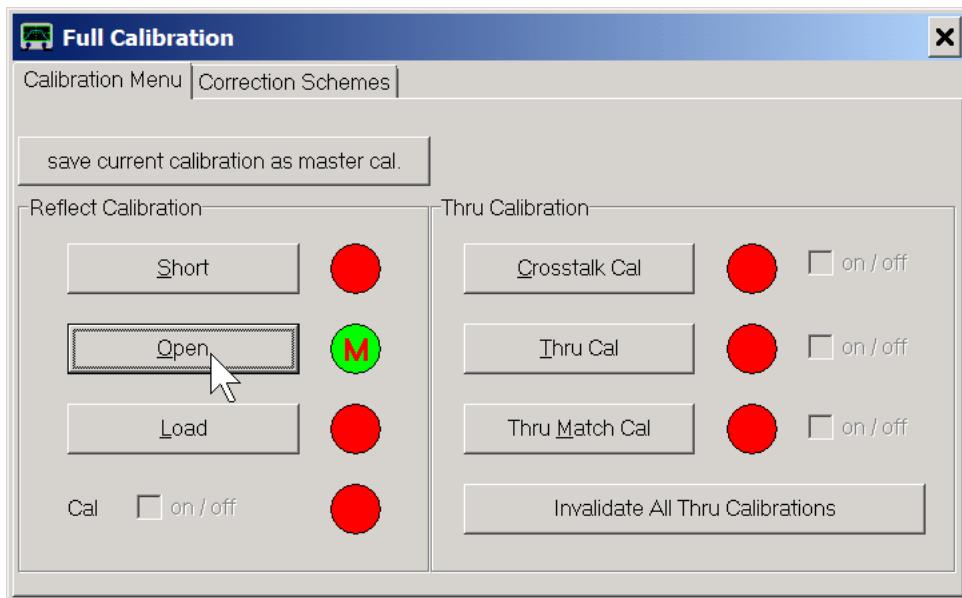


Note: Before performing any calibration measurement, make sure, that the instrument sweep rate is set to the lowest value you intent to use the calibration with, otherwise your later measurement results will be dominated by the noise of your calibration measurements. We have selected a total sweep time of 4s above which is as slow as we want to sweep, so we're ok.

First, we do a reflect calibration of the TX test cable end. Our test cables remain connected to the VNWA TX and RX ports. We connect our open calibration standard (in this case identical with the female-female through connector) to the far end of the TX port test cable:



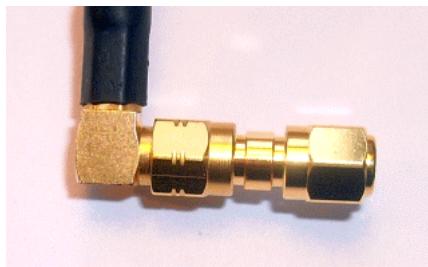
We select "Measure-Calibrate" or simply press the c-key on the keyboard, press the "**Open**" button, confirm that the open standard is connected and observe the instrument measuring the standard.
When the measurement is completed, the red Open light turns green:



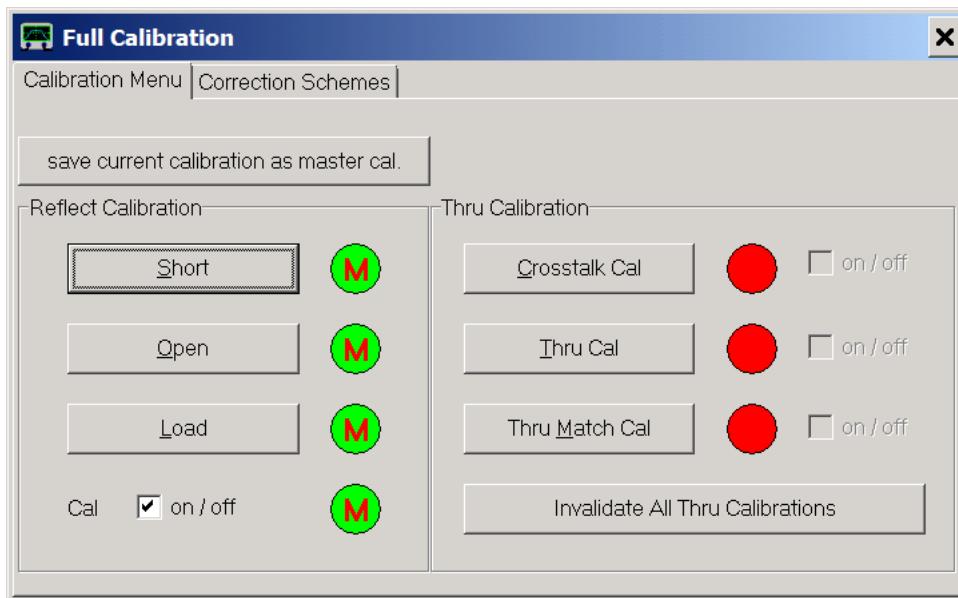
In similar fashion we measure the load calibration standard...



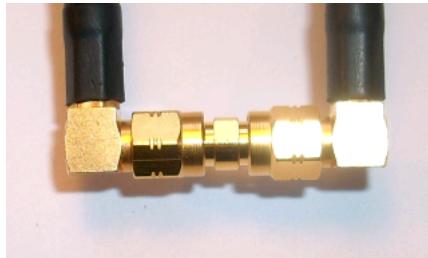
... and the short calibration standard:



Finally, all reflect calibration lamps should be green:



Next, we need to calibrate the VNWA for transmission. We therefore connect the test cable ends of TX port and RX port with the Thru standard...

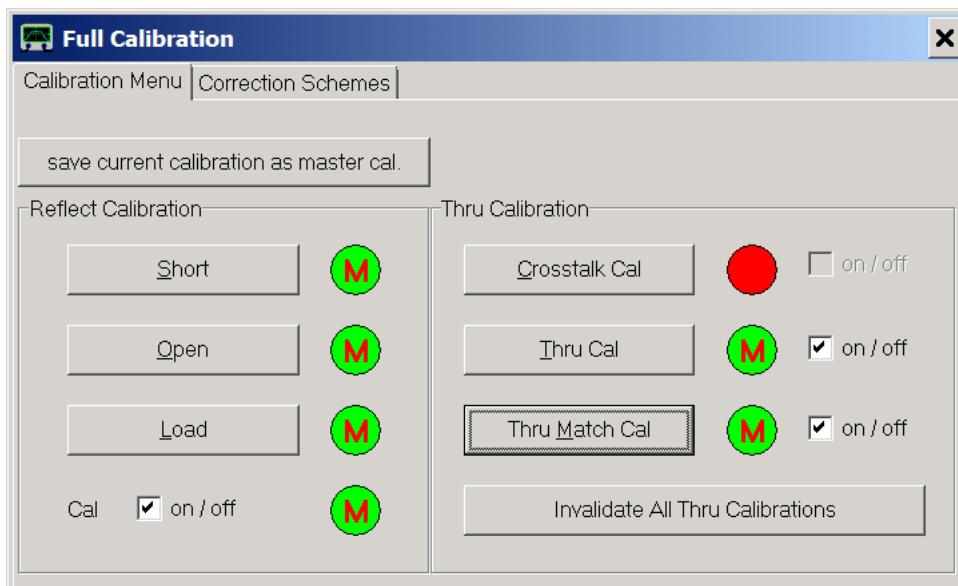


... and press the "**Thru**" button and let the instrument do the calibration sweep.

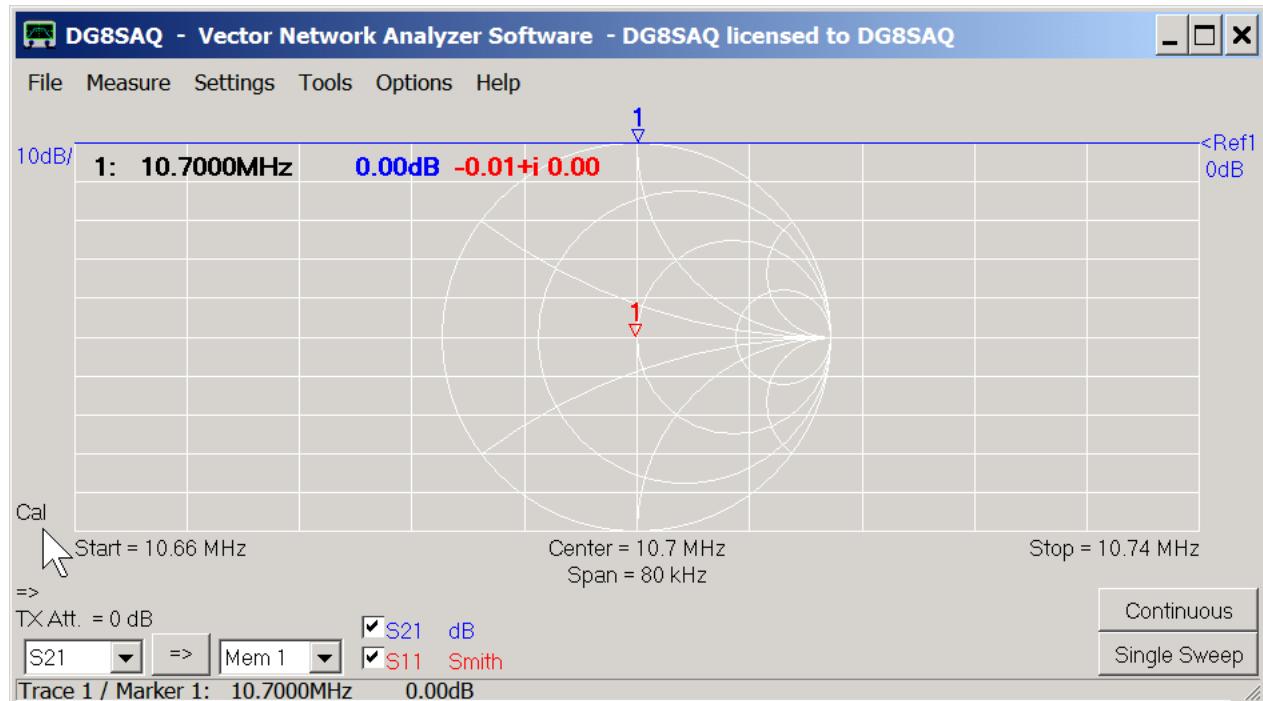
The "**Thru-Match**" calibration step requires the same thru standard as the thru calibration. While the thru calibration step measures transmission, the thru match calibration step measures reflection of the RX port in order to numerically compensate deviations from a perfect 50 Ohms RX port match.

So, once the thru calibration sweep is done, we press the Thru Match button.

We do omit the crosstalk isolation as this is usually not required. Finally, all lamps except for the crosstalk calibration are green:

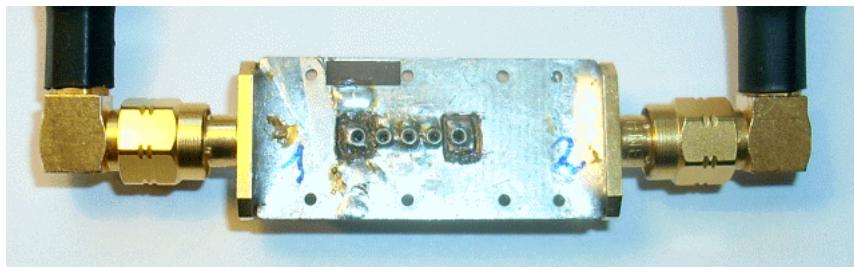


We close the calibration window and double-check the calibration by measuring the "Thru" calibration standard.

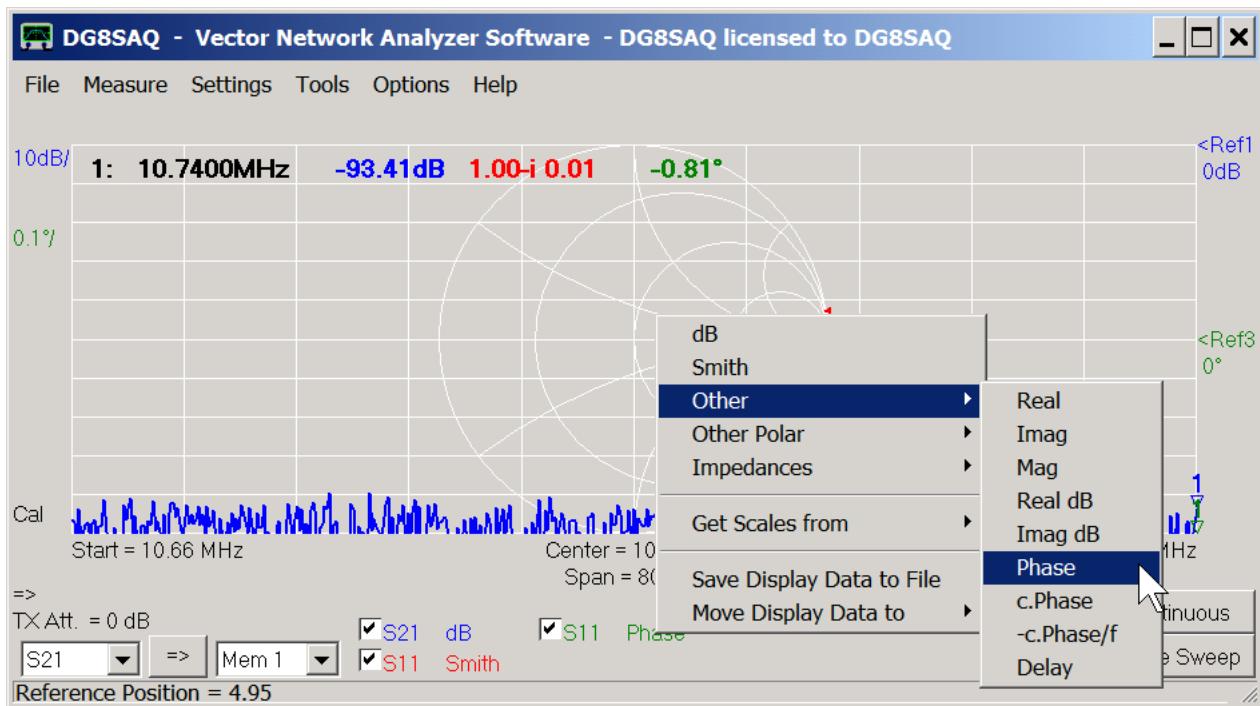


As our Thru standard has by definition 0dB attenuation, we obtain a flat blue 0dB line. We have also measured the reflection coefficient of the RX port (red trace reduced to a dot, barely visible without marker), which is very close to zero for all frequencies.

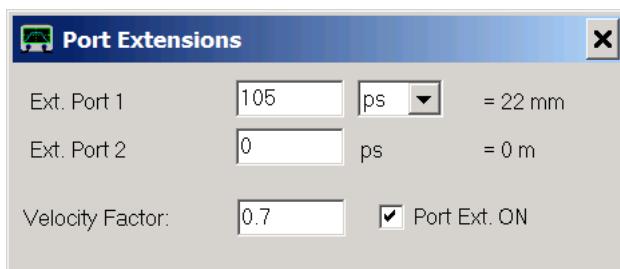
Next, we must move the calibration plane from the male reference plane position into our device test board to the wire position of our test object using port extensions. We therefore insert the test board **without** the test object with port 1 connected to the VNWA TX port ...



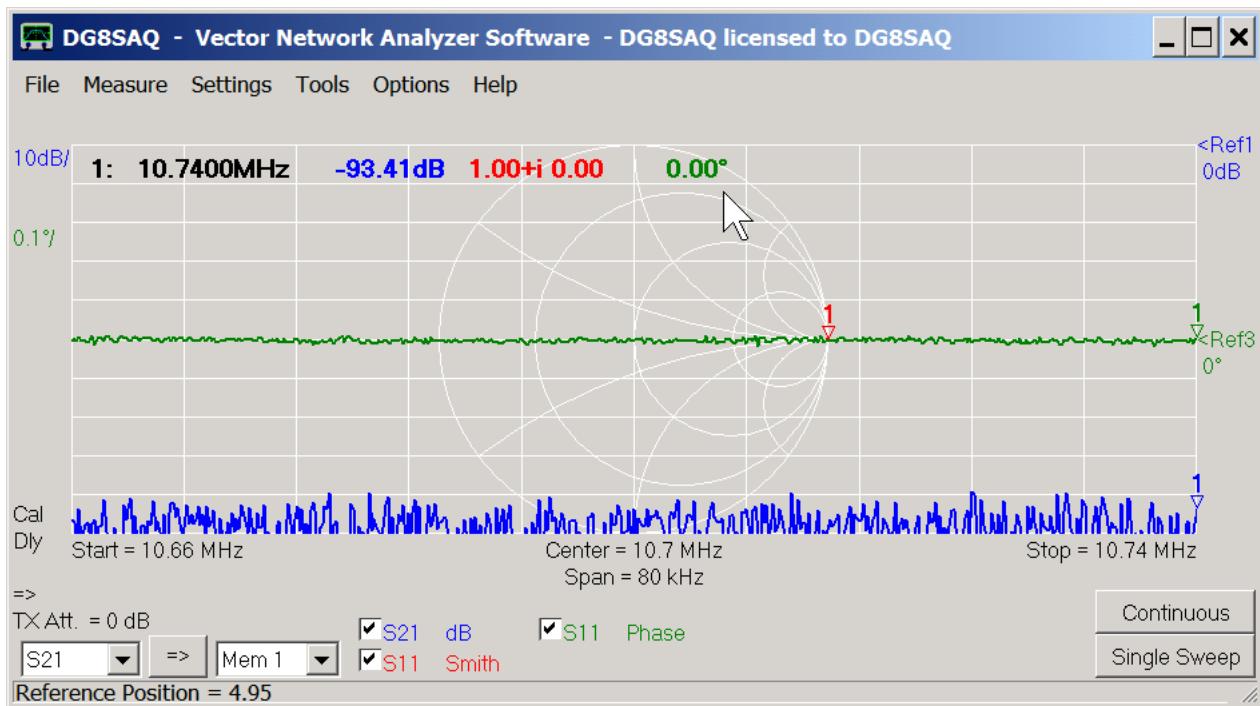
... and do a sweep. We add another trace showing S11 by pressing the + key and select it to show the phase:



Then we open the **port extensions** menu by pressing the p-key on the keyboard and place the port extensions window next to the main window so both remain visible:

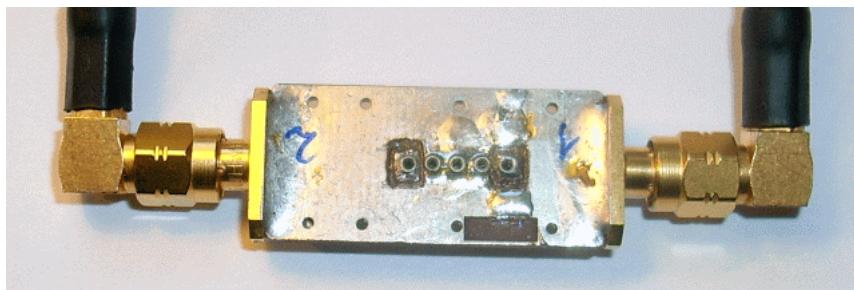


We check the **Ext. on** box (mousepointer) and set focus to the Ext. Port 1 by clicking on it. Next, while observing the green S11 phase trace in the main window, we turn the mouse wheel or edit the field until the the S11 phase is zero:

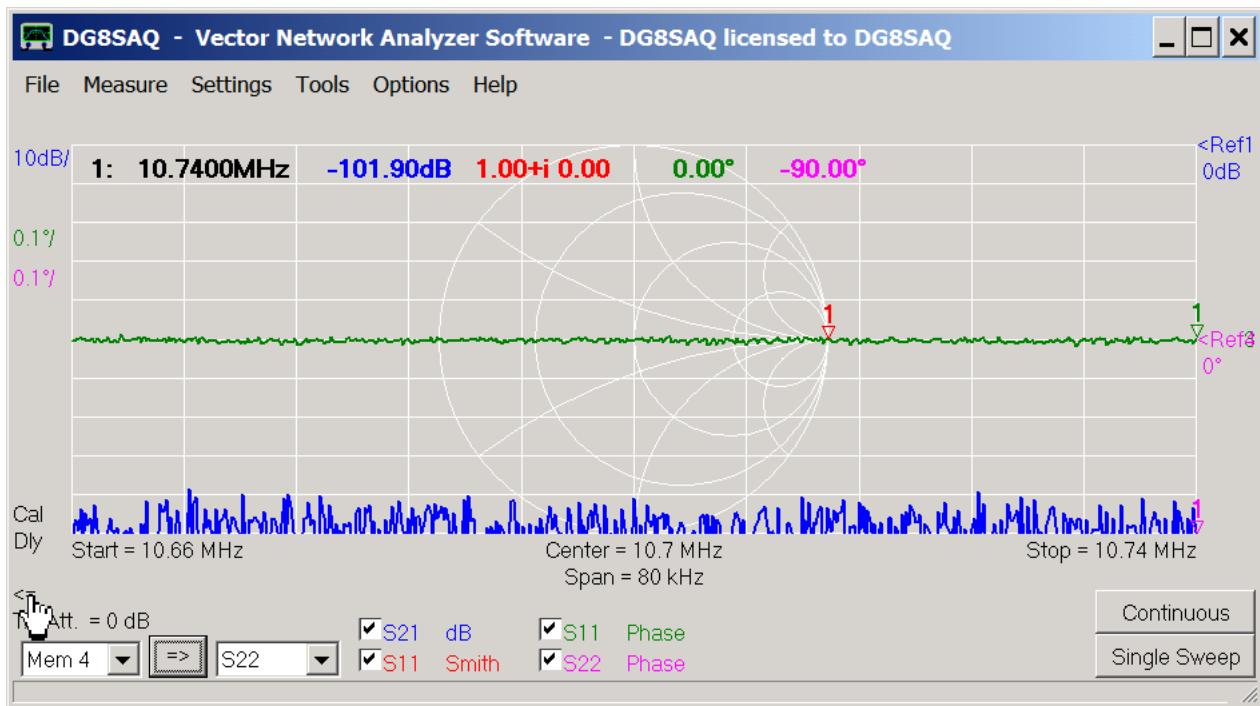


Note, that the reflection coefficient S11 is now exactly $1+i0$, which is the open reflection condition, corresponding to a dot at the right edge of the Smith chart.

Next, we reverse the empty test board, such that port 2 is connected to the VNWA TX port:

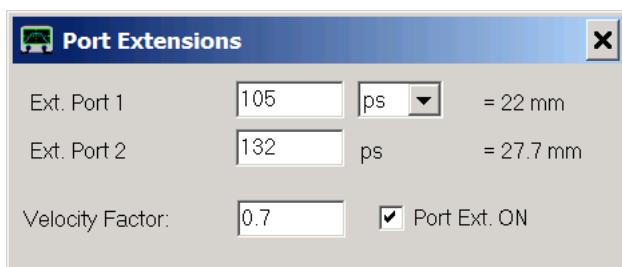


We add a fourth trace S22 also showing the phase and we reverse the measurement direction by clicking onto the direction arrow (hand mouse pointer in below figure):

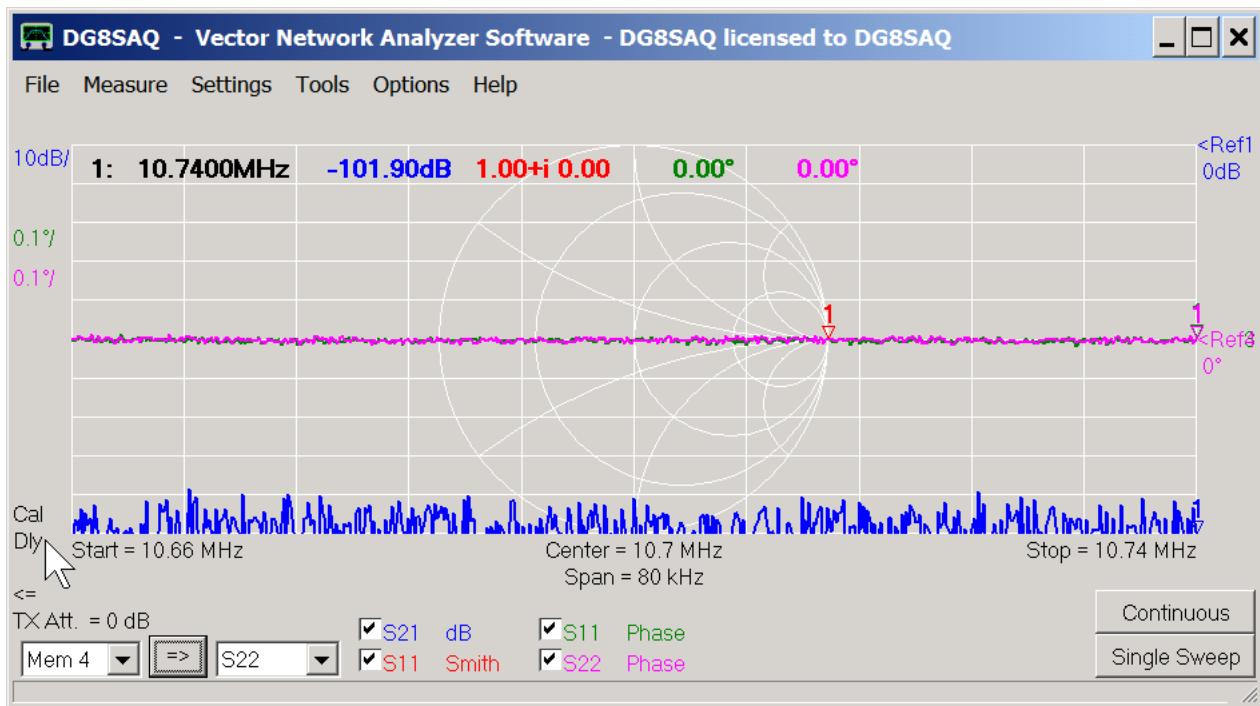


Note, that reversing the VNWA measurement direction arrow does not physically change the measurement direction. The signal still comes out of the TX port. That's why we have physically reversed the test object. Reversing the direction inside the VNWA software causes the reflection data to be collected in the S22 memory space rather than in S11 as before.

Now, do a single sweep and observe data being collected in S22. S21 and S11 data remain untouched by this sweep. Go to the still open port extensions window (or reopen it by pressing the p-key) and change the **Ext. Port 2** value while observing the pink S22 phase trace ...



... until the phase is exactly zero:



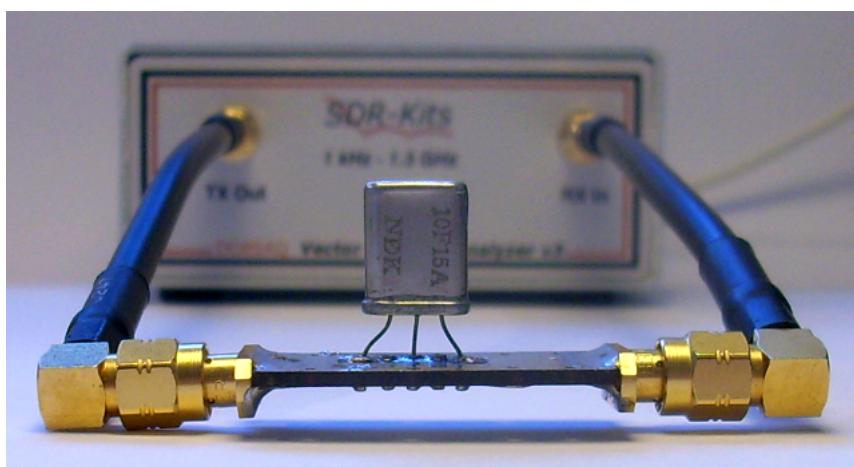
Note the labels Cal and Dly in above figure at the mouse arrow indicating that a calibration is active and port extensions (=port delays) are activated.

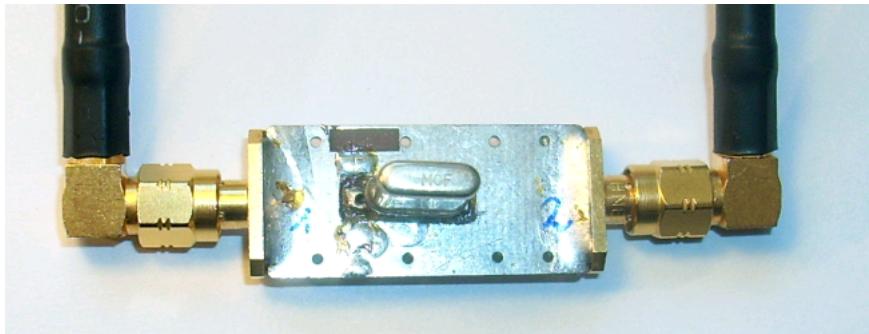
Looking at our port extensions, we observe, that the path from the test board connector to our test object on port 2 side is somewhat longer than on port 1 side, which is confirmed by visual inspection of the test board.

Now, the VNWA is fully calibrated and ready for measuring two port s-parameters and we can close the port extensions window.

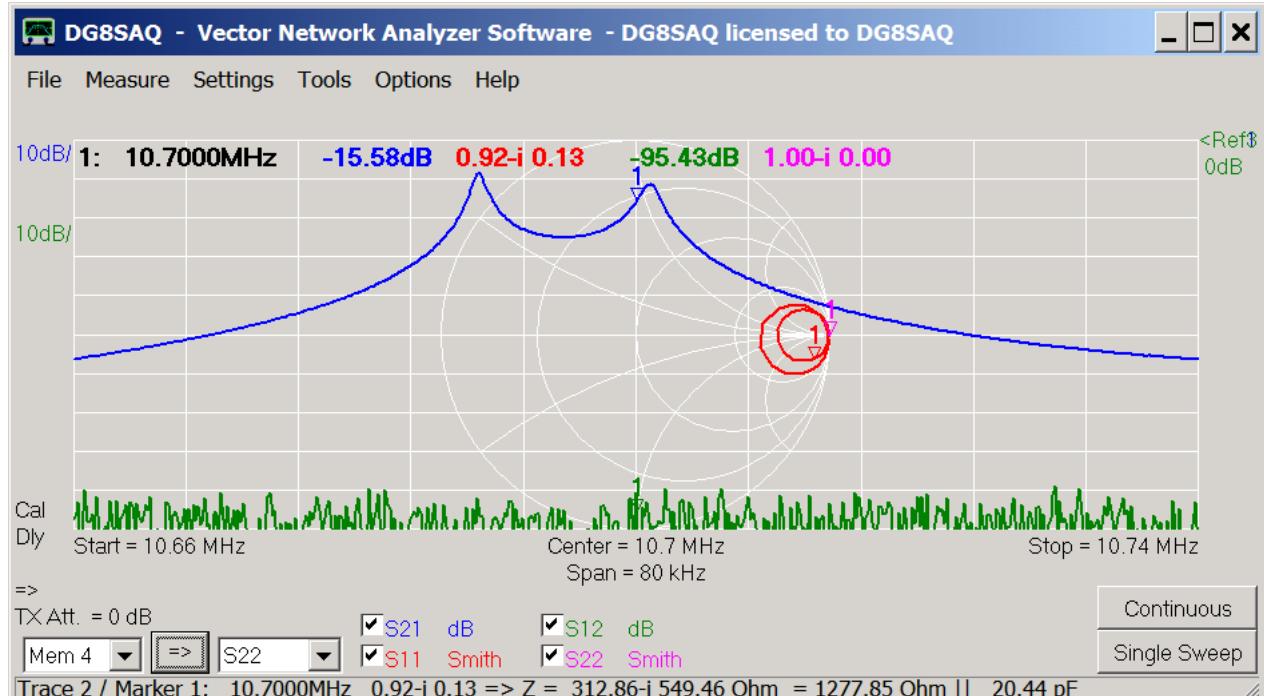
3. Measurement

Next, we put our test object (=crystal filter) onto the test board and connect the test board to the VNWA such that port 1 is connected to the VNWA TX port and port 2 is connected to the RX port, ...



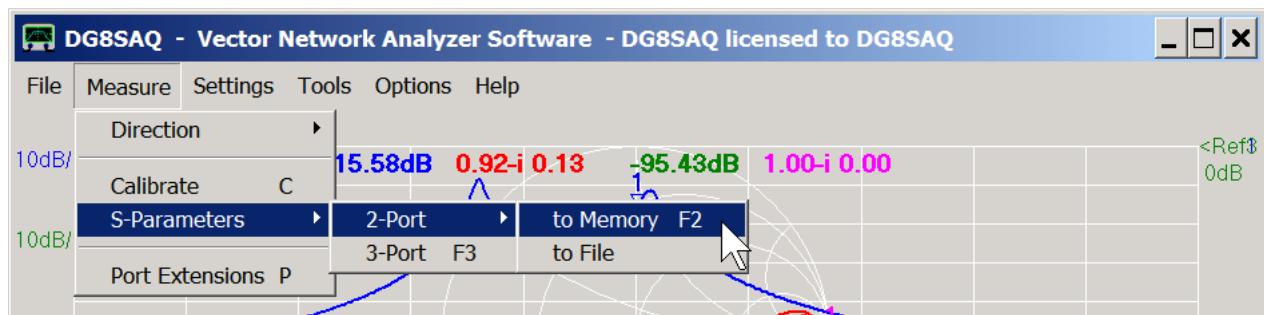


... switch the software to measure forward direction by clicking the arrow do a single sweep:

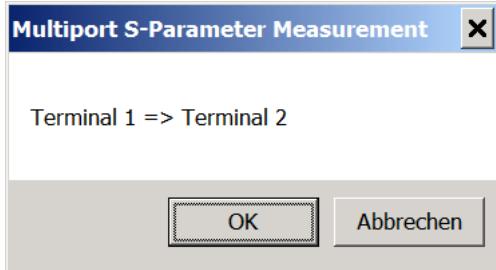


Now, we have already measured S21 and S11. Note, that the S11 data now lies completely inside the Smith chart as must be for a passive device. When clicking onto the red Smith chart marker 1, we get additional information on the S11 data at the marker position in the bottom status line. We do not only see the measured reflection coefficient ($0.92-i \cdot 10.13$), but also what this means in terms of impedance in the series equivalent circuit model ($Z=312.86$ Ohms $\cdot i^549.46$ Ohms) as well as in the parallel equivalent circuit model ($Z=1277.85$ Ohms $\parallel 20.44$ pF). So, the input impedance of our filter is quite high.

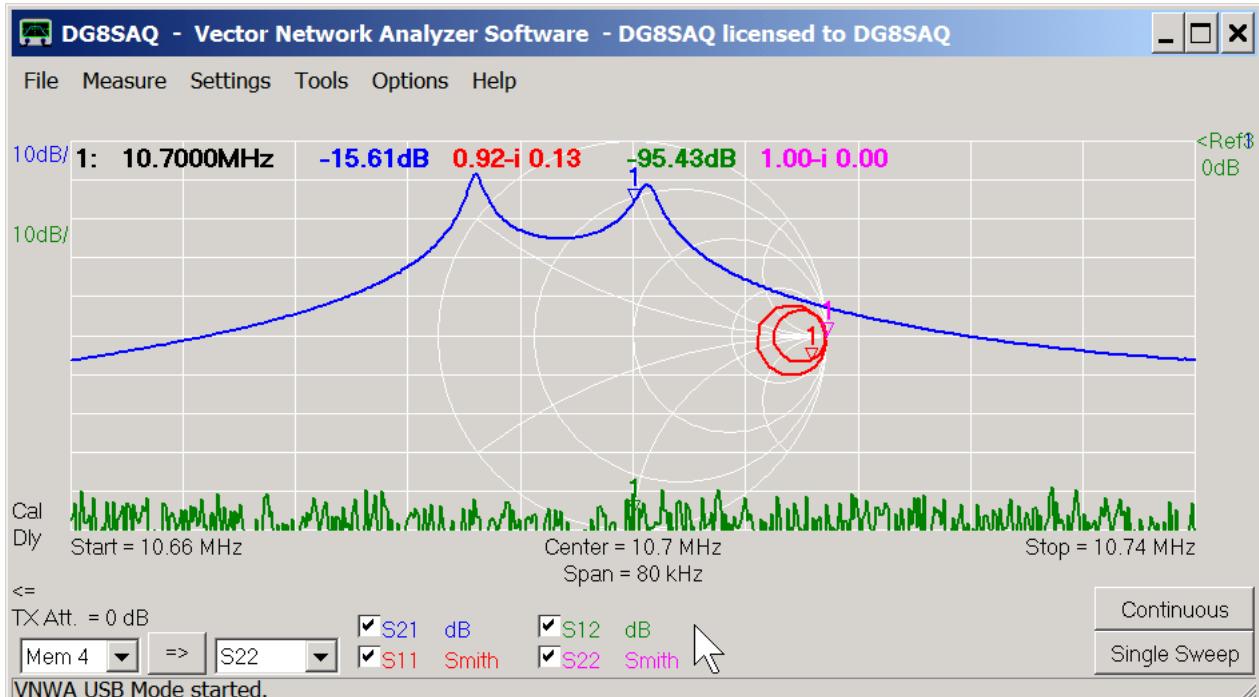
We could now collect reverse data by reversing the test object and manually reversing the measurement direction. But there is a more convenient way to collect 2 port data, which is the 2 port measurement assistant. We launch a 2 port measurement into memory by pressing the F2-key or via the main menu:



The assistant instructs us that the signal flow is Terminal1 => Terminal2, which means that the TX port should be connected to port 1 and the RX port to port 2.

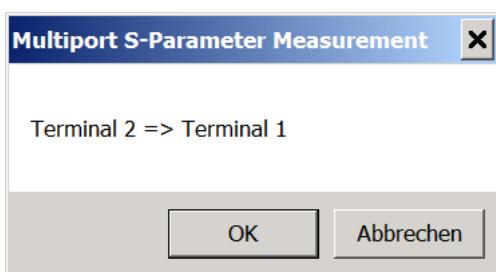


After inspection, that this is indeed the case, we press **OK** and observe how S11 and S21 are being remeasured:

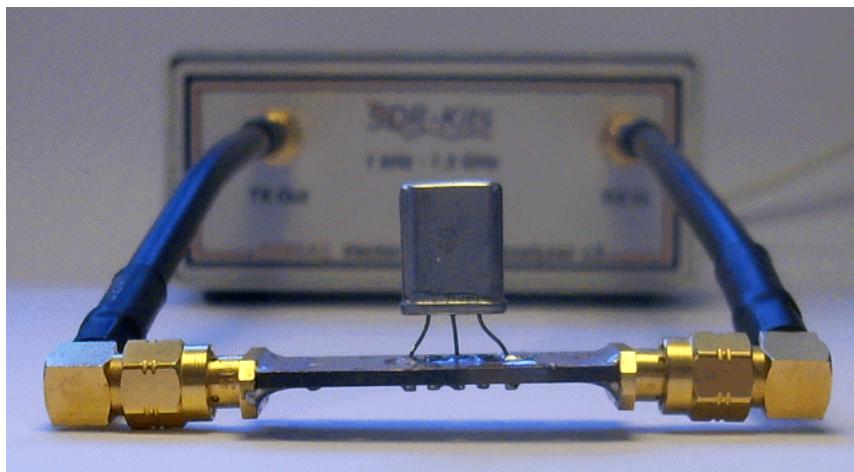


Note, that meanwhile we have chosen to display all four S-parameters S11, S21, S12 and S22.

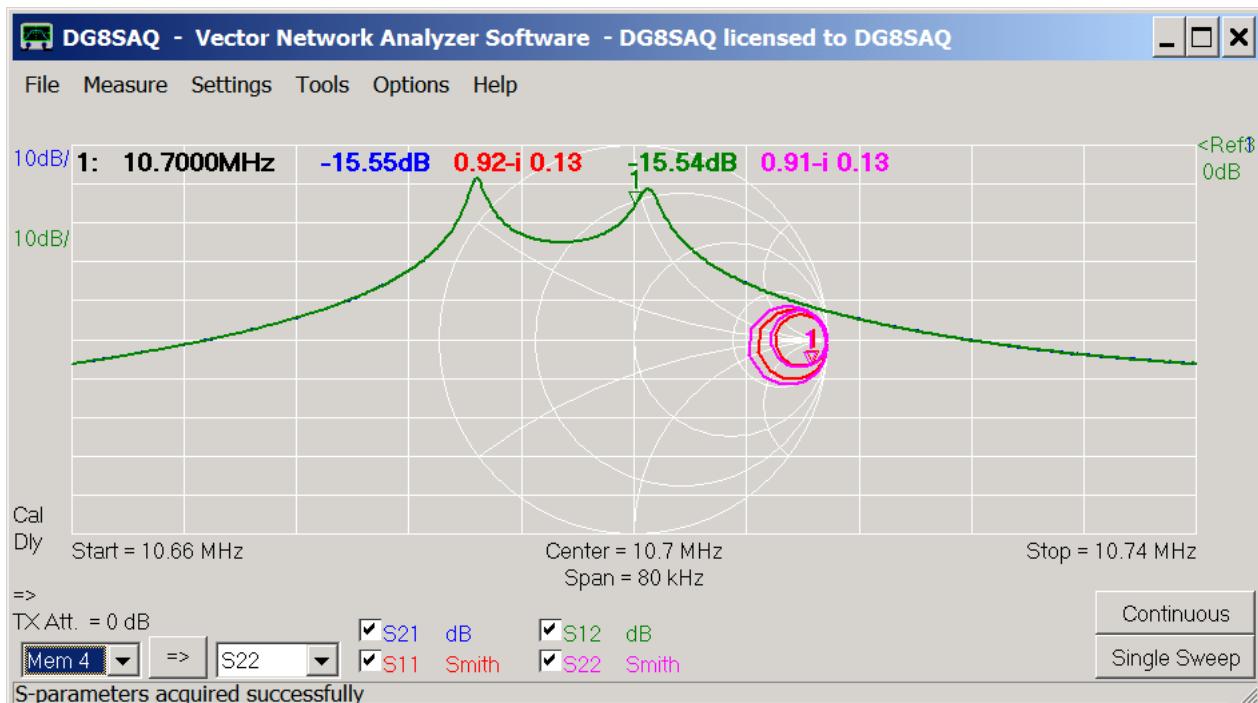
When the forward sweeps are done, the assistant pops up again indicating that next the reverse direction is to be measured:



This means, that we have to reverse our test board, i.e. connect port 2 to the VNWA TX port and port 1 to the VNWA RX port:



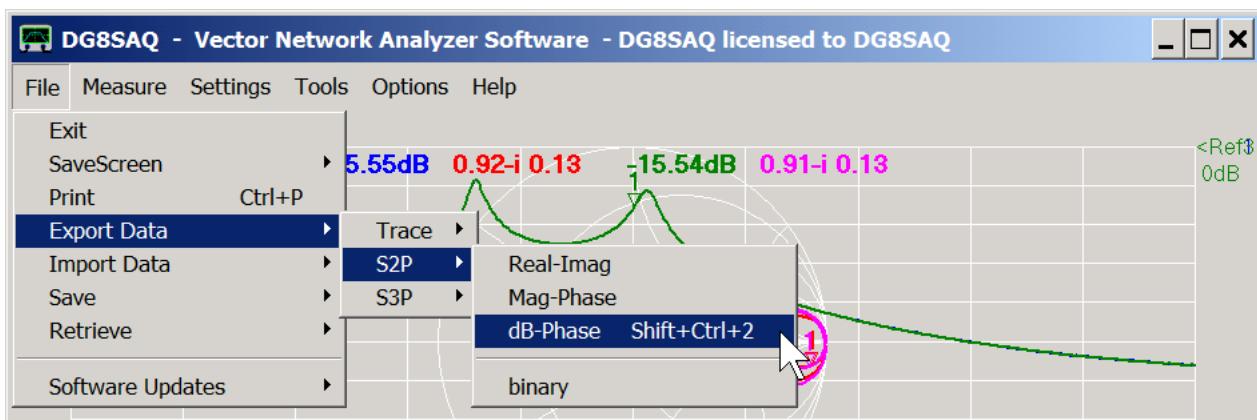
Once we are done with reconnecting the test board, we press **OK** on the multiport assistant and observe S12 and S22 being measured:



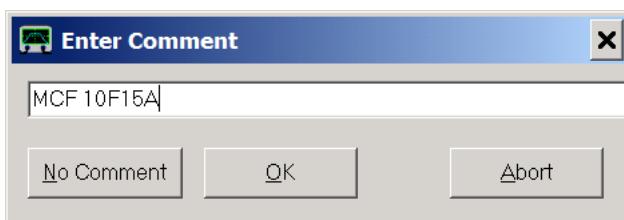
Note, that the forward S-parameters (S21, S11) might slightly change during the reverse measurement as the software uses the additional information of the reverse measurement to correct the forward measurement results in order to obtain the highest possible accuracy.

Note, that S12 and S21 are identical as must be the case for reciprocal devices like passive filters. S11 and S22 are quite similar indicating that the filter input and output are designed in similar fashion.

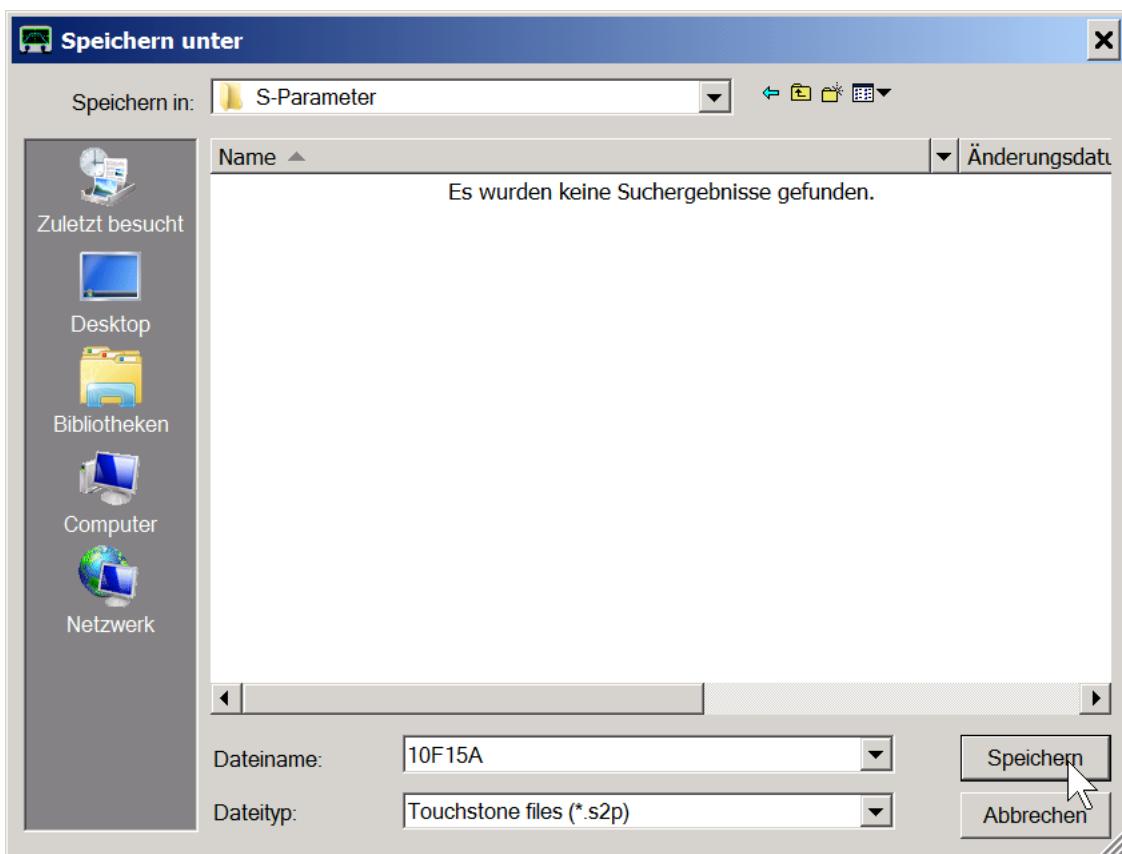
Finally, we save the full set of measured S-parameters into a Touchstone s2p-file for later use:



We may add a comment to the s2p-file...



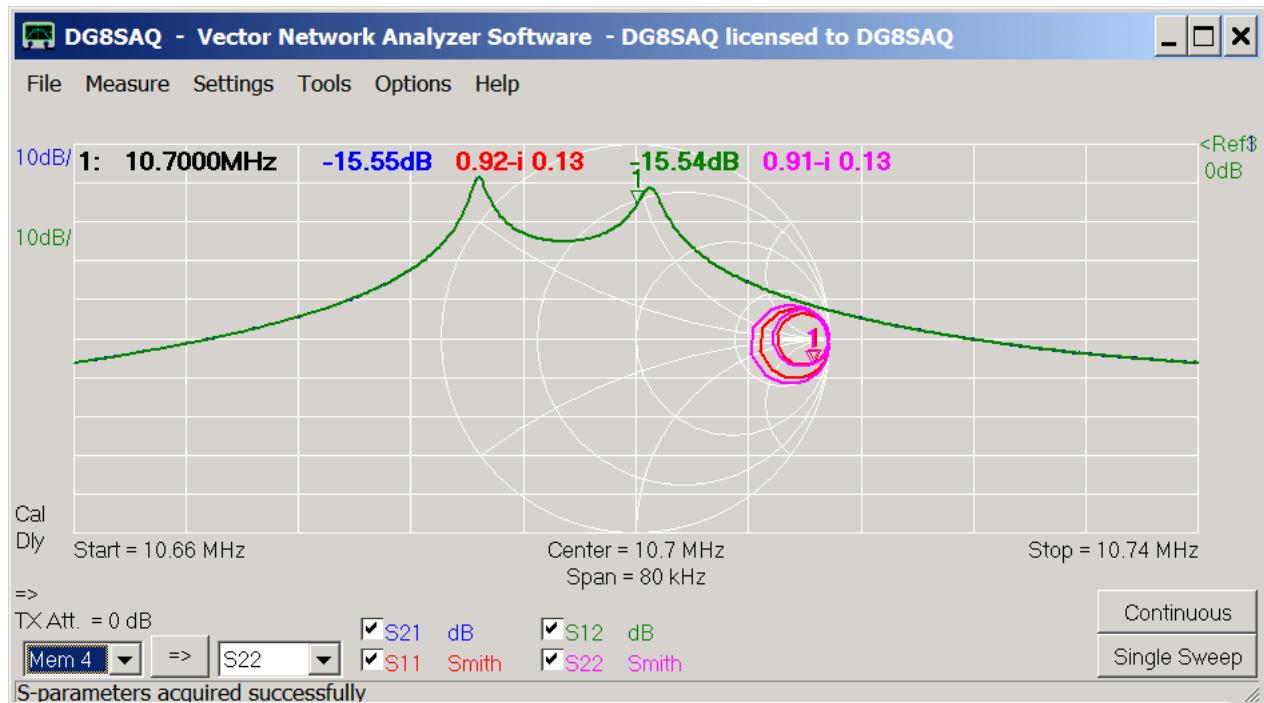
... and save it:



Now, that we have obtained the full set of S-parameters, what can we do with them?
This will be demonstrated on the following page "Interpretation and Usage of 2-Port Measurements".

INTERPRETATION AND USAGE OF 2-PORT MEASUREMENTS

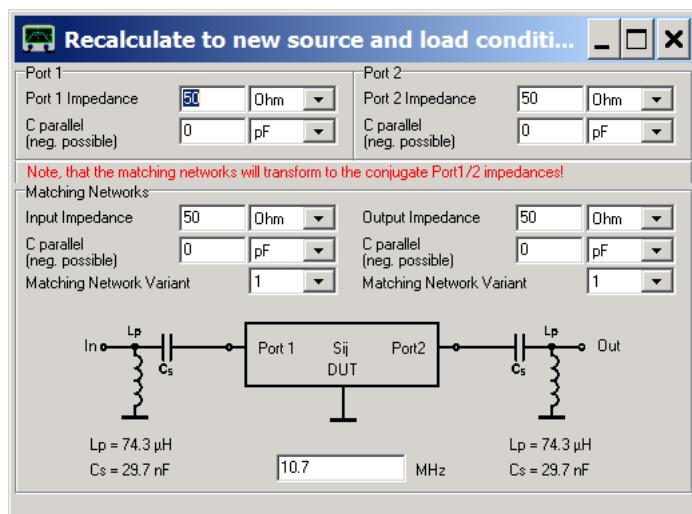
In the previous page "Practical Example on Performing a 2-Port Measurement" we have demonstrated how to measure the two port S-parameters of a two pole crystal filter:



Now, what can we learn from these and how can we make use of them? We can do some simple analysis right inside the VNWA software. For more sofisticated simulations we need to make use of network simulation software, which will be described at the end of this page.

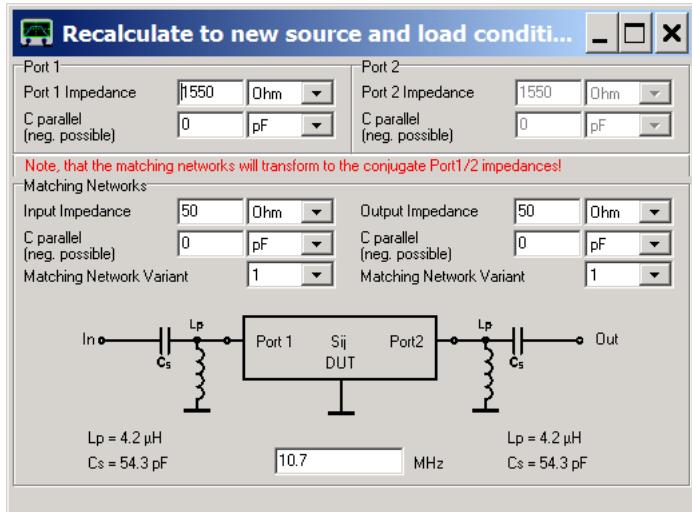
Analyzing S-Parameters inside the VNWA Software

Note, that both S11 and S22 data lies at the open circuit side in the Smith chart, i.e. our filter appears to have a relatively high input and output impedance compared to 50 Ohms. We have measured the transmission in the 50 Ohms impedance environment of the VNWA, though. Now, it could be interesting to find out, how the same filter would behave in a high impedance environment, i.e. in an environment where both the filter input and output are power matched. We actually can do this analysis inside the VNWA software using the Matching Tool, which we activate by pressing the m-key on the keyboard:

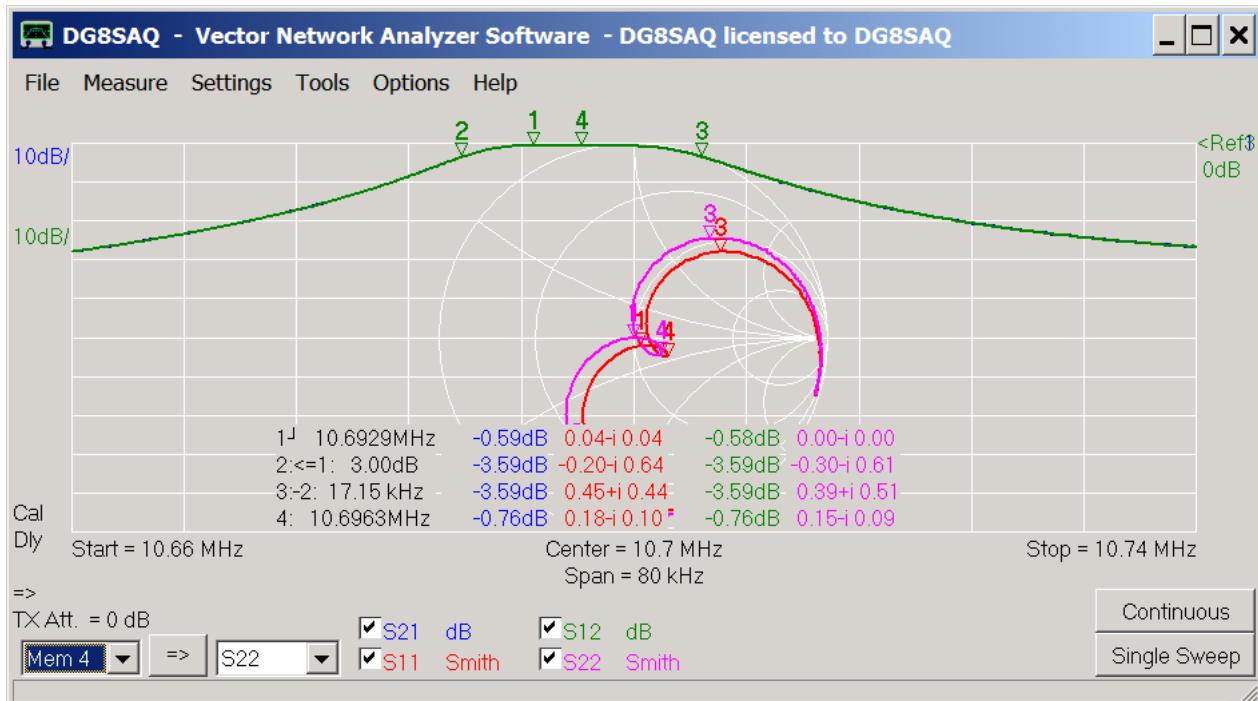


We can now modify port 1 and port 2 impedances by using the keyboard or the mouse wheel (right-click onto the edit fields to change the mouse wheel increments) and observe in the main window how the S-parameters change when transformed to different terminating conditions, e.g. changing the terminating impedances on both sides to 1550 Ohms

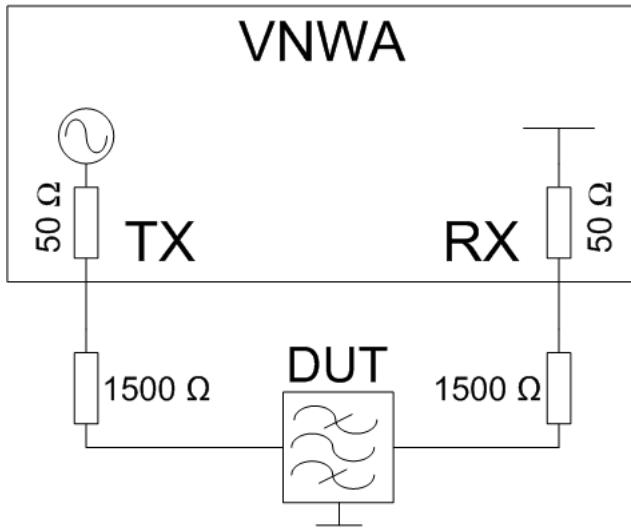
...



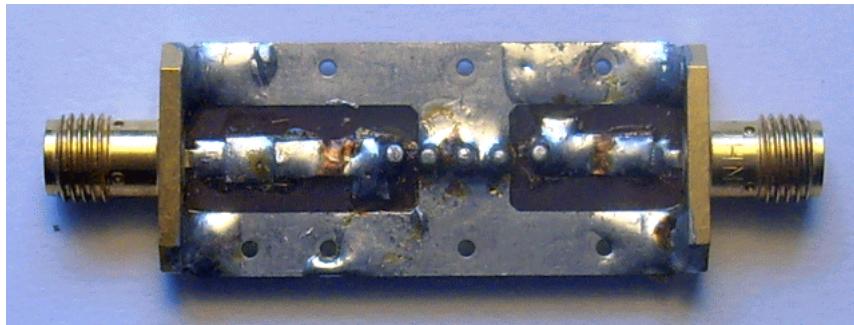
... will yield a reasonable power match condition and a flat passband:



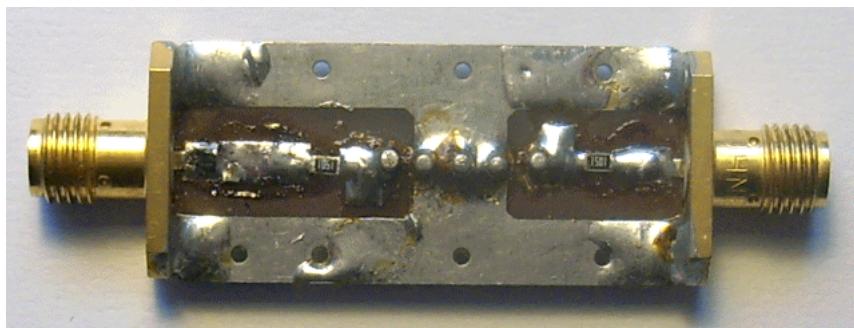
We can easily implement this impedance condition in an experiment by connecting 1500 Ohms resistors in series with the VNWA RX and TX ports. The RX and TX port impedances of 50 Ohms will add to these 1500 Ohms to yield 1550 Ohms impedance at the filter:



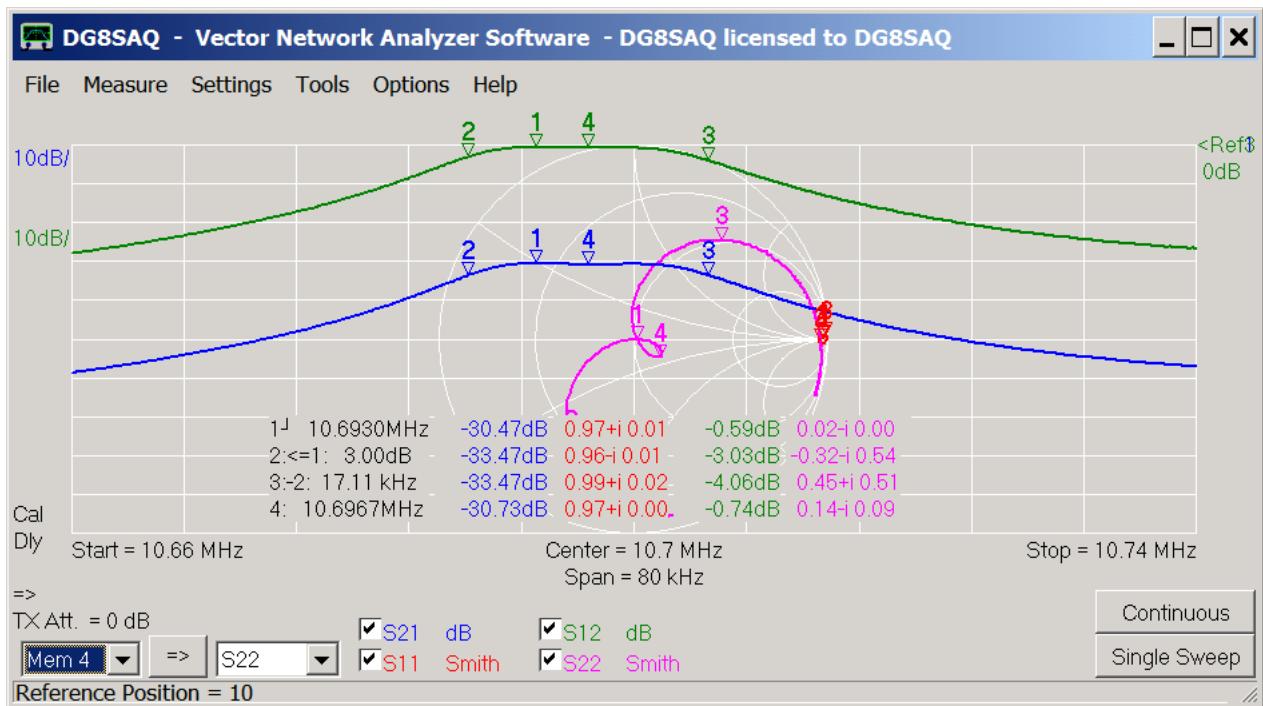
We therefore modify the bottom of our test board ...



... to include two 1500 Ohms resistors:

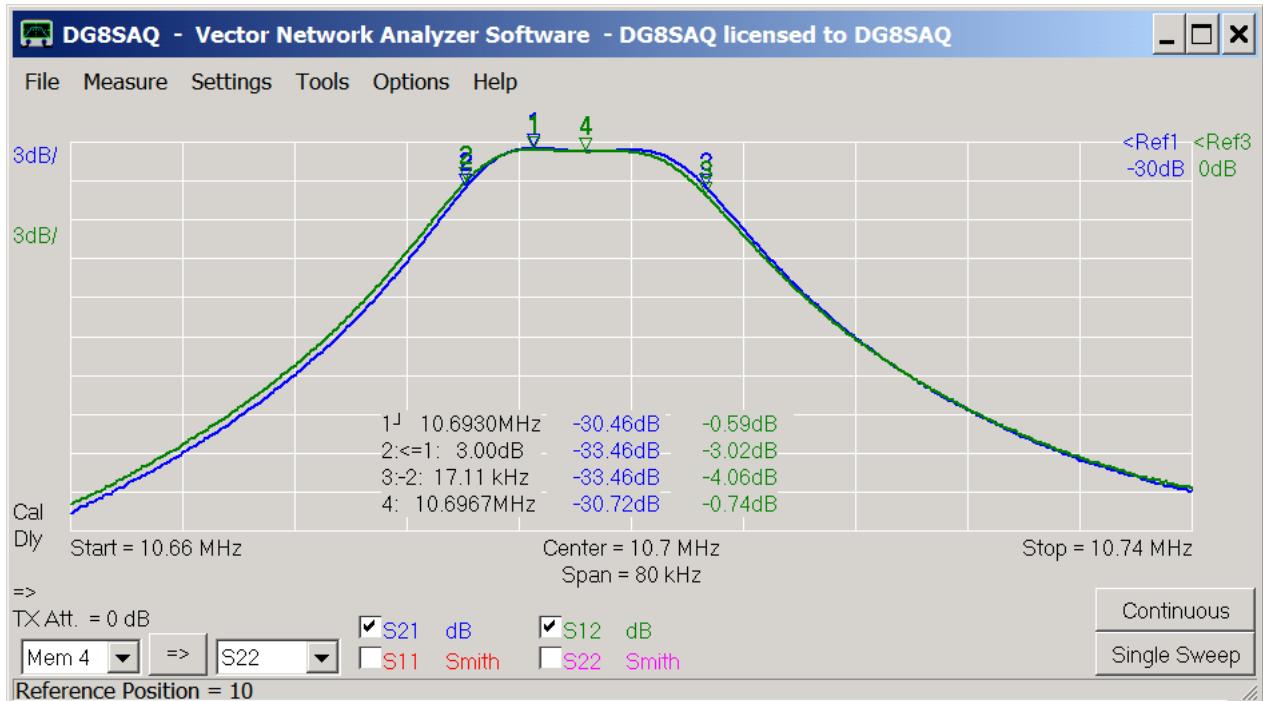


We connect the modified test board with the crystal filter to the VNWA and do a single sweep in forward direction using the original calibration and settings from the S-parameter measurement:



Indeed, the filter passband is flat now, but the insertion loss has increased by about 30dB in comparison to our S12 trace which has survived from our software matching experiment!

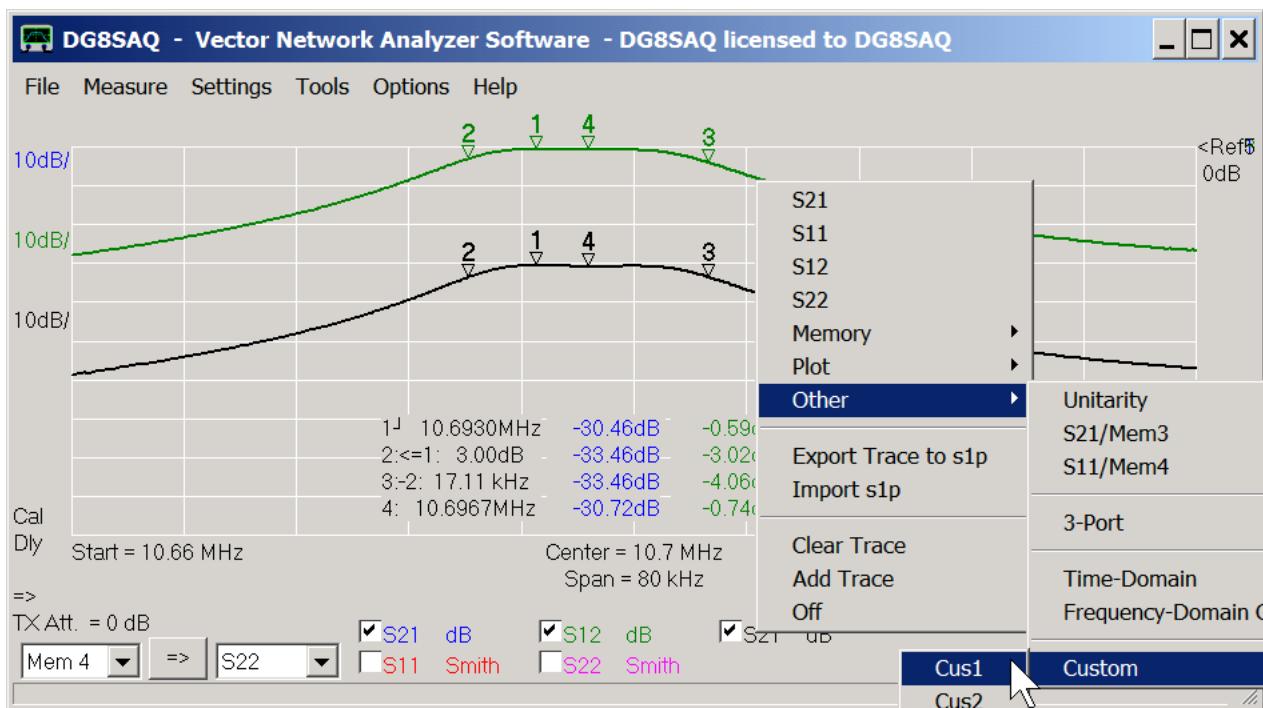
We get a better comparison when we adjust the scales and shift the reference level of the blue S21 trace to -30dB:



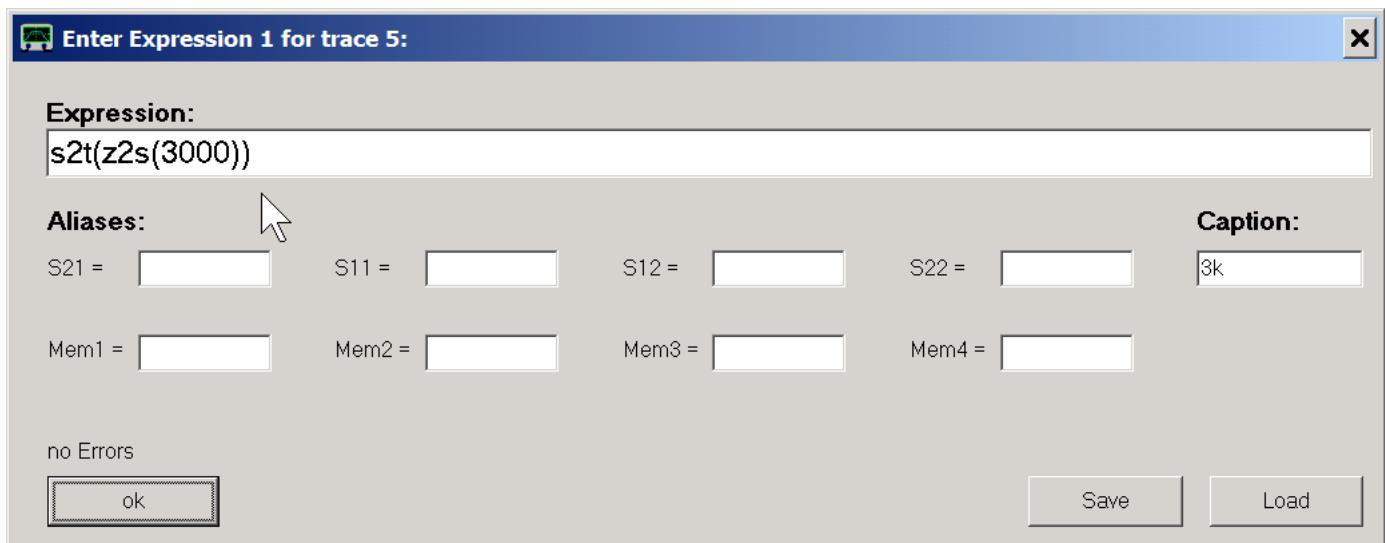
So, except for the high loss, our measurement looks like anticipated.

The 30dB extra loss is no surprise, though, as we have effectively introduced $2 \times 1500 \text{ Ohms} = 3000 \text{ Ohms}$ into the signal path.

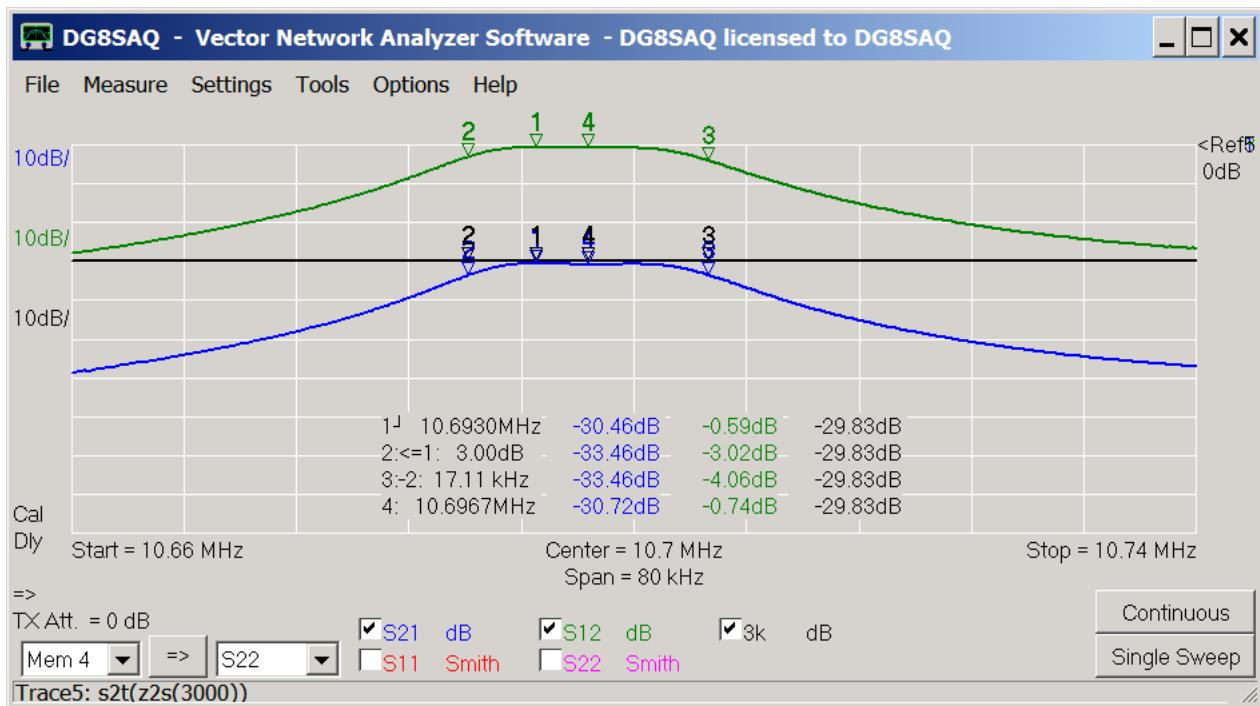
Next, we want to quickly find out, how much extra loss the insertion of 3000 Ohms into the signal path will yield. We add another trace and make it a Custom trace:



The custom trace editor will open:



We edit the expression for the function to be plotted as shown above. The number 3000 denotes our resistor of 3000 Ohms which we want to simulate. The function z2s() converts this resistance into a reflection coefficient. The outermost function s2t() converts the reflection data (which could also be measured as S11 with our 3000 Ohms connected from the TX port to ground) into transmission data (which could be measured as S21 with our 3000 Ohms connected between TX port and RX port). We also assign a descriptive caption, which will later show on the main window. Closing the custom trace editor, the new trace will be computed and plotted:



For clarity, we have reverted to the original scales 10dB/ and a unique reference level of 0dB on top. We can read off the black 3k simulation trace, that 3000 Ohms will introduce an extra loss of 29.83dB, which looks about right compared to the blue S21 trace.

Now, can we correct the green S12 trace by the just computed attenuation to directly compare it to the measured blue S21 trace? Yes, we can. We open the custom trace editor again by double clicking onto the black 3k label and modify the expression such that we multiply S12 with the previous expression:

Enter Expression 1 for trace 5:

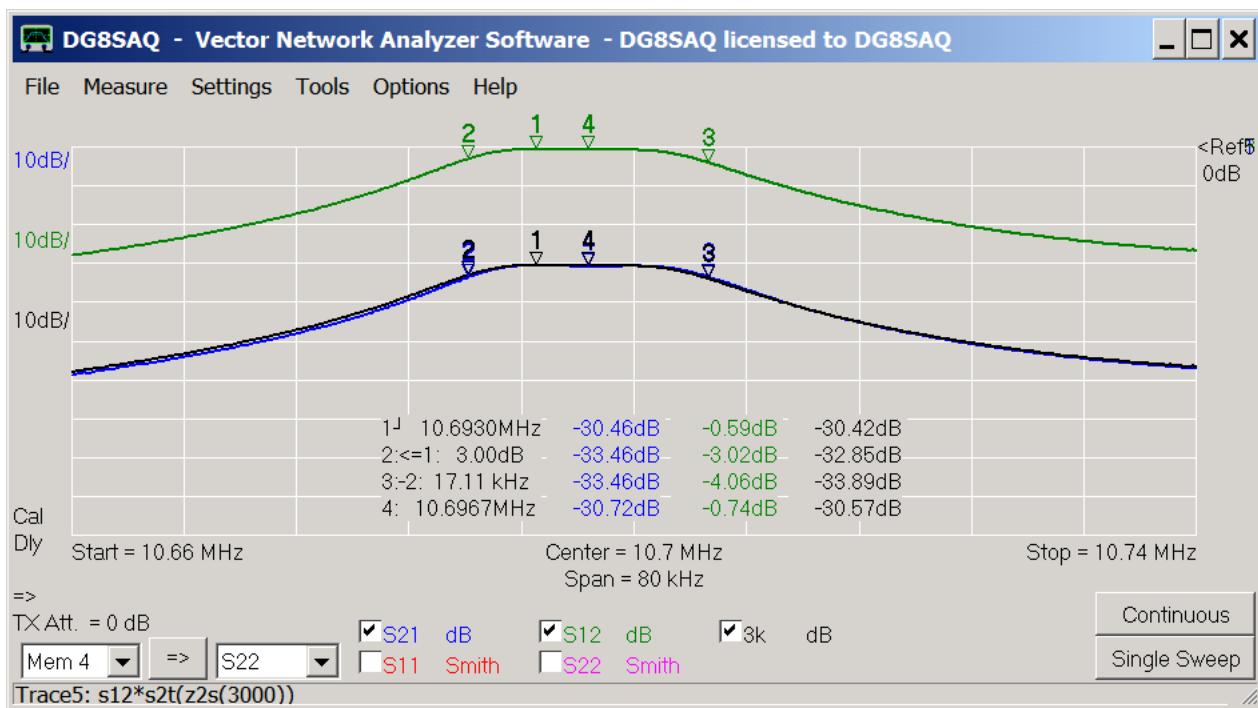
Expression:
s12*s2t(z2s(3000))

Aliases:
S21 = S11 = S12 = S22 = 3k =
Mem1 = Mem2 = Mem3 = Mem4 =

Caption:

ok **Save** **Load**

Indeed, the simulated black transfer characteristic and the measured blue S21 trace are in good agreement:

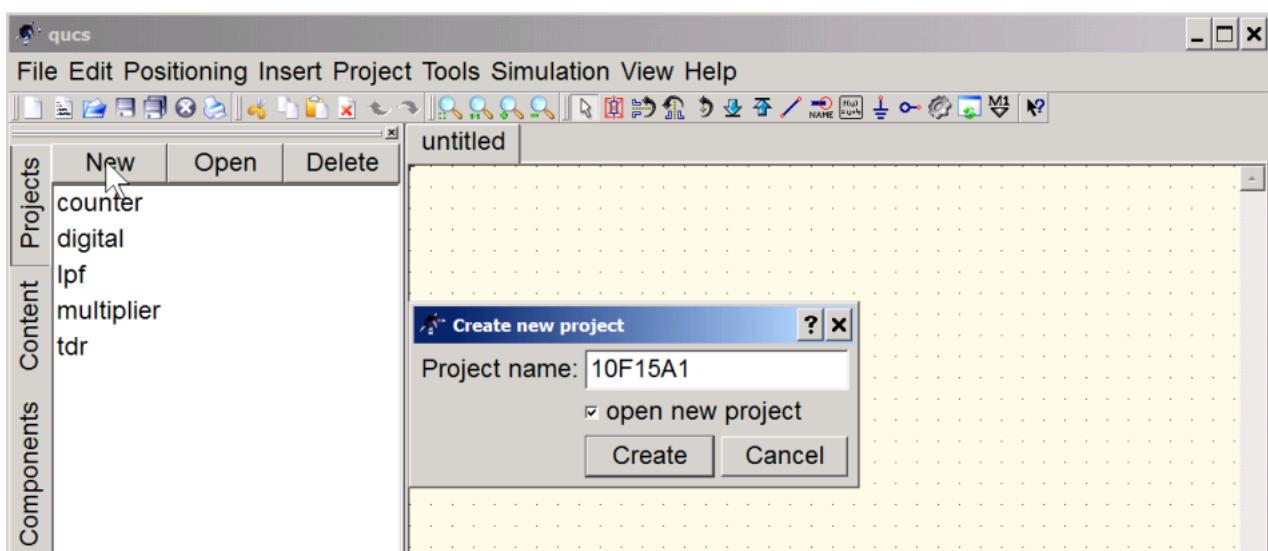


We have arrived at our black simulation curve in a handwaving way. Next, we want to verify it by doing a real network simulation

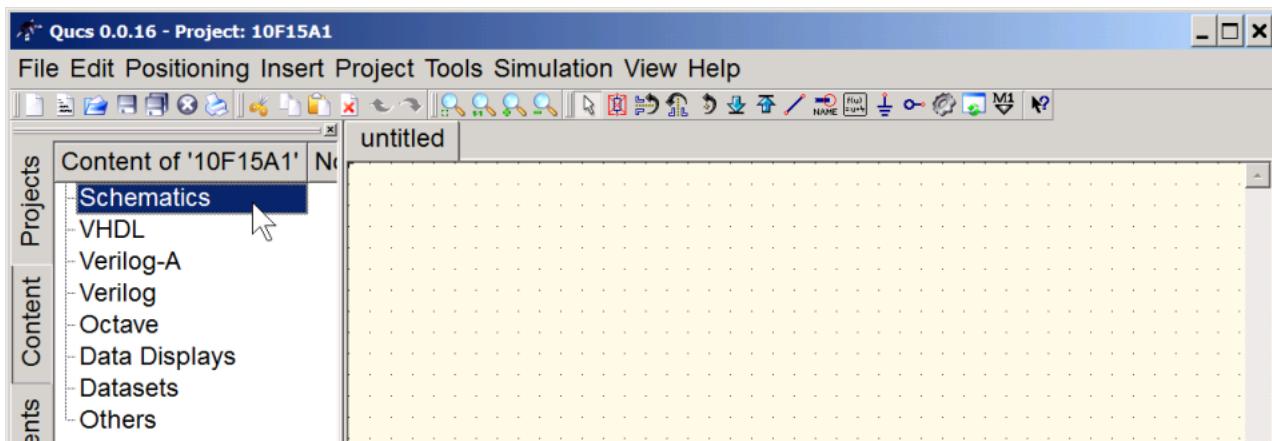
Playing with S-Parameters using the Network Simulator QUCS

QUCS is a powerful yet free network analysis software available for download at <http://sourceforge.net/projects/qucs/>. In the following, we will set up a QUCS project using our crystal filter S-parameters which we have just measured. First, we want to simulate how the filter behaves in between the two 1500 Ohms resistors as measured above.

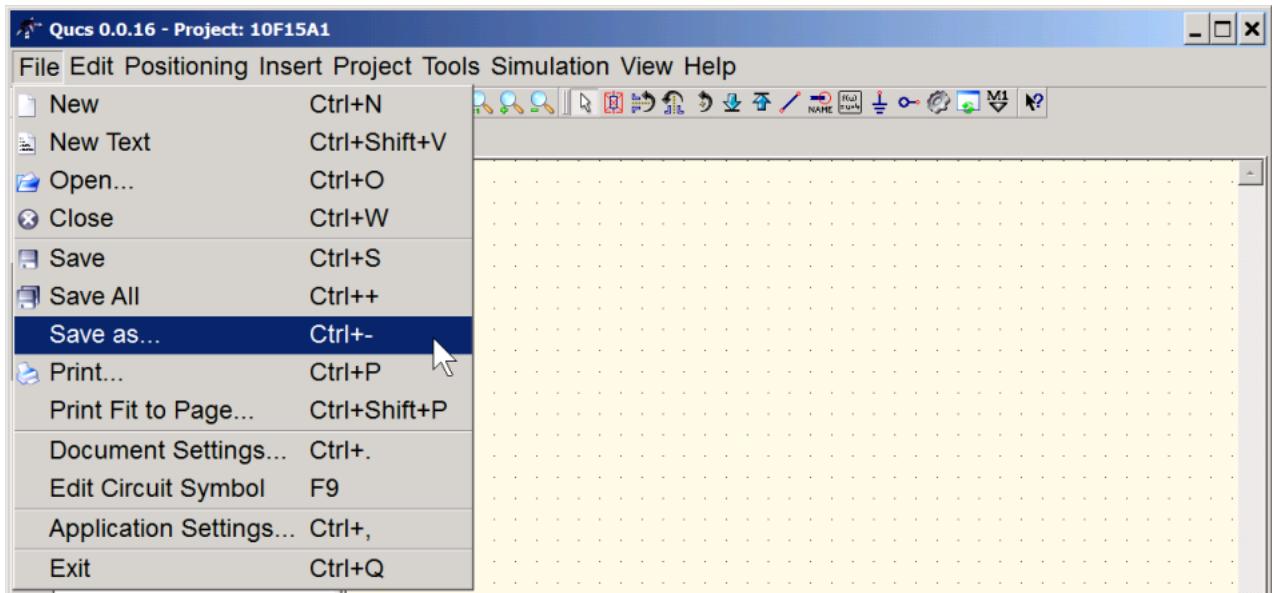
After launching QUCS, we create a new project by clicking onto **New** in the **Projects** tab and by assigning a new project name:



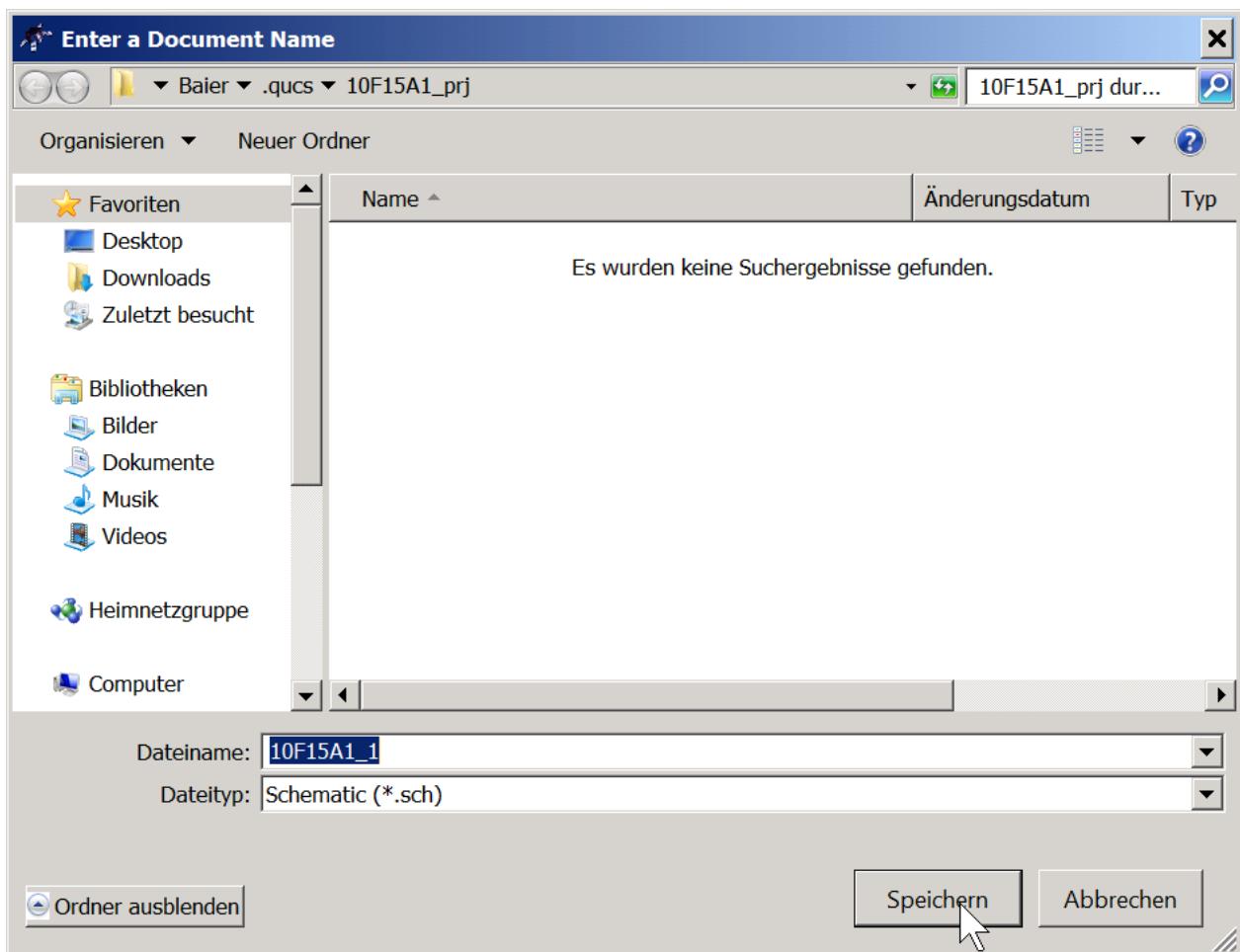
An untitled schematic opens:



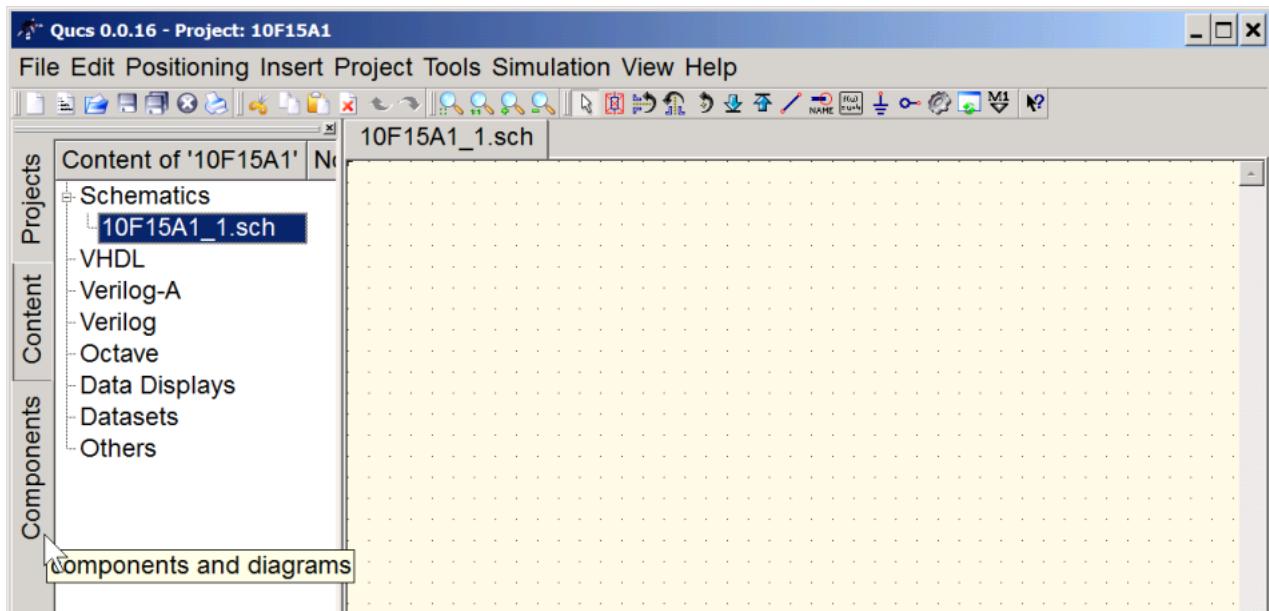
We save it ...



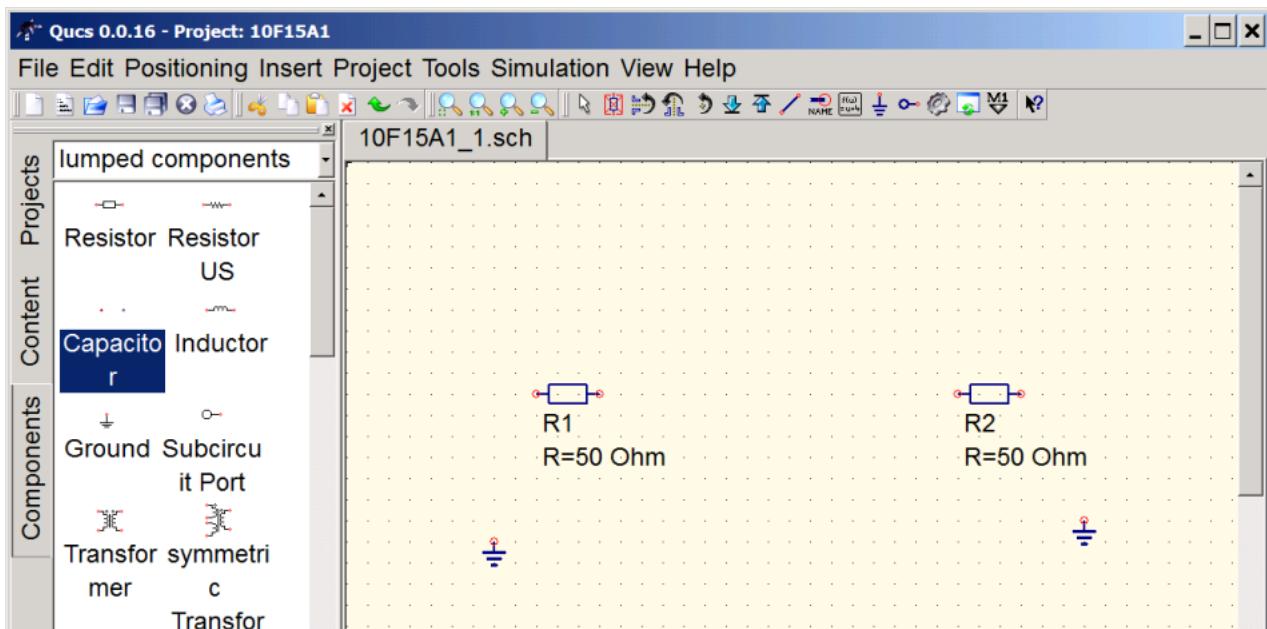
... and thus assign a name to it:



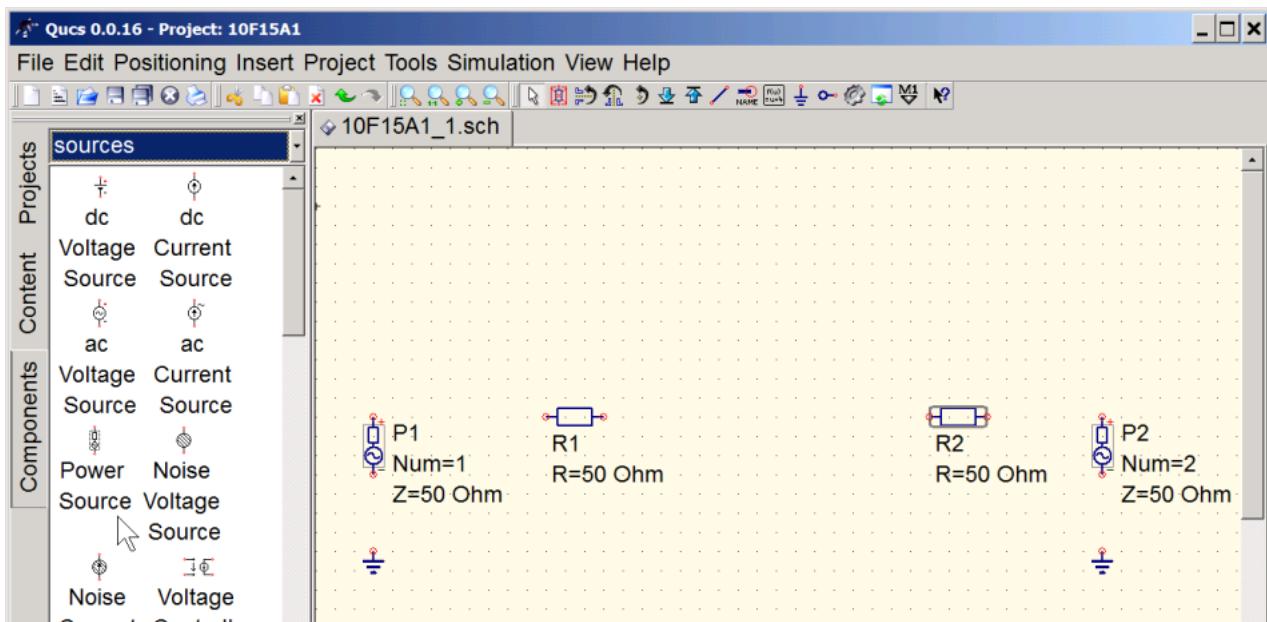
We change to the **Components** tab, ...



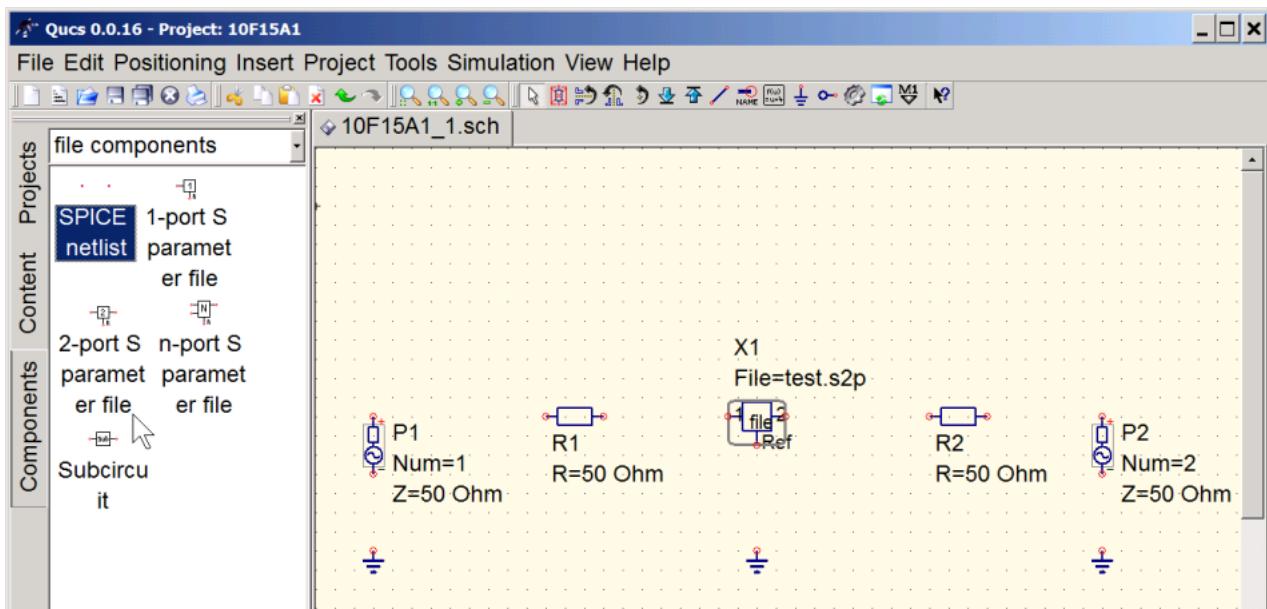
...select **lumped components** dropdown menu and add two resistors and two ground symbols:



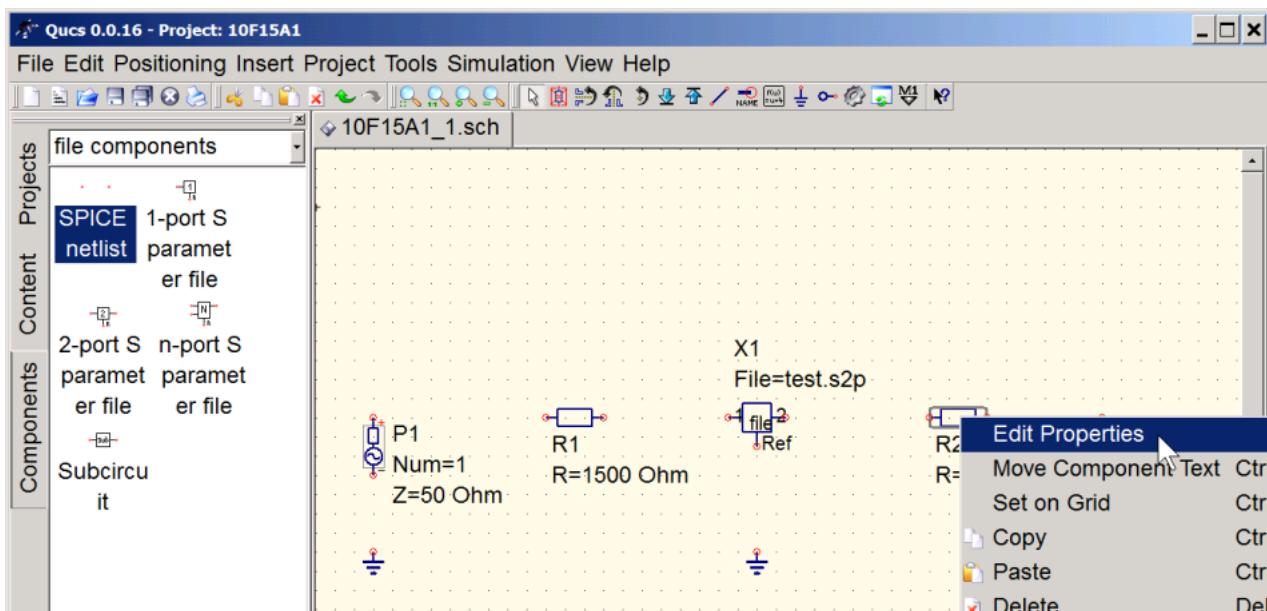
Next, we select the **sources** dropdown menu and add two power sources.



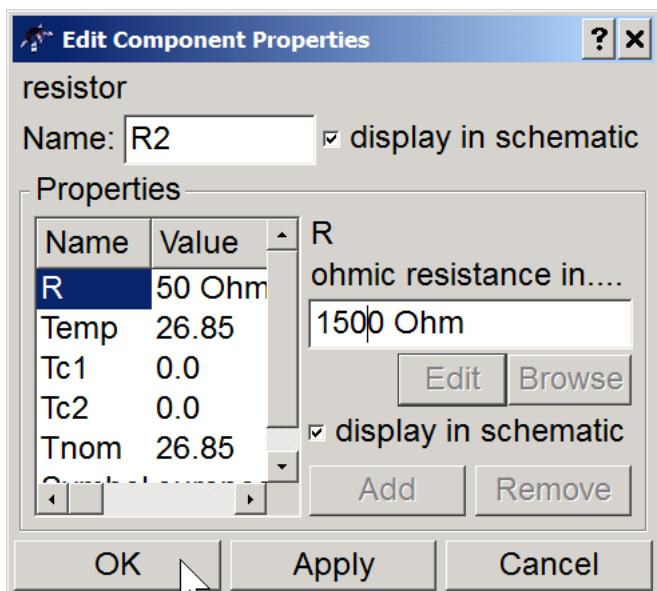
Then, we switch to **file components** and add a **2-port S parameter file** component (and from the lumped components another ground component which we have previously forgotten) :



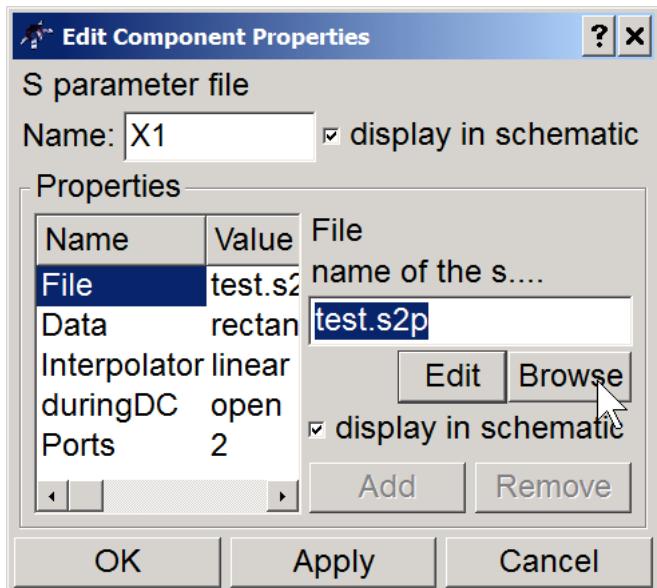
We need to change our resistors to 1500 Ohms, therefore we right-click the resistors and select **Edit Properties**:



We highlight the R-property and change the value to 1500 Ohm

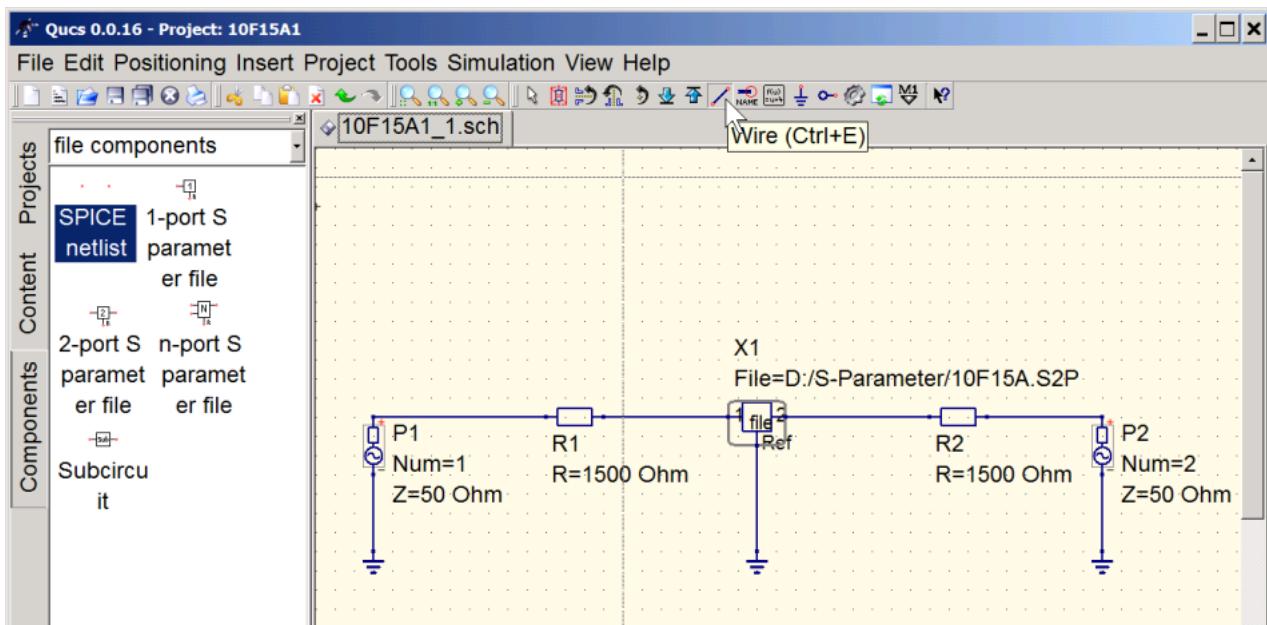


In similar fashion, we edit the property of the **2-port S parameter file** component by right-clicking on it and selecting **Edit Properties**:



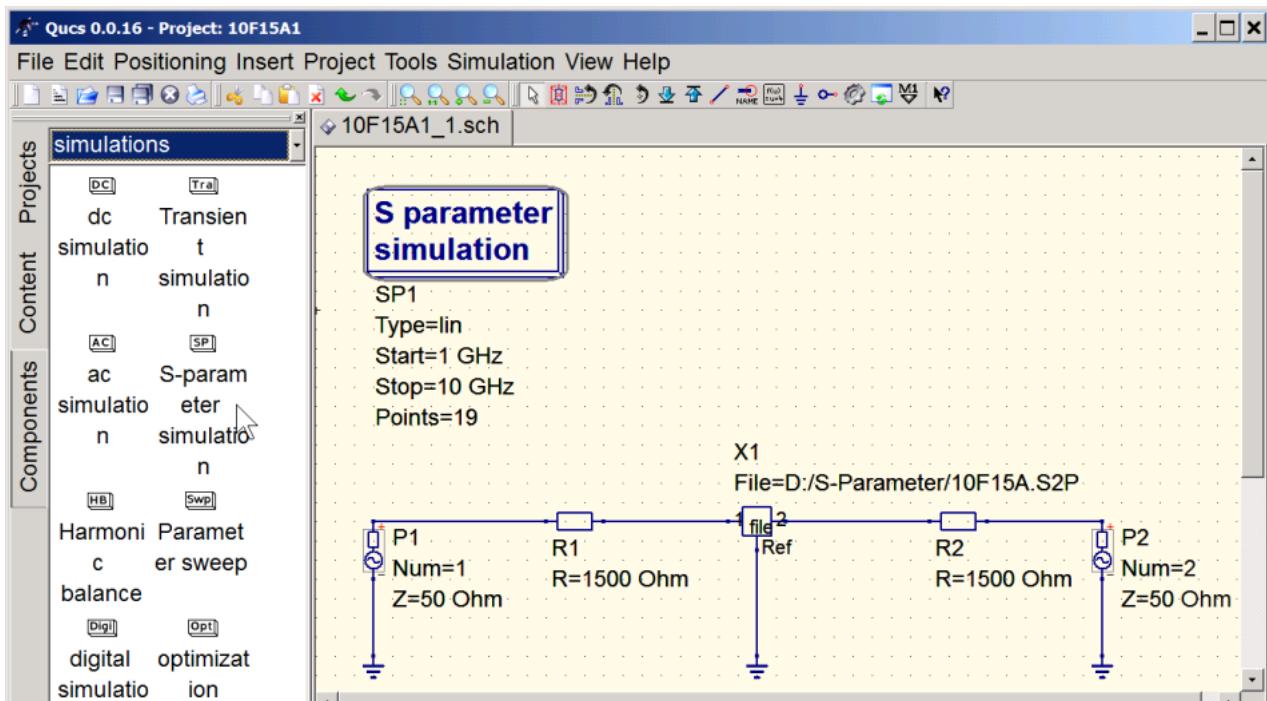
Here, we browse for our measured S-parameter file which we have saved from the VNWA software in the previous section.

Next, we wire the components:

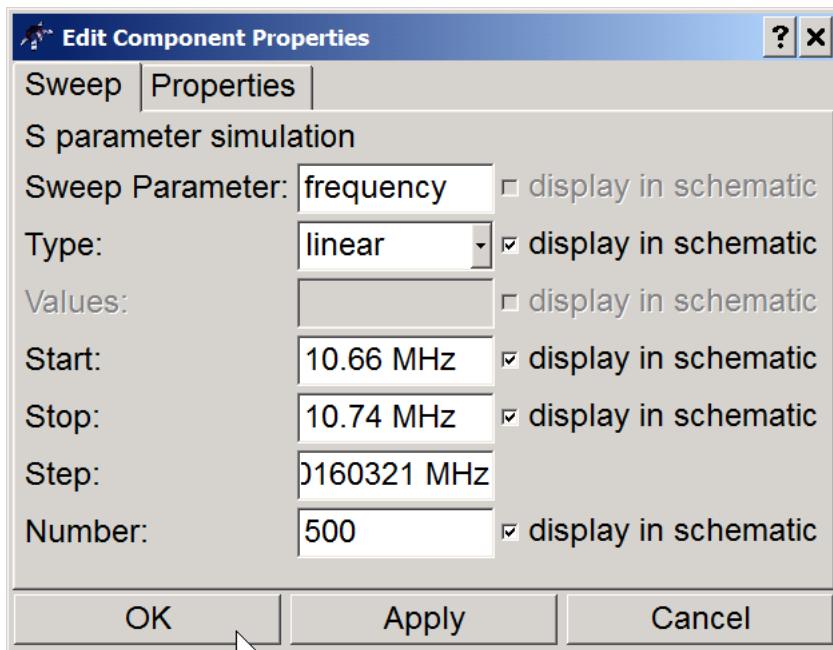


The schematic is done. It looks like the hardware in our previous measurement.

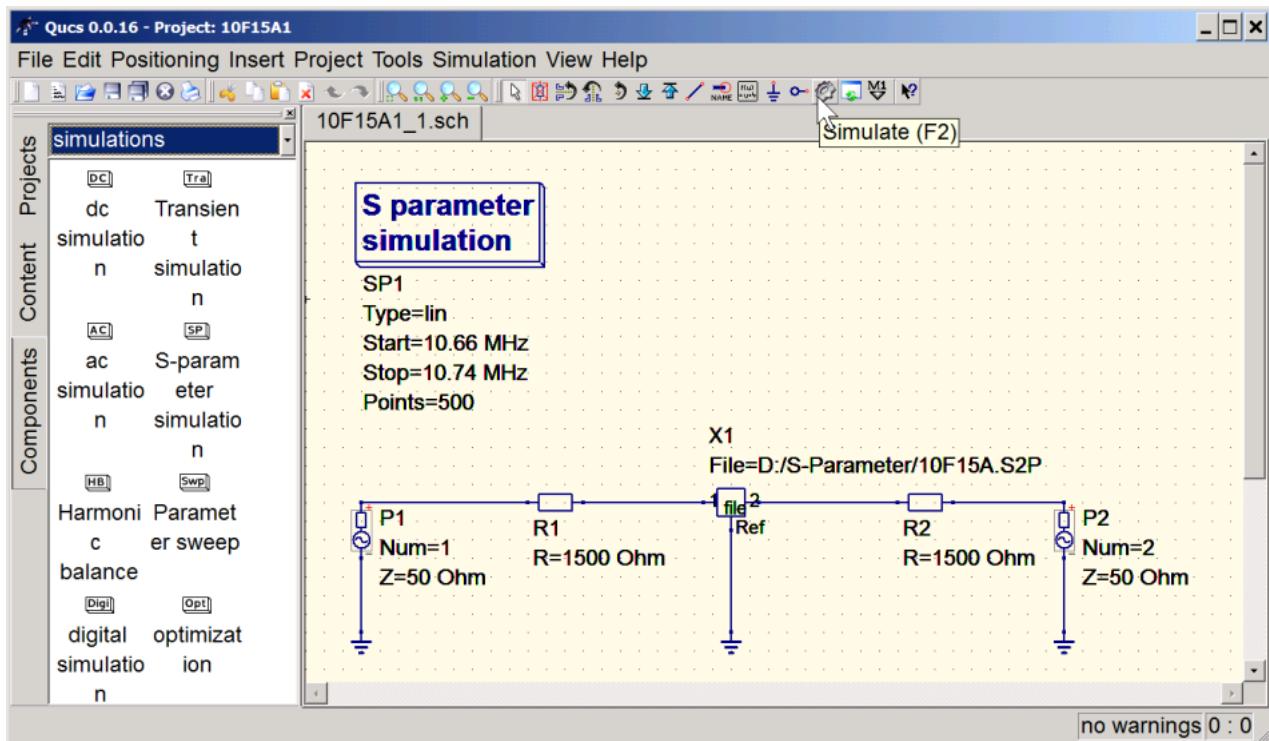
Now, we have to tell the simulator what kind of simulation we want. We change to the **simulations** components and add an **S parameter simulation**:



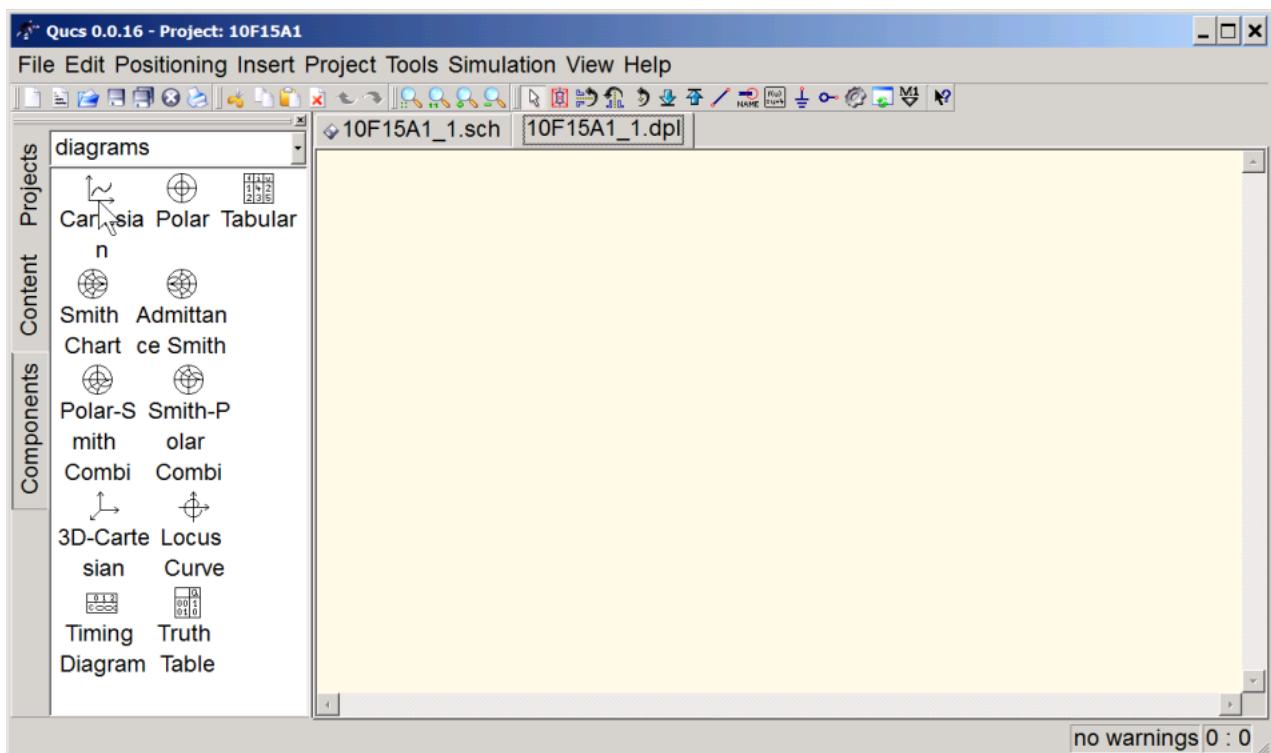
We edit the S parameter properties by right-clicking them and entering the desired frequency range and number of points:



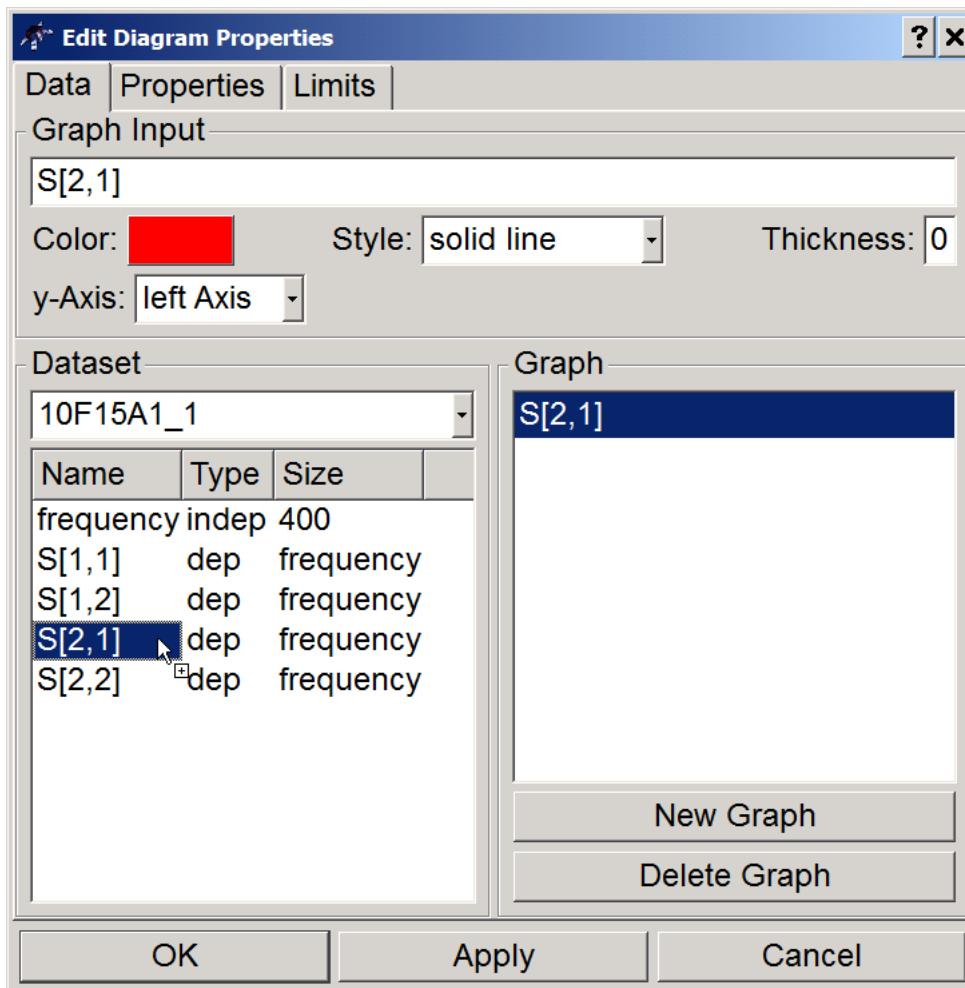
We use the same frequency grid as in our S-parameter file. Now, we can launch the simulation by pressing the **Simulate** button:



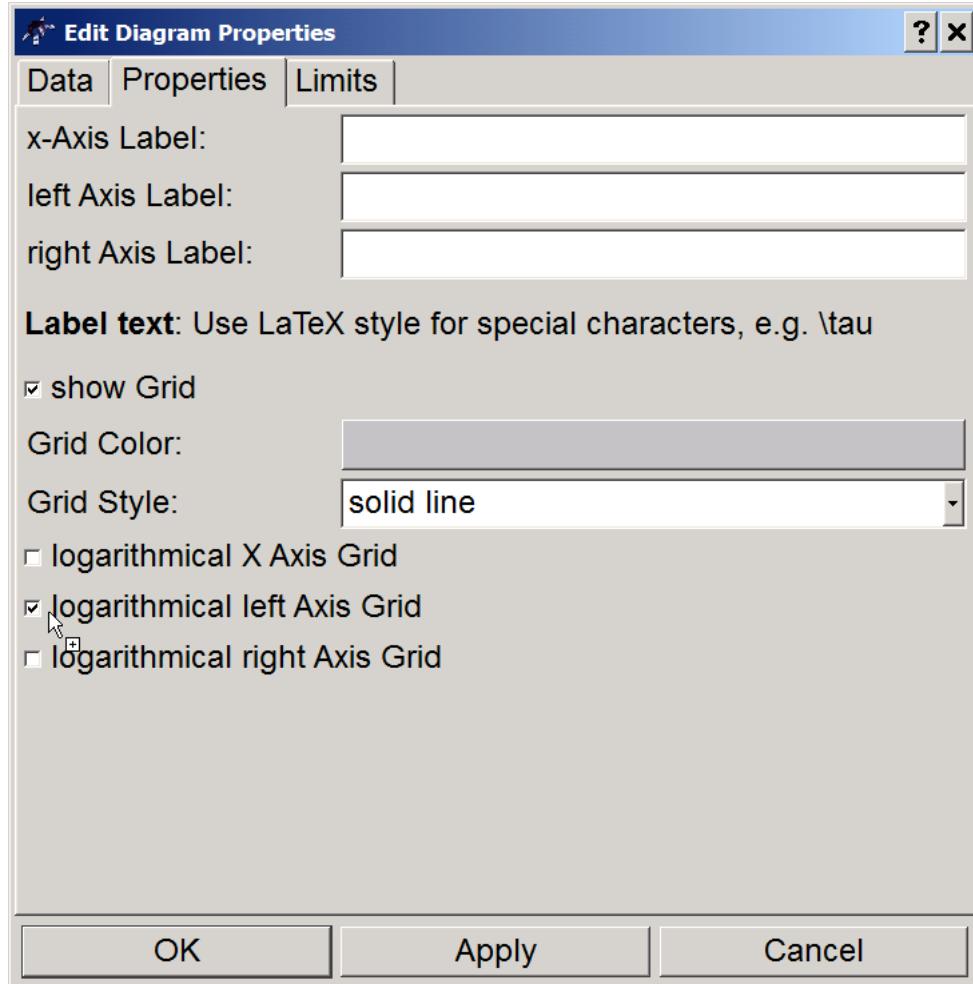
The plot window opens, but it is empty. We still need to add a diagram component. We select a **Cartesian** plot...



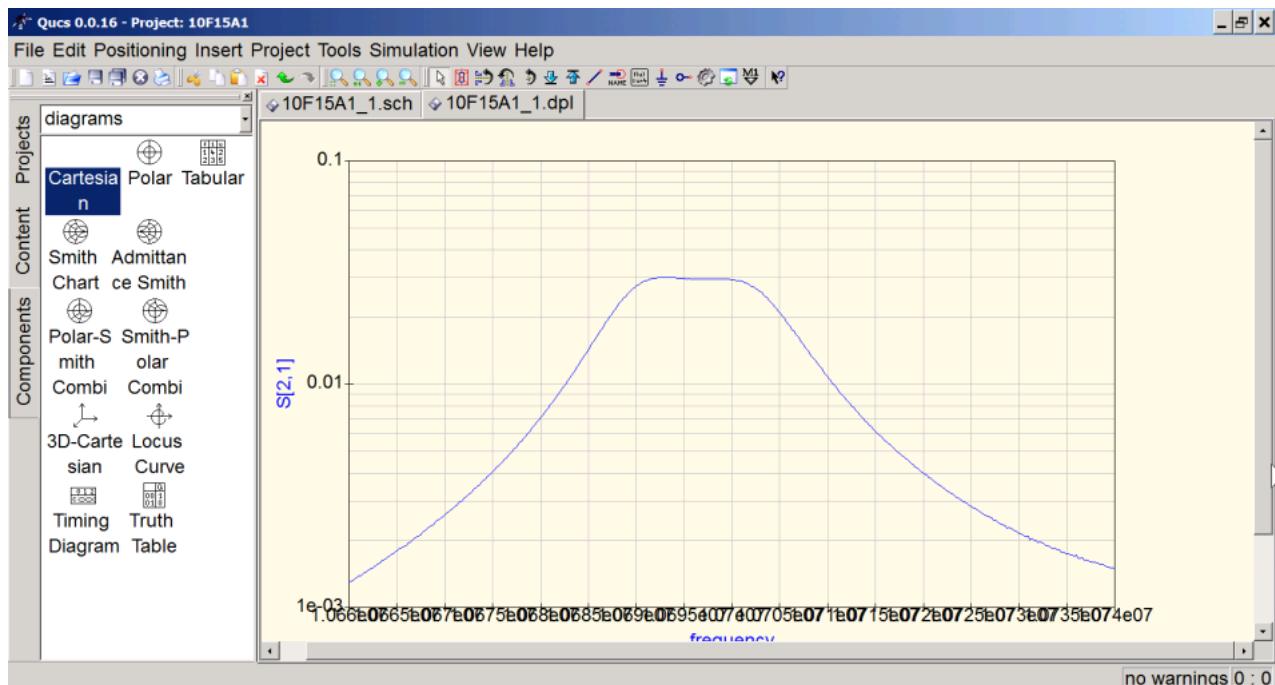
... and add it to the document. The diagram properties pop open. We add an S21 trace by double-clicking onto the highlighted **S[2,1]** property ...



... and specify that we want a logarithmic left axis:

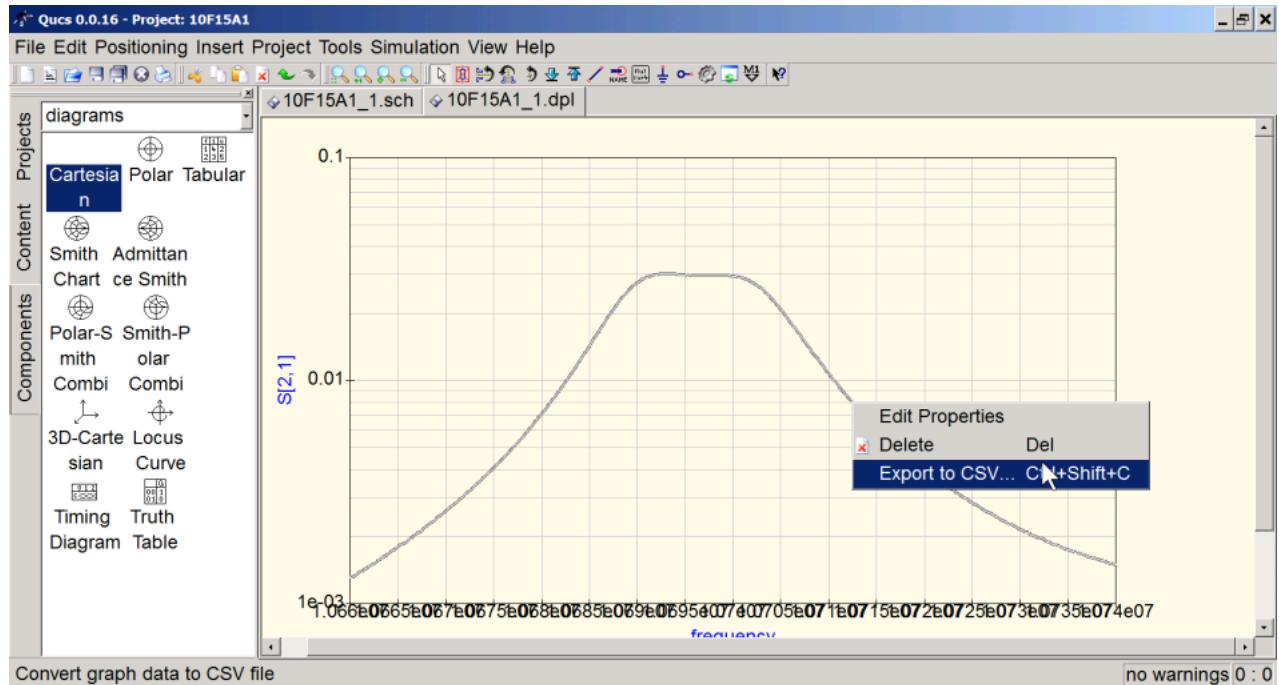


So, finally, we see our simulation result:

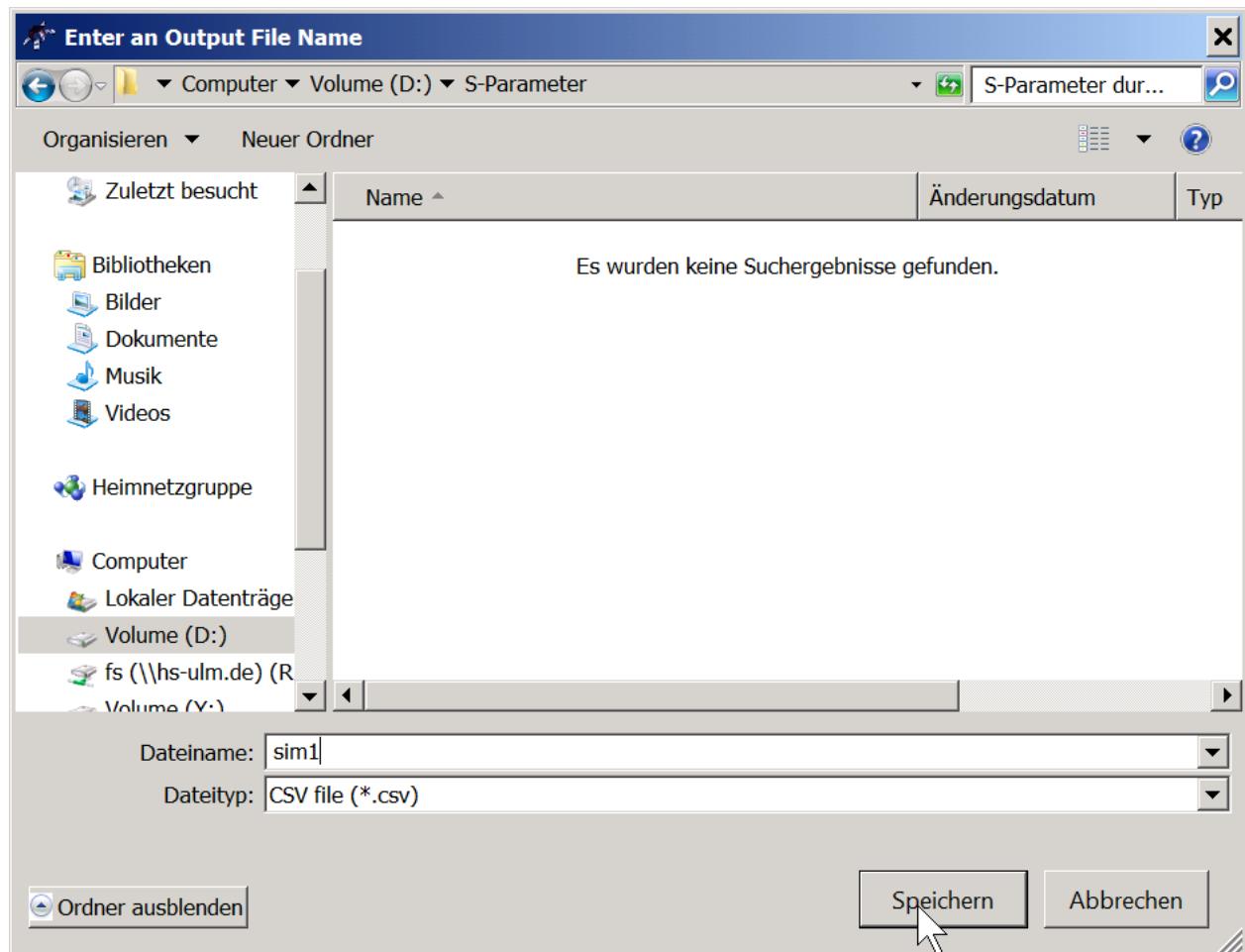


The plot is not very nice, and we want to compare this simulation with our previous measurement. Therefore, we are

going to export the simulation data back into the VNWA software.
To do so, we right-click the blue simulation curve and select **Export to CSV**:

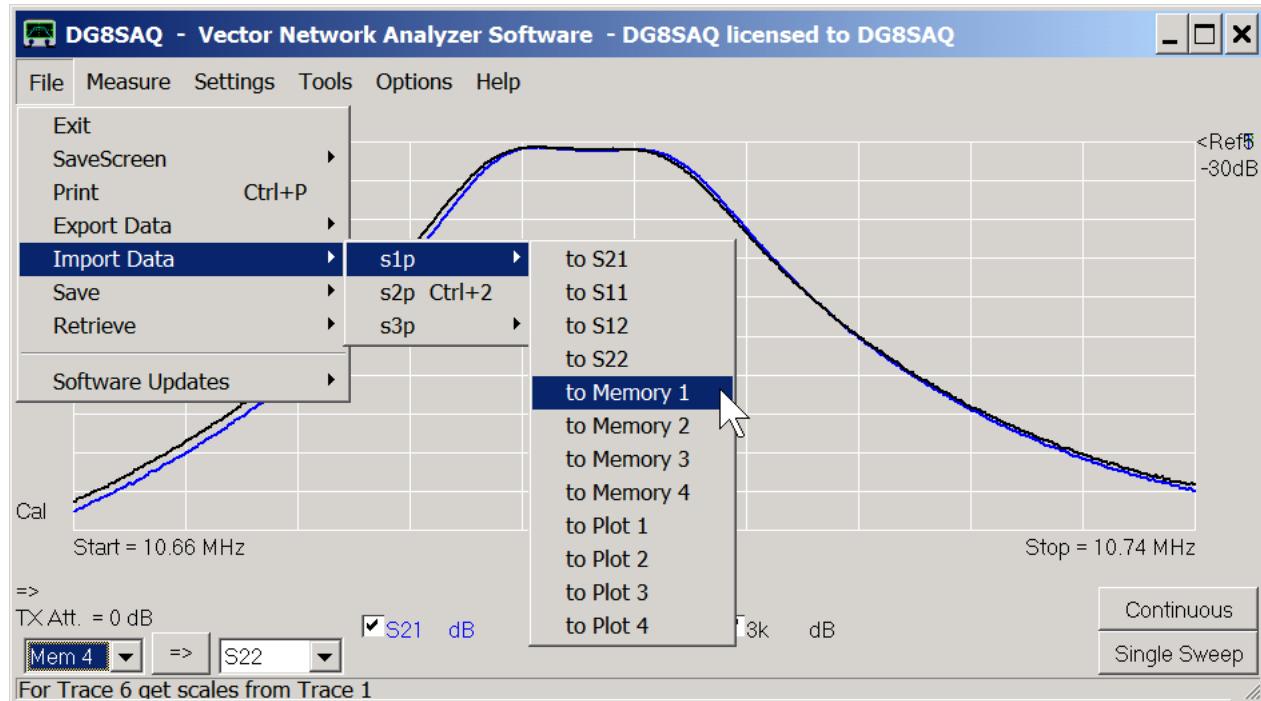


We save the simulation result as csv-file on disk:

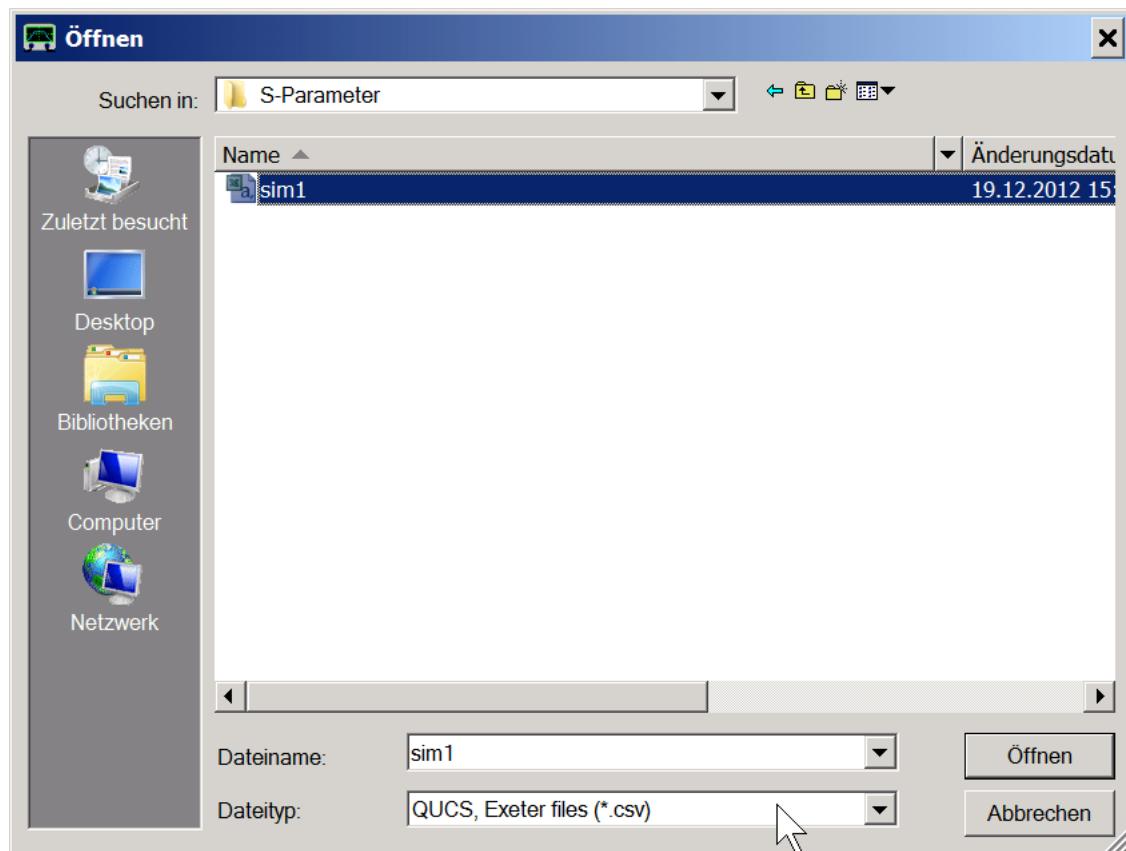


And we switch over to the VNWA software, which still has our old measurement and our handwaving simulation result on

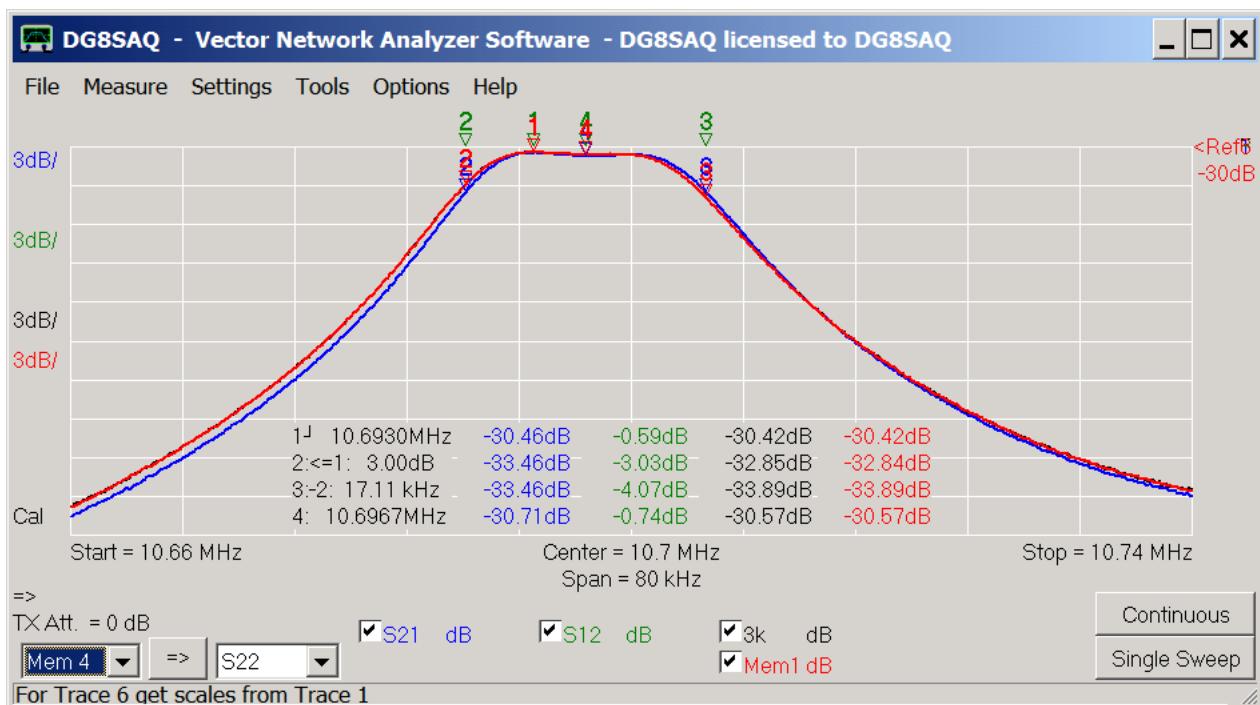
screen. There, we import the just saved QUCS simulation result into Memory 1:



We select file type QUCS, ... (*.csv) and select our simulation file from disk:



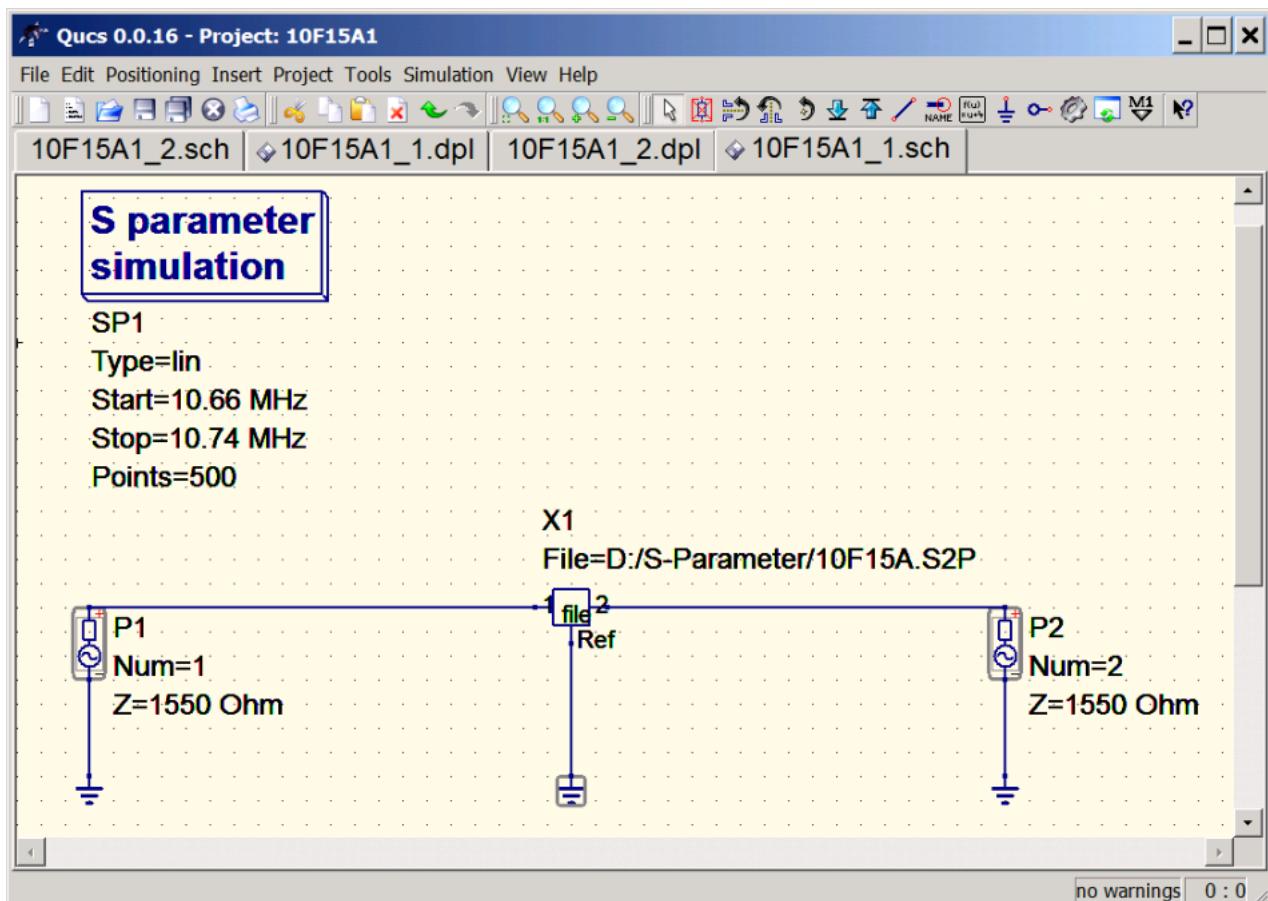
We open another trace to display our QUCS simulation in Mem1:



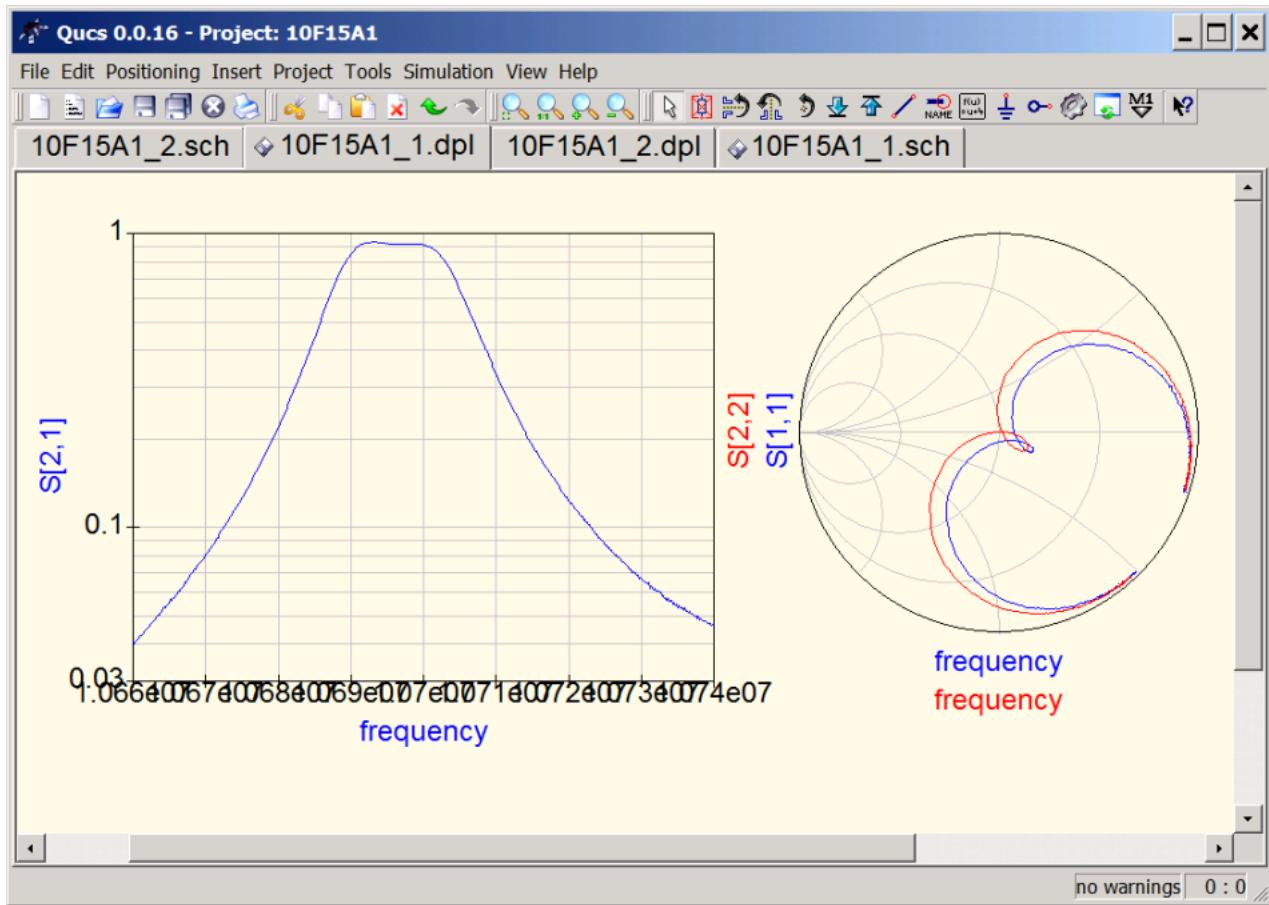
Note, that our handwaving black simulation agrees with the red QUCS simulation to almost the last digit. Therefore, the black trace lies hidden behind the red trace.

What else can we simulate with our S-parameters?

We can simulate what our filter would look like in a 1550 Ohms environment like we have done with the VNWA matching tool. To do so, we remove the resistors and change the sources' impedances to 1550 Ohms:

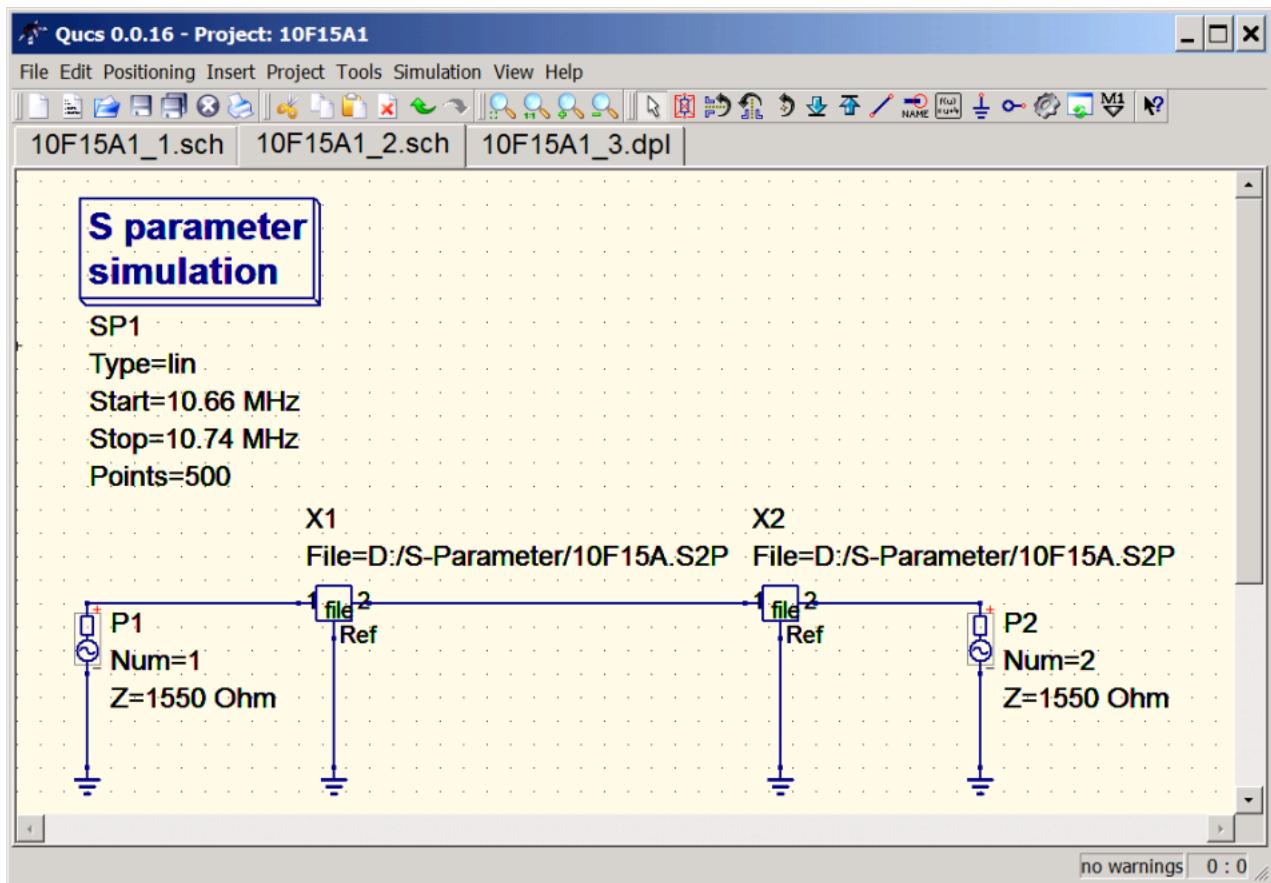


Not surprising, the result is identical with the VNWA result:

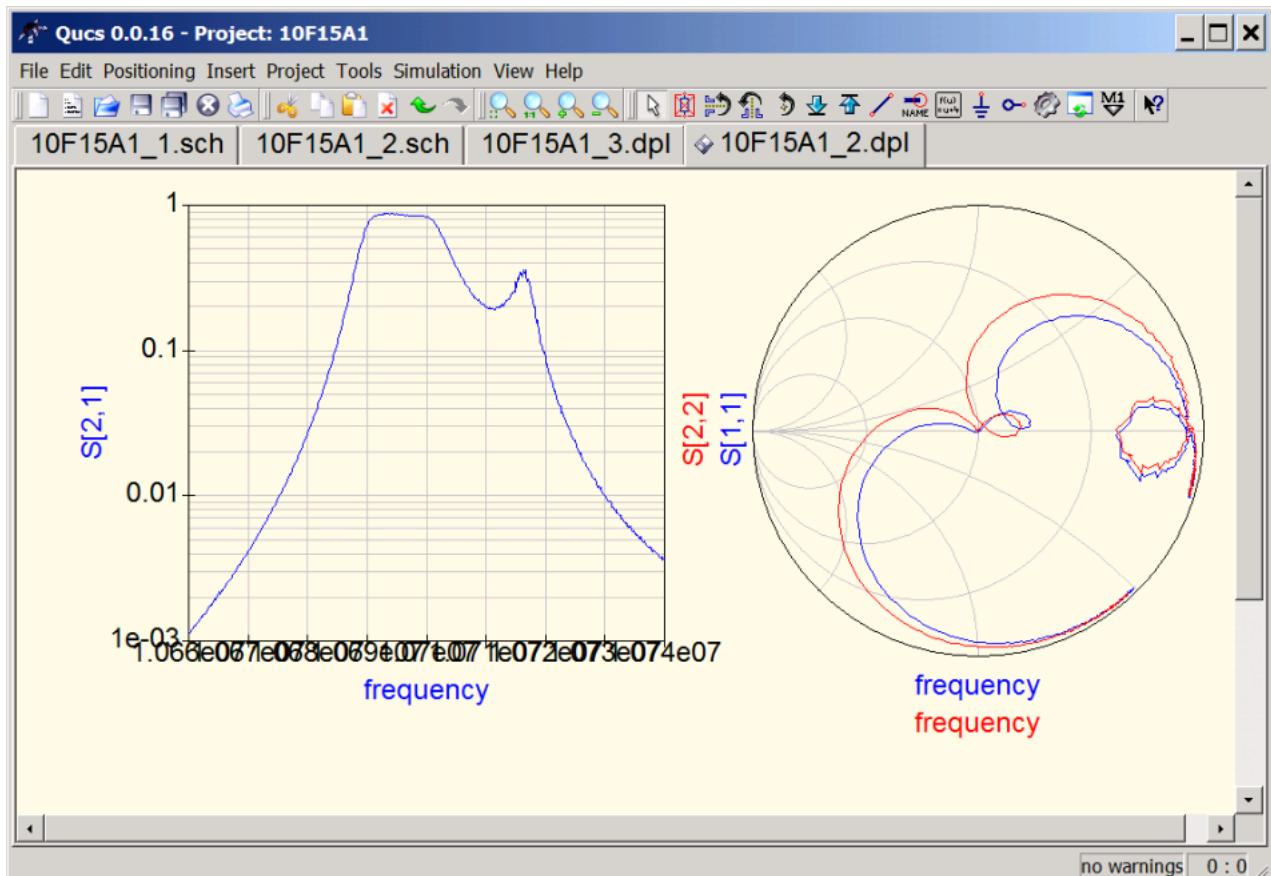


We have added a Smith chart to display the simulated S₁₁ and S₂₂.

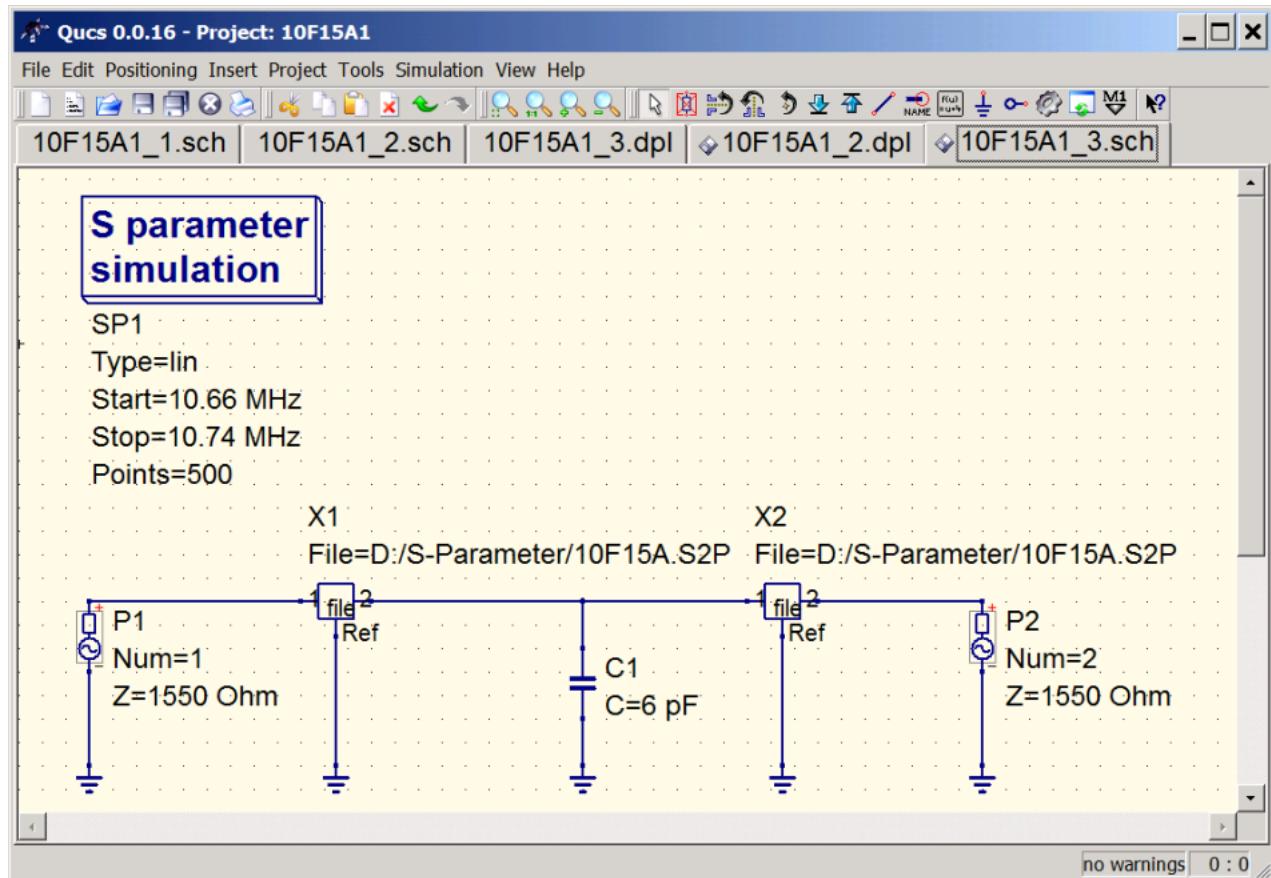
Note, that the stopband attenuation of our two pole filter is not very high. We might improve it by cascading two of these filters. We can easily simulate the expected performance by adding another S parameter file component which actually points to the same file as the first one, since we want to cascade two identical filters:



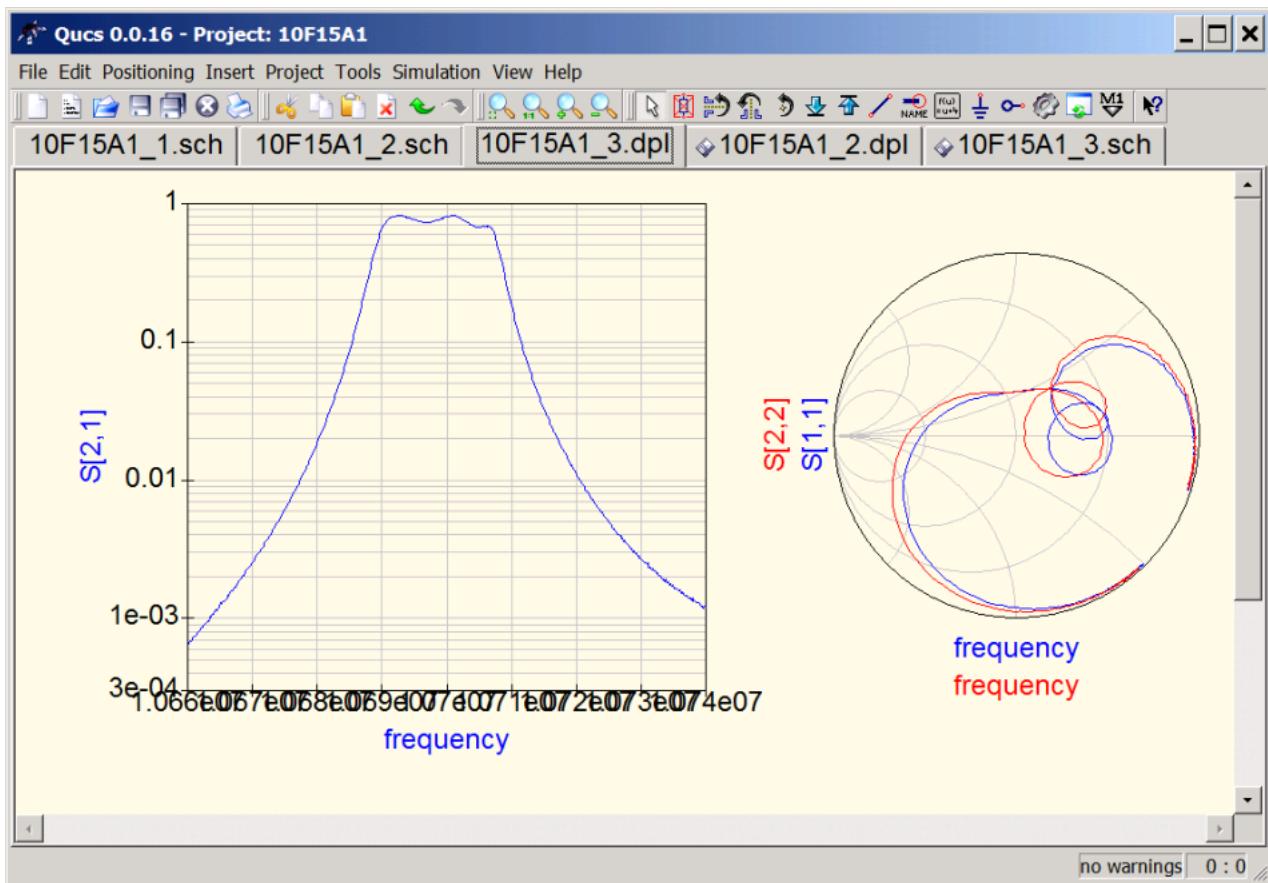
The simulated S21 data shows an ugly hump at the high frequency skirt which is due to internal mismatch between the two pole filters:



We add a capacitor to the internal filter connection to improve the match:



Now, we have a decent passband and a considerably larger stopband attenuation.



We could easily continue to optimize internal and external matching and thus design a perfect four pole crystal filter. Obviously, a simulator like QUCS together with the VNWA for acquiring S parameter data is a powerful pair for RF design.

TIME DOMAIN MEASUREMENTS

The VNWA software allows to perform a realtime **FFT** on measured or imported data from frequency to time domain, display data in time domain, manipulate data in time domain by **gating** and perform an **inverse FFT** back to frequency domain.

A special topics section on time domain measurements demonstrates examples of both usages:

Time domain reflectometry is a useful technique to search defects in transmission lines.

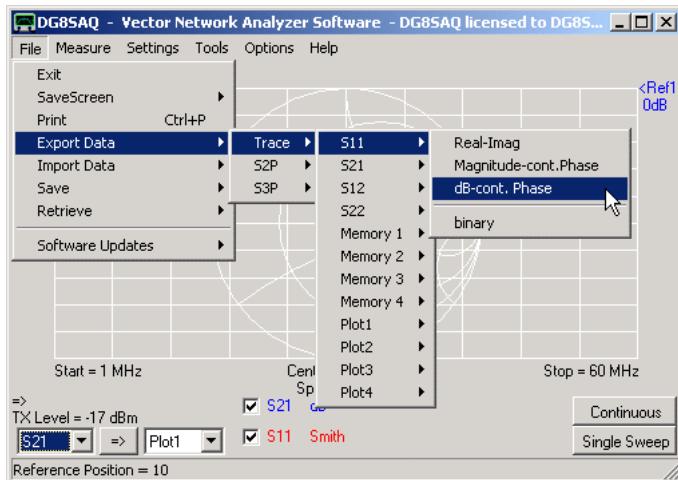
The step response is useful to determine impedance variations along a coaxial cable.

Gating is used to separate responses depending on their arrival times, e.g. to separate the slow mechanical response of a crystal filter from the fast electromagnetic feedthrough of the test board.

FILE

The VNWA main menu "File" offers the following functions:

- Exit**
- Save Screen**
- Print**
- Export Data**
- Import Data**
- Save**
- Retrieve**
- Software Updates**



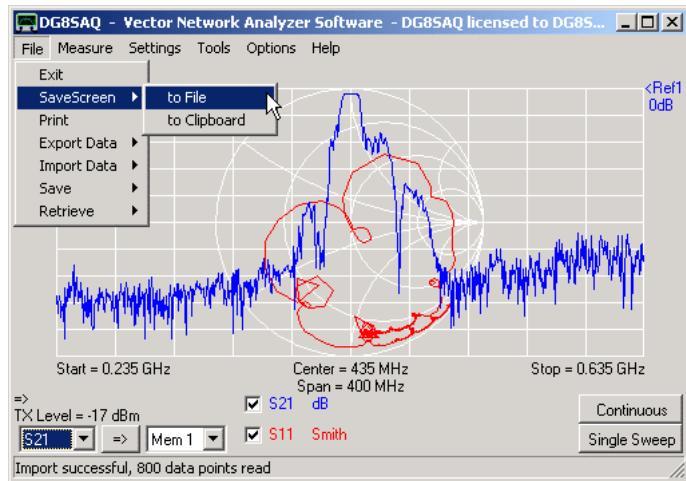
FILE - EXIT

Exit

Exit the VNWA software. All settings will be saved for the next session.

FILE - SAVE SCREEN

Save Screen...

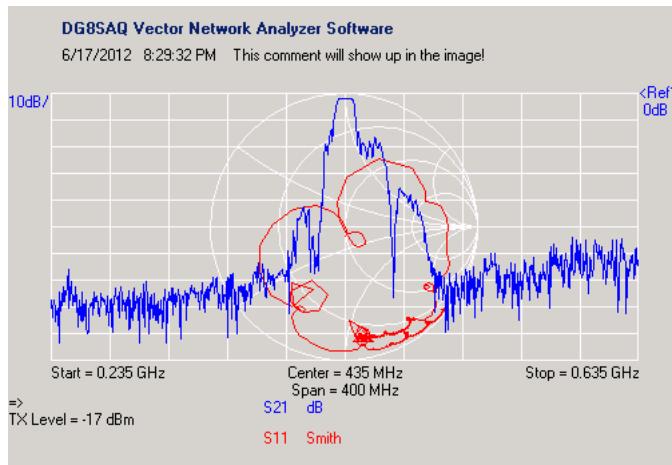


...to File

Save the current main graphics window into a bmp, jpg or png file. When saving, you will first be asked for a file name, then an input mask for an optional image comment pops up:



Enter a **comment** or press no comment. The exported image with the above comment will look like this:



Note, that the buttons and menus are suppressed. The image size in pixels is identical with the VNWA window client size. If you want the saved image to be bigger, increase the VNWA window size.

...to Clipboard

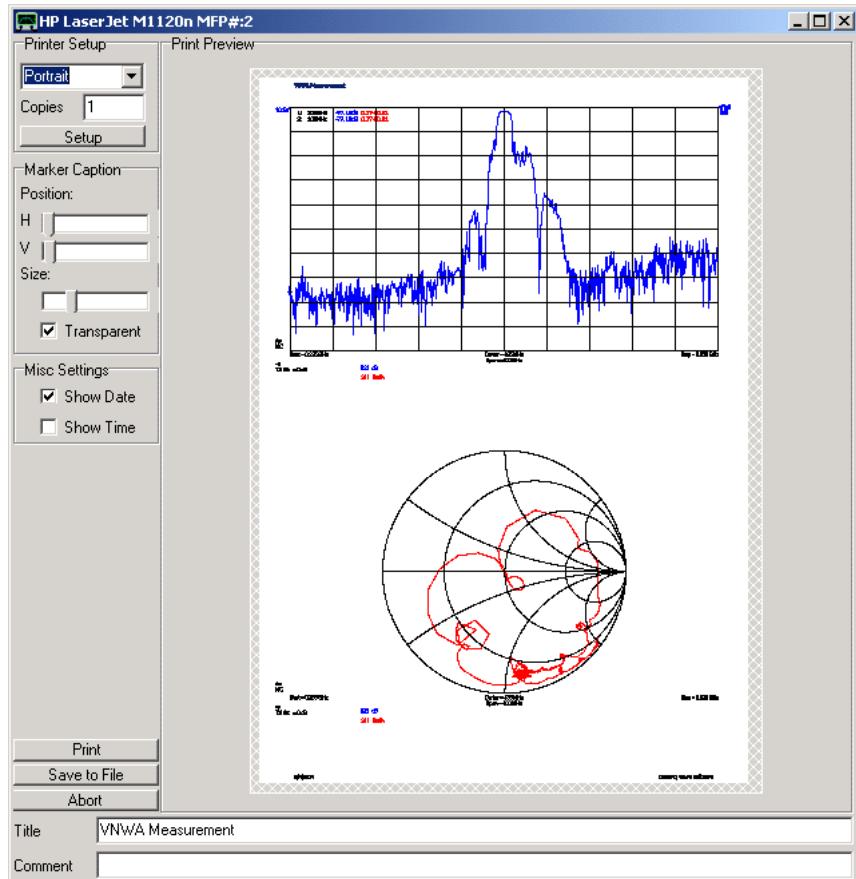
copies the main window client area as bitmap to the Windows clipboard. From there, the image can be pasted into documents or presentations e.g. by pressing ctrl-V.

FILE - PRINT

Print

Prints the current main graphics window.

Upon activation the print control window comes up:



On the right hand side, a preview of the printout including the paper size and printable area are visible.

In the **Printer Setup** box you can:

- select portrait or landscape style
- select the number of printed copies
- setup the printer, e.g. select a different printer.

The selected printer shows in the blue window title bar.

In the **Marker Caption** box you can:

- change the horizontal and vertical positions and the size of the marker caption labels in the printout
- for better readability transparency of the marker caption labels can be switched off.

In the **Misc Settings** box you can:

- activate/deactivate printout of the current date
- activate/deactivate printout of the current time

On the bottom a **Title** and a **Comment** can be specified to be printed out together with the graphs.

The **Print** button sends the image to the selected printer.

The **Save to File** button saves the image to a file without sending it to the printer.

The **Abort** button closes the print window without printing.

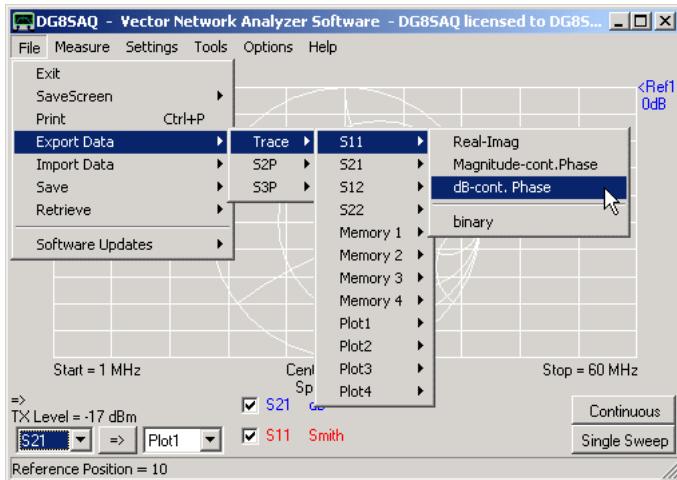
FILE - EXPORT DATA

Export Data

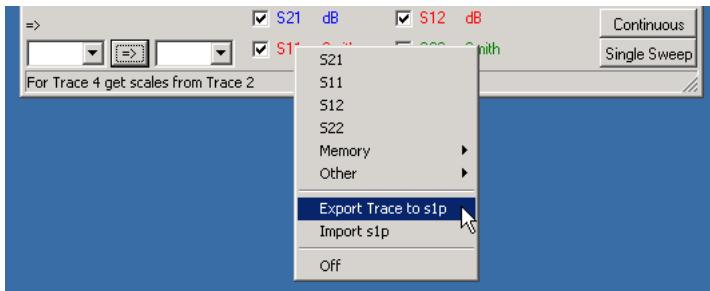
►►► **Note:** Exported data is always normalized to 50 Ohms.

Export Trace

VNWA can export any trace data into a Touchstone style *.s1p-file or a binary *.v1b file. To do so, select the main menu item "Export Data"- "Trace"-....



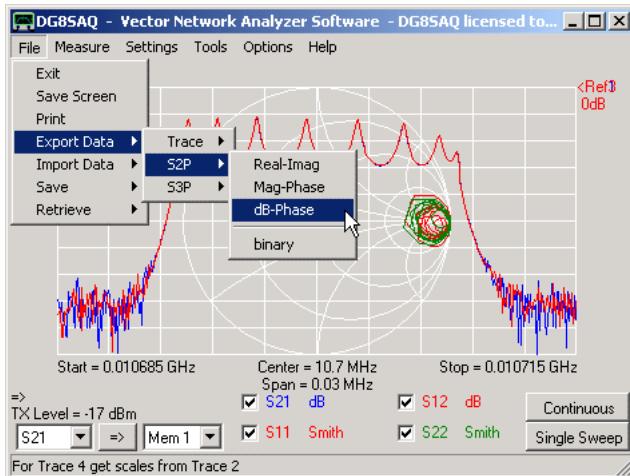
You can also export a trace's data by right-clicking onto the according main window's trace label:



►►► **Note:** The s1p-file standard defines, that an s1p-file must contain one-port reflection data. Nevertheless, I use the same format to store S21-data and other trace data as well, as it is a useful format.

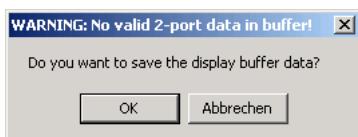
Export S2P or S3P

You can export a complete set of 2-port (3-port) S-parameters to a *.s2p-Touchstone file (*.s3p-Touchstone file) in various formats or to a *.v2b binary file



►►► **Note:** Exported data is always normalized to 50 Ohms.

►►► **Note:** If VNWA doesn't hold valid 2-port S-parameters, you are asked, if the data contained in the display data spaces (S21,S11,S12,S22) should be saved instead:



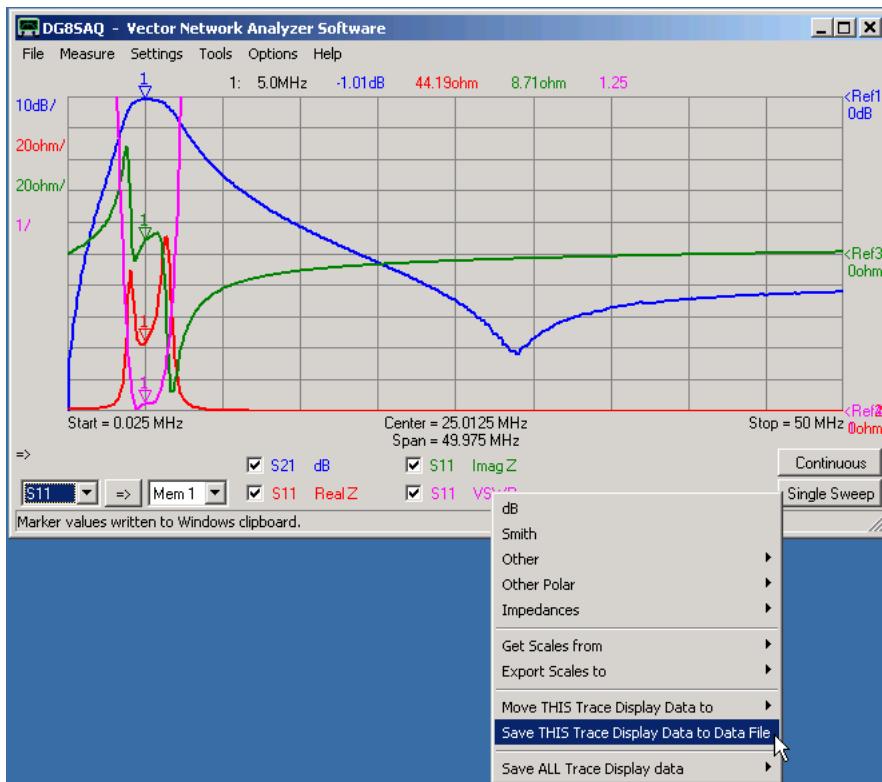
►►► **Note:** If VNWA doesn't hold valid 3-port S-parameters, an error message is displayed:



Export Display Data

Data can also be exported as displayed, e.g. as VSWR, impedance, dB,

If you want to e.g. export displayed VSWR-values, right-click the VSWR label and select "Save THIS Trace Display Data to Data File":



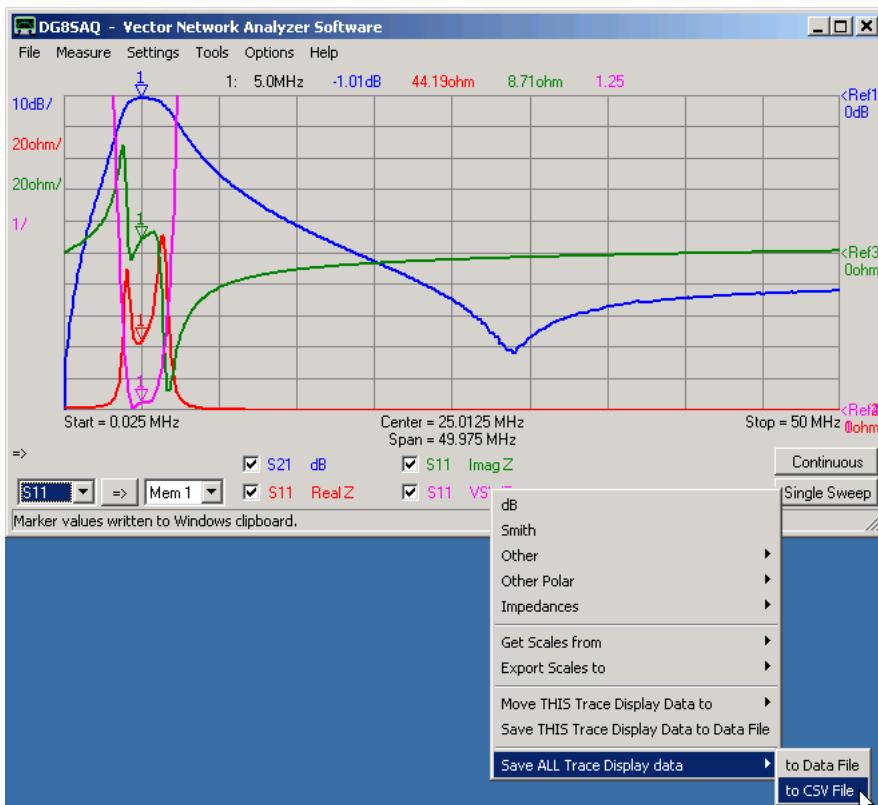
The data will be saved as ASCII *.dat frequency-value pairs with frequency given in Hz:

```

Lister - [c:\tmp\vsqr.dat]
Datei Bearbeiten Optionen Codierung Hilfe   6 %
2.50000000000000E+0004 3.84848229667418E+0001
1.58258626566416E+0005 3.80970098257350E+0001
2.75501253132832E+0005 3.76516345332977E+0001
4.00751879699248E+0005 3.72483851918688E+0001
5.26002506265664E+0005 3.67318938504056E+0001
6.51253132832080E+0005 3.61222453530426E+0001
7.76503759398496E+0005 3.53505880560941E+0001
9.01754385964912E+0005 3.46385499286121E+0001
1.02700501253132E+0006 3.37762325561865E+0001
1.15225563909774E+0006 3.28572168778373E+0001
1.27750626566416E+0006 3.19508571033166E+0001
1.40275689223058E+0006 3.08848957596705E+0001
1.52800751879699E+0006 2.97472946579285E+0001
1.65325814536341E+0006 2.86173137853440E+0001
1.77850877192982E+0006 2.74133419131909E+0001
1.90375939849624E+0006 2.61205511688826E+0001
2.02901002506266E+0006 2.48232442102542E+0001
2.15426065162907E+0006 2.35083503737653E+0001
2.27951127819549E+0006 2.21159880904574E+0001
2.40476190476190E+0006 2.07157932458545E+0001
2.53001253132832E+0006 1.92437213995629E+0001
2.65526315789474E+0006 1.77694016797048E+0001
2.78051378446115E+0006 1.62117814586998E+0001

```

It is also possible to save ALL displayed data at once into a single file, either a *.dat ASCII data file or a csv-file:



A csv-file can easily be imported into Excel:

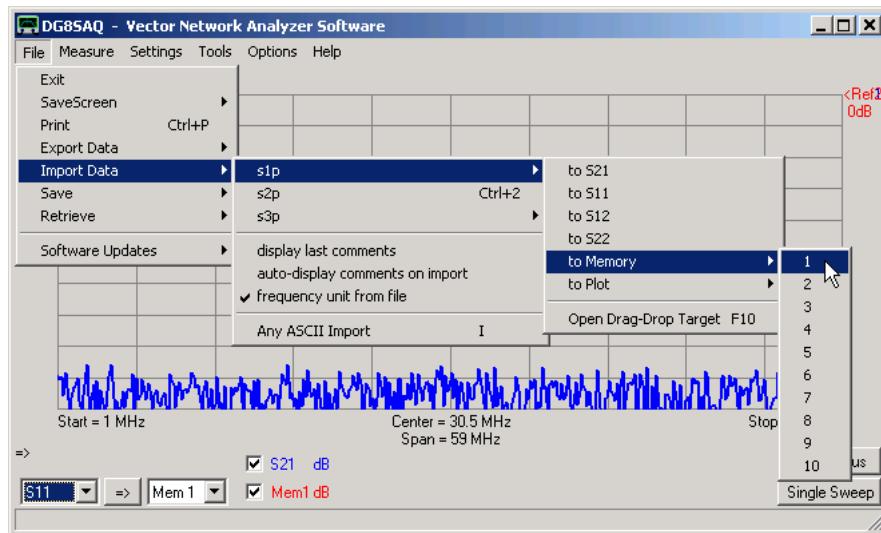
	A	B	C	D	E	F	G
1	f/Hz	S21:dB	S11:Real Z	S11:Imag Z	S11:VSWR		
2	2,50E+04	-1,12E+02	1,30E+00	1,53E-01	3,85E+01		
3	1,50E+05	-8,31E+01	1,31E+00	8,98E-01	3,81E+01		
4	2,76E+05	-7,77E+01	1,33E+00	1,64E+00	3,77E+01		
5	4,01E+05	-7,20E+01	1,35E+00	2,39E+00	3,72E+01		
6	5,26E+05	-6,78E+01	1,37E+00	3,15E+00	3,67E+01		
7	6,51E+05	-6,34E+01	1,39E+00	3,91E+00	3,61E+01		
8	7,77E+05	-6,00E+01	1,43E+00	4,70E+00	3,54E+01		
9	9,02E+05	-5,65E+01	1,46E+00	5,51E+00	3,46E+01		
10	1,03E+06	-5,34E+01	1,50E+00	6,33E+00	3,38E+01		
11	1,15E+06	-5,06E+01	1,55E+00	7,19E+00	3,29E+01		
12	1,28E+06	-4,78E+01	1,61E+00	8,08E+00	3,20E+01		
13	1,40E+06	-4,53E+01	1,67E+00	9,01E+00	3,09E+01		

FILE - IMPORT DATA

Import Data

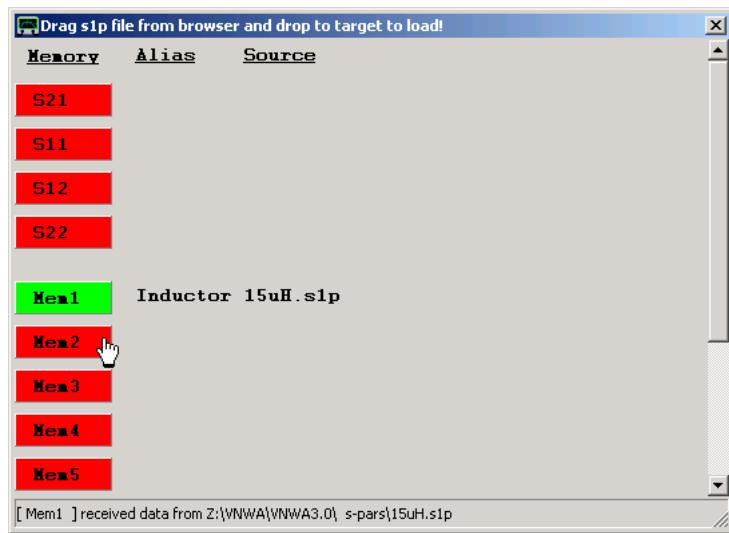
VNWA can import certain S-parameter files in **Touchstone** format (s1p, s2p, s3p), binary format (v1b, v2b) and csv-format. To do so, select the main menu "**File**" - "**Import Data**" -

►►► **Note:** Importable files must contain S-parameters normalized to 50 Ohms.



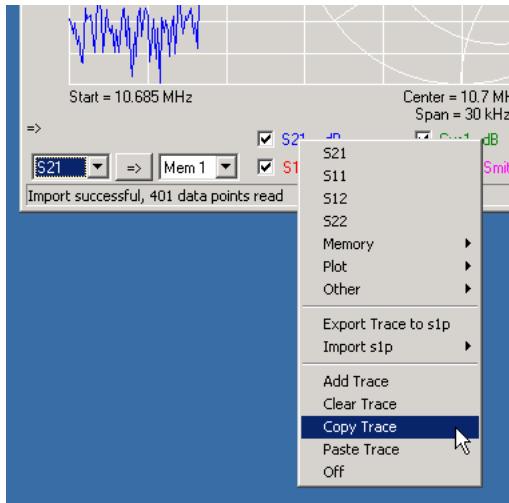
►►► **Note:** VNWA normally also imports the frequency units from an S*P-file. If you want to keep the pre-selected frequency units, un-check the "frequency init from file" menu item.

►►► **Hint:** Menu "Open Drag-Drop Target" will open the drop target form. This menu can also be invoked via the keyboard shortcut F10:

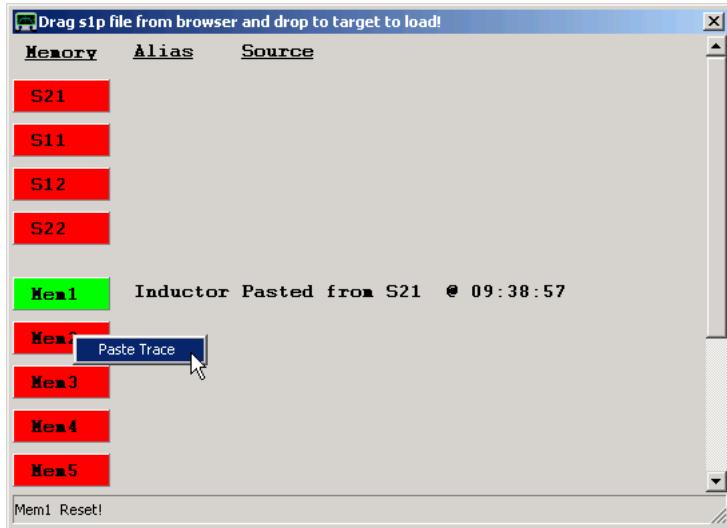


S1P-files can be loaded by dragging them with the mouse from a file browser to the red drop targets on the form and dropping them there. When loaded, the drop target will turn green and the Source column will display the name of the file that was loaded. The Alias column displays the alias assigned in the custom trace form to the respective memory space. Clicking onto a green target will turn it red again and delete the Source info. It will NOT delete the contents of the memory space, though.

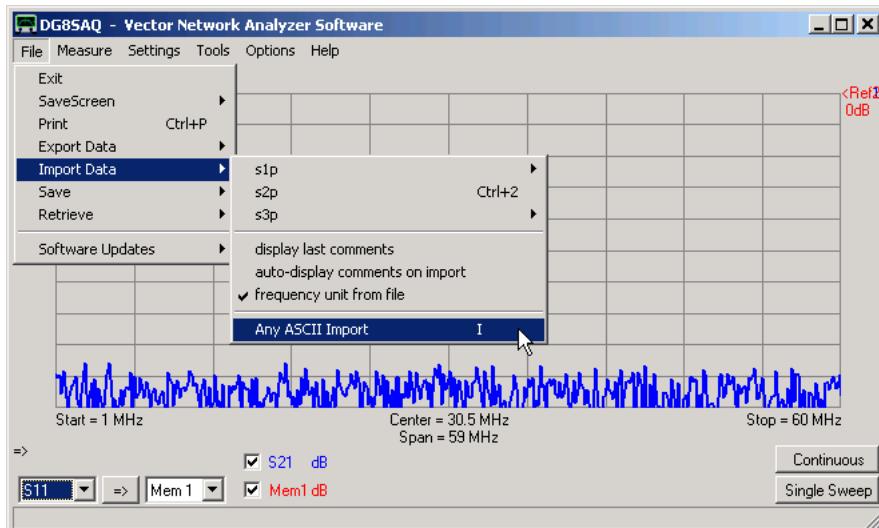
Note, that data can also be copied and pasted from memory space to memory space by invoking the right-click menus "**Copy Trace**" and "**Paste Trace**".



The copied data can also be pasted into the drop target form by right-clicking the respective red drop target:



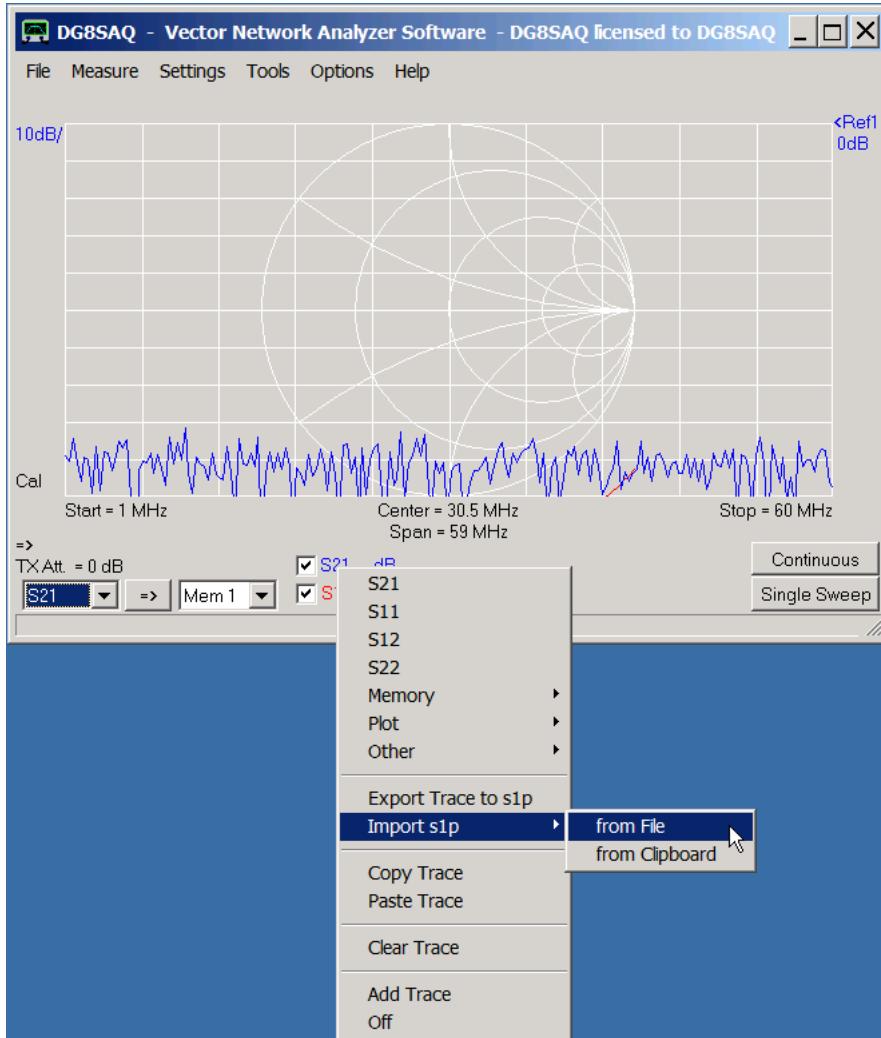
Arbitrary ASCII Data Import:



Menu "Any ASCII Import" will open an ASCII import tool.

Additional Data Import Functionalities:

►►► **Hint:** You can directly import trace data into a specific memory space by right-clicking the trace label (e.g. S11 in below example) and selecting "Import s1p". Thus, the s1p, v1b or csv file will be loaded into the S11 memory space.



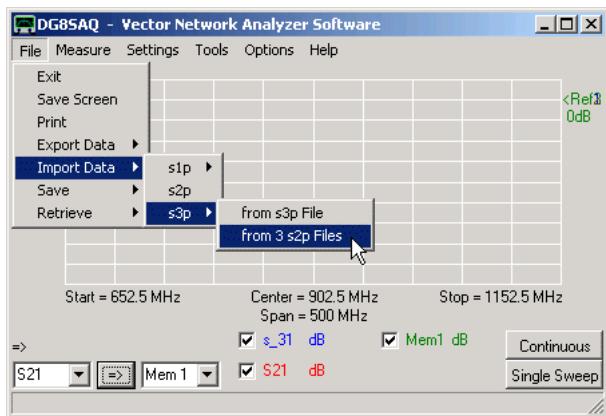
►►► **Hint:** You can also import S-parameters by dragging s*p, v*b or csv-files from any file browser and dropping it onto the VNWA application.

►►► **Hint:** VNWA will accept s*p, v*b or csv file names as runtime parameter. A second numeric parameter specifies the data space where an s1p record is stored (1:S21, 2: S11...). Thus, if you associate e.g. the file ending s2p with VNWA.exe, then you can start VNWA and load an s2p file by double-clicking onto the file.

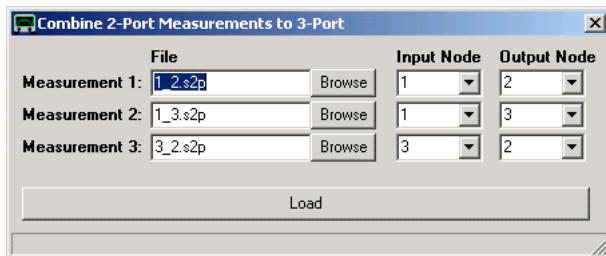
►►► **Hint:** You can also import 3-port data from three 2-port measurements with the third port terminated by 50 Ohms.

FILE - IMPORT - DATA - S3P - FROM 3 S2P FILES

It is possible to automatically combine 3 2-port measurements or data files to 3-port data. Note, that the unused port must be terminated with 50 Ohms in order to obtain valid 3-port S-parameters.
To do so, select the menu highlighted below:



In the popping up window, you may specify 3 filenames and the corresponding DUT port numbers.



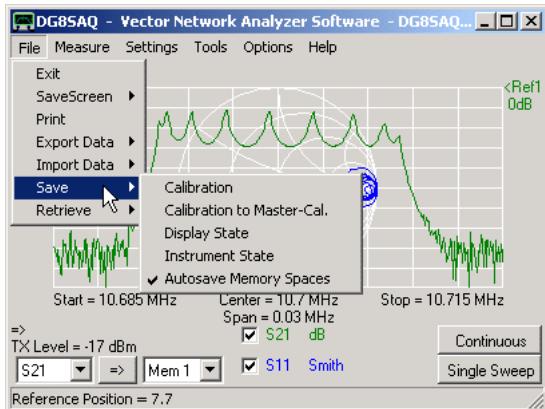
Pressing the **load button** will load the 3-port buffer.

Note, that in above example S33 from the second file import will be overwritten by S33 from the third file import.

FILE - SAVE

Save:

VNWA allows to save various software states:



Save Calibration

You can save a calibration to a *.cal file. You are asked for a file name and an optional comment to be stored in the calibration file.

If the instrument is not calibrated, the command will be ignored.

Save Calibration to Master-Cal.

You can save a calibration to a *.cal file. At the same time the file is loaded as master calibration.

Save Display State

You can save the display state, i.e. the color scheme, displayed kinds of traces, grids and units into a file. You are asked for a file name. If you enter a file extension, it will be ignored, as the extension of the backup file is predefined.

Save Instrument State

You can save the complete instrument state including display state, hardware setup, calibration, multiplier table... into a set of files or a single zipped instrument state file (*.zis-file). You are asked for an optional comment and for a file name. All backup files will start with the name specified by you. Predefined extensions will be added automatically.

In case of the single zipped file the file name matches the archived file names. Therefore do not rename this zis-file, otherwise it can no longer be read. Use the instrument state manager to unzip, zip, rename or delete instrument states. The instrument state manager is launched via menu File - Retrieve - Instrument State.

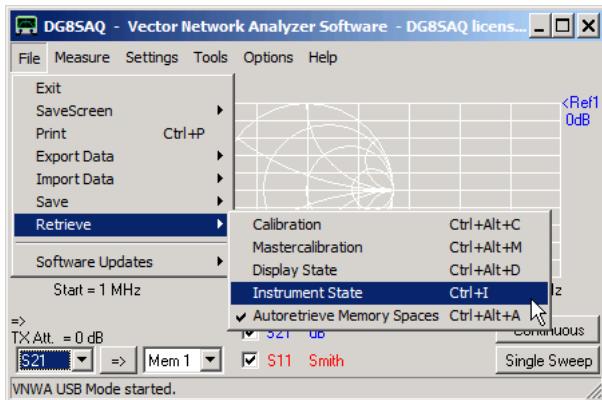
Note: The master calibration will NOT be saved!

Autoretrieve Memory Spaces

If this option is checked, VNWA will automatically write data from all memory spaces into the file MemorySpaces.dat at program termination. This allow to reload the displayable traces at a later VNWA session.

FILE - RETRIEVE

VNWA allows to save and retrieve various software states:



Retrieve Calibration

You can retrieve a previously saved calibration from a *.cal file. When invoking this command, the calibration file manager opens to let you select the *.cal file. When loading the cal file, the frequency range will be changed to that of the calibration file. Also, the calibration standard settings are saved in the cal file and will be restored upon load.

►►► **Warning:** If the number of data points or the frequency span of the calibration differs from that of your displayed data, **all data in the display buffers will be lost.**

►►► **Note:** The master calibration will NOT be overwritten!

Retrieve Mastercalibration

You can retrieve a previously saved calibration from a *.cal file and load it as mastercalibration. When invoking this command, a file browser opens to select the *.cal file. The frequency range will remain unchanged.

Retrieve Display State

You can retrieve a previously saved display state, i.e. the color scheme, displayed kinds of traces, grids and units from a backup file. A file browser will open to assist you selecting the backup.

Retrieve Instrument State

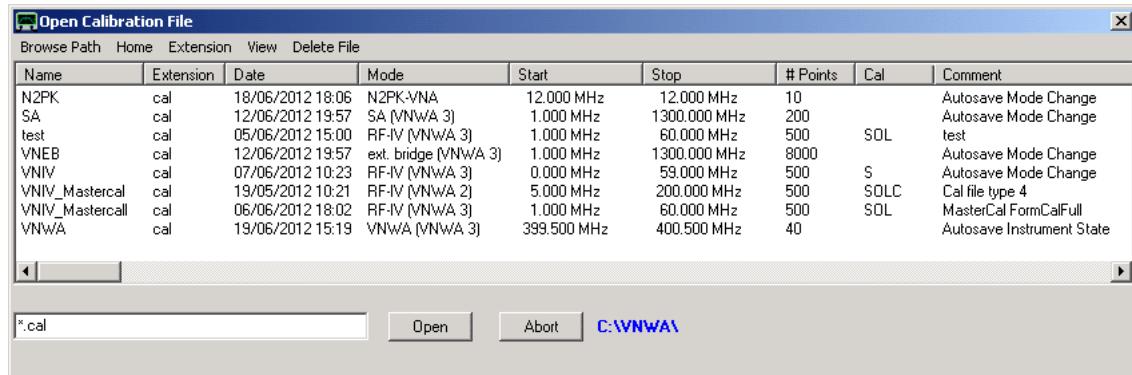
You can retrieve a previously saved complete instrument state including display state, hardware setup, calibration, multiplier table... from a set of backup files or from a single zipped instrument state file (*.zis). Selection of this menu will launch the instrument state manager, which allows to browse, select, rename, zip/unzip and delete instrument states.

Autoretrieve Memory Spaces

If this option is checked, VNWA looks at program start for the file MemorySpaces.dat and loads all memory spaces from this file if available. This will reload the displayable traces from the last VNWA session.

FILE - RETRIEVE - CALIBRATION: THE CALIBRATION FILE MANAGER

Upon invoking the main window menu **File-Retrieve-Calibration** the calibration file manager opens to let you select a calibration file:



The manager lists all calibration files in the selected directory and displays their main properties.

Column "Cal." entries:

- S =Short
- O =Open
- L =Load
- C =Low Loss Capacitor
- T =Thru

Column "Comment" lists the comments which the user has entered when the cal file was saved. Automatically generated cal files show automatically generated comments (e.g. autosave...).

Main menu commands:

Browse Path: lets you select the directory to browse. The current working directory is shown by the blue label on the bottom.

Home: invokes a jump into the VNWA program folder.

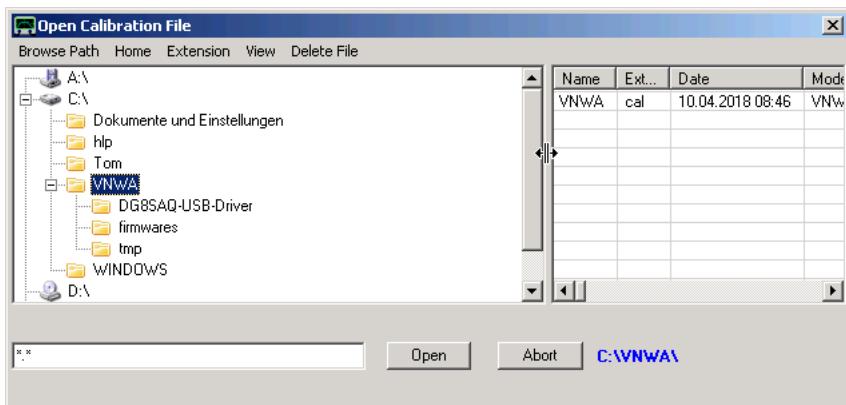
Extension: lets you filter file extensions. Cal files may also have the extension *.bak. Note, that the manager will only show calibration files regardless of their file name and extension.

View: allows to customize the displayed file list by e.g. filtering calibrations with appropriate VNWA mode.

Delete File: deletes the marked file. This functionality is also achieved by right-clicking the file name.

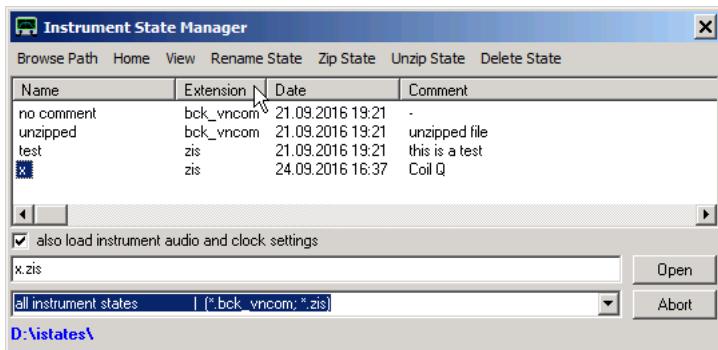
Note:

- The wildcard * can be used in the file input field to apply a file list filter.
- Clicking a file name will copy the name to the name input field.
- Double-clicking a file will open it.
- The file name input field also accepts file paths.
- From software version 36.7.5 on the cal file manager includes a **directory tree** for directory selection. The tree may be expanded by clicking the "browse path" menu, by double-clicking the path label or by dragging the right tree boundary to the right with the mouse. A folder is being selected by clicking it in the tree.



FILE - RETRIEVE - INSTRUMENT STATE: THE INSTRUMENT STATE MANAGER

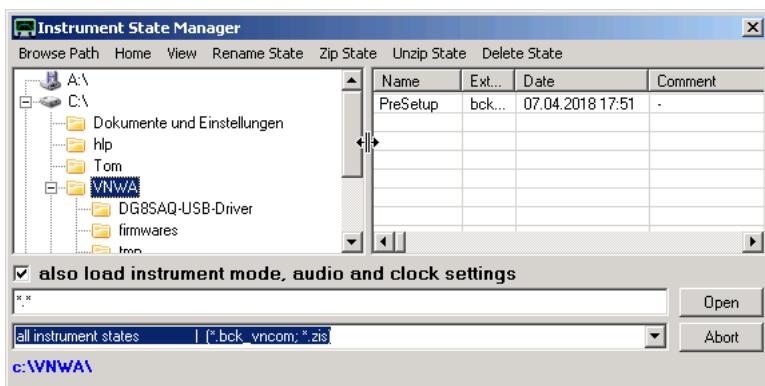
When trying to retrieve an instrument state, the instrument state manager is invoked:



It lists all instrument states (zipped and/or unzipped depending on the filter settings) found in the selected path.

The **path** can be changed by

- selecting the menu **Browse Path**.
- clicking the **blue path label** on the bottom.
- manually entering a path into the **file name field** next to the Open button and pressing the return key.
- From software version 36.7.5 on the cal file manager includes a **directory tree** for directory selection. The tree may be expanded by clicking the "browse path" menu, by double-clicking the path label or by dragging the right tree boundary to the right with the mouse. A folder is being selected by clicking it in the tree:



Note, that the list of instrument states can be sorted by clicking on the list headers or filtered by entering a file name with wildcards, e.g. z*

A **file is selected** by

- clicking onto its filename in the list.

A selected file can be **opened**, **renamed**, **zipped** (if the state is unzipped) **unzipped** (if the state is zipped), **renamed** or **deleted**.

These function can be accessed via the main menu or by right-clicking the file to be operated on.

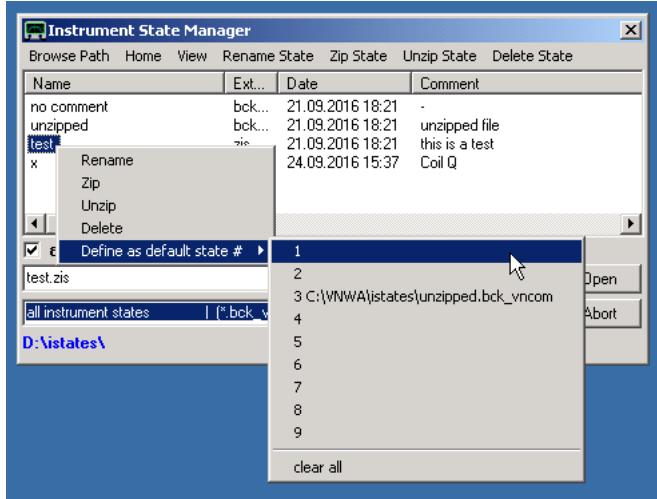
A **file can be opened** by

- selecting it in the list and pressing the **open button**.
- **double-clicking** the file name in the list.
- **manually entering** the file name in the file name field next to the open button and pressing the return key.

►►► **Note:** There are two loading modes: You can load the full instrument state including audio and clock settings or ignore these. The loading mode is selected by the check box below the file list. If you load instrument states from a foreign computer, it is advised NOT to load the full instrument state. If the full instrument state is loaded, then you likely have to do an audio setup and check your clock settings before taking a new measurement.

►►► **Note:** A zipped instrument state (*.zis) may also be loaded by simply dragging it from a Windows file manager and **dropping** it onto the VNWA main window.

▶▶▶ **Note:** Up to 9 instrument states can be defined as **default states** by right-clicking the state and selecting the default state number it is to be associated to:

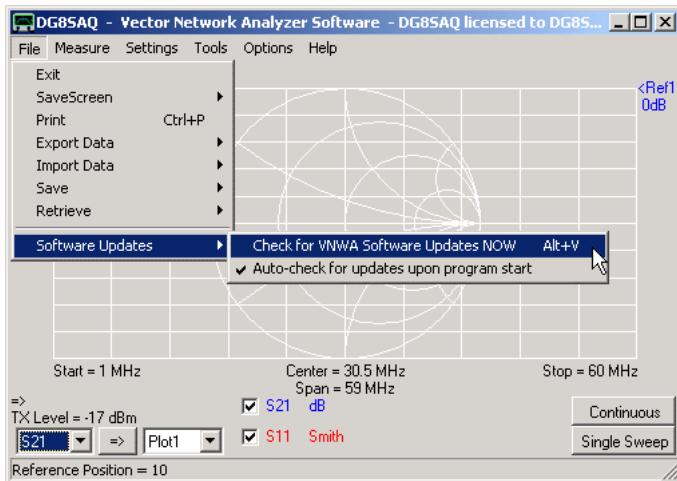


In above example, default state number 3 is associated to instrument state "unzipped" and we are about to associate default state number 1 to the instrument state "test".

Default states can be loaded via keyboard short-cuts **alt+state number**, e.g. with above association the instrument state "unzipped" can be loaded by simply pressing the keyboard combination **alt+3** in the VNWA main window without invoking the instrument state manager, also see page "Keyboard Shortcuts". This is useful when frequently switching between instrument states.

FILE - SOFTWARE UPDATES

Starting VNWA software version 36.2 it is possible to initiate a software update from within the VNWA software.



The menu item "**Software Updates**" allows to manually or automatically check for VNWA software updates over the internet.

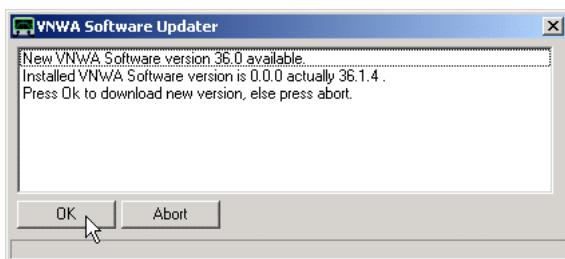
If "**Auto-check for updates upon program start**" is checked, then the VNWA software will consult the web for software updates every time the software is launched manually.

If the software is remote controlled via script file or via Windows messages, web access is suppressed, so the remotely launched application won't hang and wait for user interaction.

If VNWA finds an available update newer than the installed software, the user is being notified:

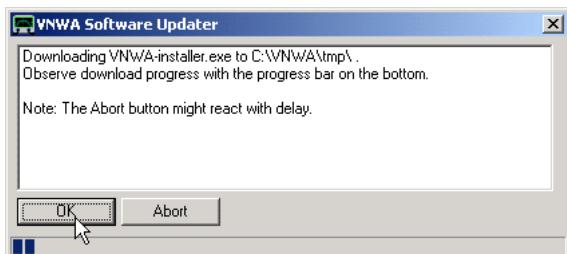


If you accept downloading the new version by pressing OK, then the **VNWA Software Updater** will be started:



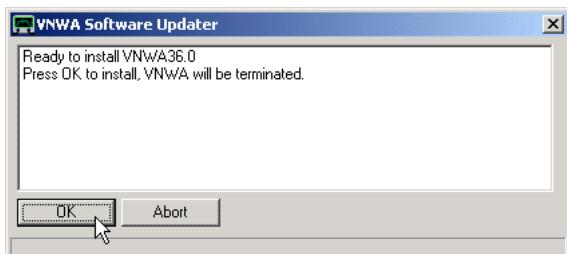
►►► **Note:** The VNWA Software Updater can also be launched manually by the main menu item "**File - Software Updates - Check for VNWA Software Updates NOW**".

The updater displays the new available and the currently installed software version and asks for confirmation to download. Pressing OK will actually start the download:



▶▶▶ **Note:** The download process can be interrupted by pressing Abort, but only after downloading has actually started as can be seen by the blue progress bar.

Once, the file is downloaded, the user is asked for confirmation in order to install of the new software version:

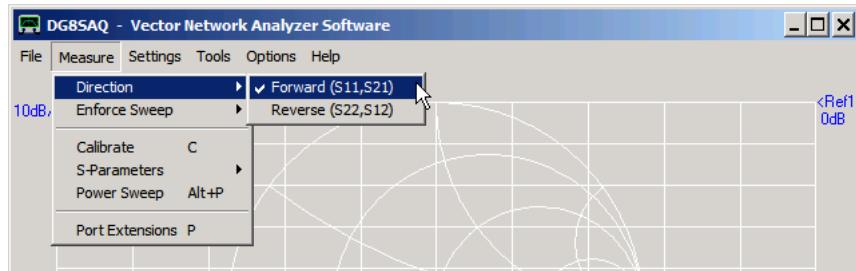


Pressing OK will terminate the VNWA software and launch the just downloaded aoutomatic installer.
Abort will close the software updater and return to the VNWA main window. The just downloaded installer will not be deleted and can later be launched manually from a file browser.

MEASURE

The VNWA main menu "**Measure**" offers the following functions:

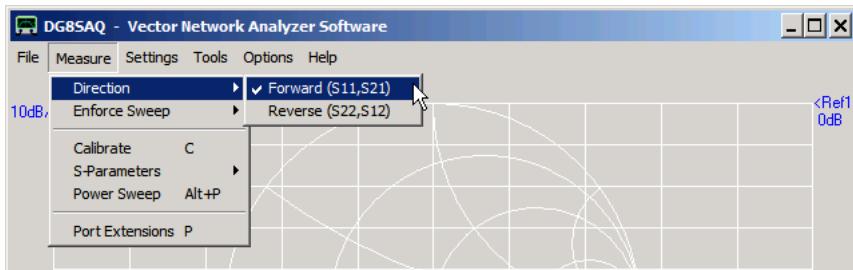
- Direction**
- Enforce Sweep**
- Calibrate**
- S-Parameters**
- Power Sweep**
- Port Extensions**



MEASURE - DIRECTION

Direction:

Use this menu item to select the measurement **direction forward or reverse**.



►►► **Hint:** The measurement direction can also be altered by clicking on the arrow label => on the lower left side of the VNWA main window:



►►► **Note:** Without usage of an **S-parameter test set** the direction menu only determines, in which data spaces the measured data is stored. The function of the Tx port and Rx port remains unchanged. This means, **the VNWA2.x cannot automatically alter the signal flow direction.**

If you want to measure S12 and S22 in this case, you need to **manually turn the DUT around, e.g. exchange input and output terminals of the DUT.**

►►► **Note:** The VNWA software supports usage of an S-parameter test set, which actually allows for automatic change of signal flow direction. If activated, the direction menu will actually alter the signal flow direction.

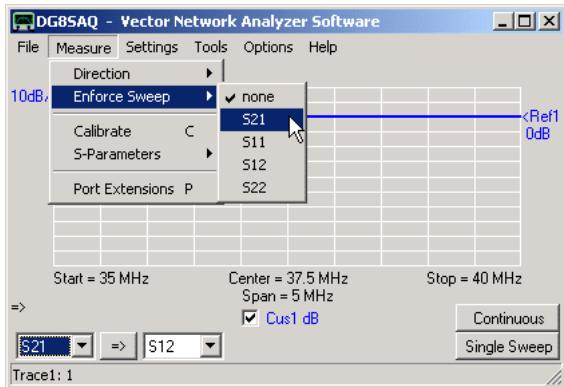
►►► **Note:** The direction menu will be disabled in the presence of an S-parameter test set **with automatic direction switching** being activated.

MEASURE-ENFORCE SWEEP

Depending on what traces are being displayed, the VNWA software determines what to measure. I.e. if only S21 is being displayed, VNWA only measures S21 under normal conditions.

When only a custom trace is being displayed, VNWA will report "nothing to sweep", even though the custom trace may access e.g. S21.

In this case sweeping of S21 may be enforced by selecting the main menu "Measure-Enforce Sweep"...



... and checking S21.

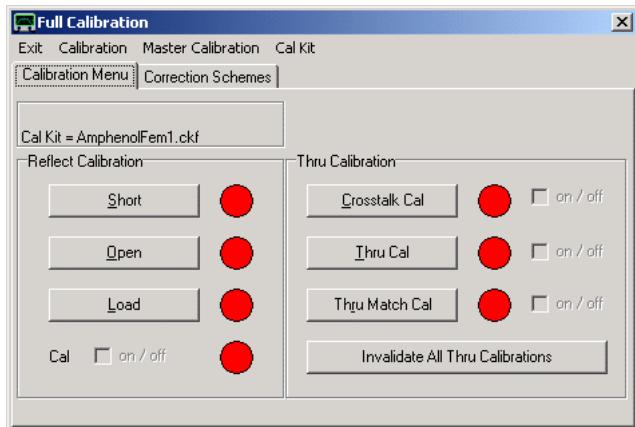
Selecting "None" will clear all checked items and switch the software back to automatic determination of what is to be measured.

Note, that these settings are being ignored in RF-IV mode.

MEASURE - CALIBRATE

Calibrate:

The "Calibrate" menu allows to calibrate the VNWA using calibration standards.



From VNWA version 36.3.7.9 this window contains a new submenu:

Exit	exits the full calibration menu
Calibration-	
Load	Loads a calibration
Save as	Saves current calibration under a user definable name
Invalidate	Invalidates the complete calibration, all lamps turn red.
Master Calibration-	
Load Mastercal	Loads a master-calibration
Save Cal as Mastercal	Saves current calibration as master calibration
Off	Turns the current master calibration off without deleting the master calibration file. The master calibration will be reloaded on the next program start.
Delete	Deactivates the current master calibration and deletes the master calibration file, so it will never be reloaded.
Cal Kit-	
Load	Loads cal kit settings from a file
Save as	Saves current cal kit settings to a file
Info	Shows currently used cal kit, e.g.



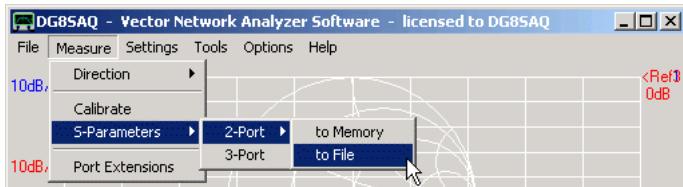
►►► **Note:** Cal kit info is also stored in the calibration file. Thus loading a calibration will also rewrite calibration kit settings.

A detailed calibration guide can be found on page Instrument Calibration.

A discussion on available error correction schemes can be found on page Error correction models.

MEASURE - S-PARAMETERS

This menu allows you to **measure S-parameters of a 2-port or a 3-port device**, e.g. of a crystal filter (2-port) or a balun (3-port).

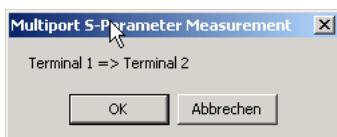


2-Port

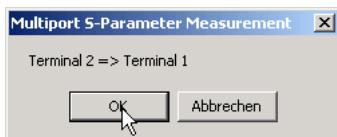
You can select, if you want to store the acquired data in the internal s2p-buffer (=memory) or additionally store it in a file on your hard disk.

►►► **Note:** Acquired or loaded 2-port S-parameters will be stored in the VNWA data spaces (S21,S11,S12,S22) and **additionally in an internal s2p buffer**. This is necessary for recalculating S-parameters to new varying normalization impedances, e.g. like in the matching tool.

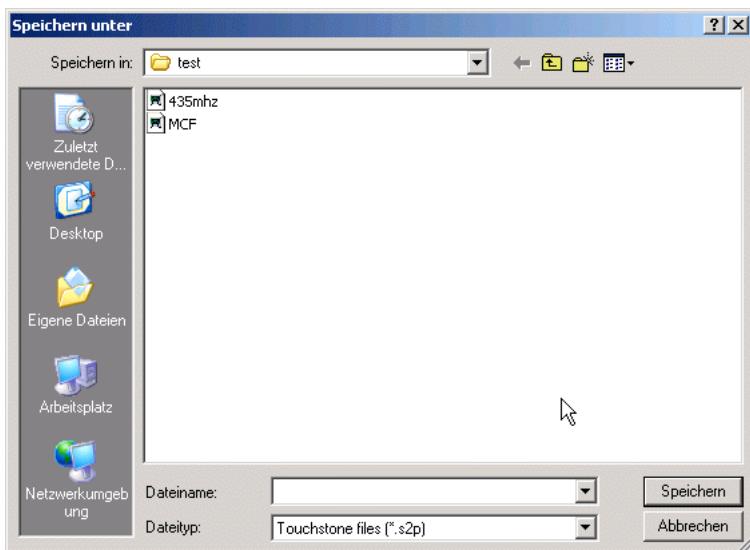
If you invoke the measurement, you need to confirm by pressing the **ok**-button, that you are ready to measure.



After the forward measurement (S11,S21) has finished, you need to manually turn the DUT and confirm with the **ok**-button again, that you are ready for the reverse measurement.



If you have selected to measure "**to File**" then you are asked to select a file name:

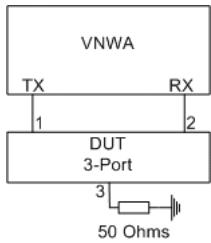


The data is saved in Touchstone s2p format.

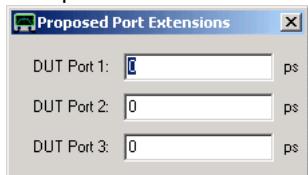
►►► **Note:** If you have decided to measure "to Memory", you can still save your measured data manually.

3-Port

3-port measurements basically work like 2-port measurements, except that **you must terminate the unused port with a 50 Ohms resistor**:



►►► **Note:** If port extensions are activated upon measurement start, you may specify 3 different port extensions for the 3 DUT ports:



►►► **Note:** Measurement data will be collected into an internal s3p buffer (=memory containing s11,s12,s13,...,s33). There is a distinction between e.g. **S11** (2-port data) and s11 (3-port data). These are stored in different memory spaces. This is necessary when analyzing 3-ports by reducing them to a 2-port with a balanced port by means of the 3-port analyzer tool.

►►► **Note:** You can save your measured data manually to a 3-port Touchstone *.s3p-file .

MEASURE - POWER SWEEP

The VNWA's TX output power can be controlled in a very defined way over a range of 50dB with a resolution up to 0.01dB by means of an internal digital attenuator.

The **Power Sweep** utility allows to measure a test device's response to changing input power instead of changing frequency on a user defined attenuation grid.

This kind of measurement is useful to determine the compression point of an amplifier or mixer. Note, that the VNWA's maximum output power of -17dBm is too low to drive most test objects into compression.

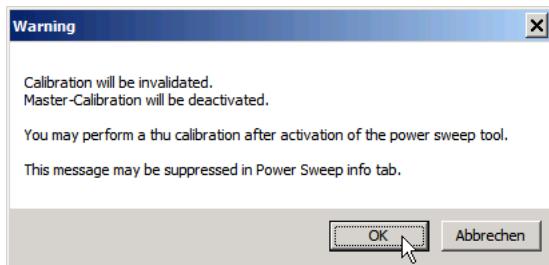
Thus, a booster amplifier is needed to bring the VNWA TX output power to a suitable level.

It is recommended to place a band pass filter between the VNWA TX output and the booster amplifier in order to only amplify the desired signal as the VNWA delivers a wide frequency spectrum.

The VNWA performs multiple zero frequency span sweeps, each one at a constant predefined power level. An analysis function can be specified to e.g. extract the average sweep dB-value of the RX input power level.

The result is plotted against the TX output power level.

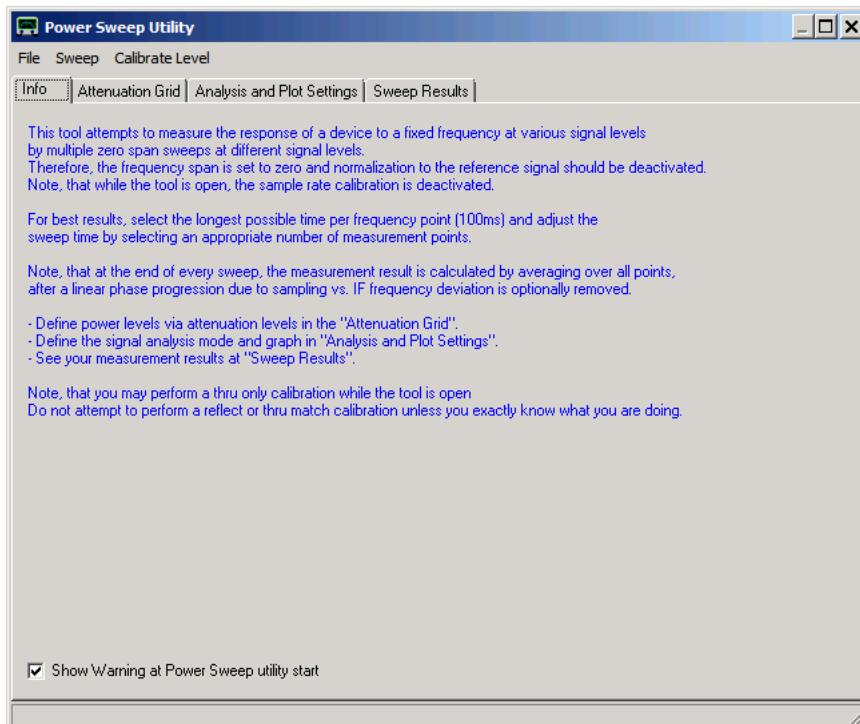
When launching the utility, a warning is issued as long as the warning window is not suppressed:



The power sweep utility will measure the received signal power in arbitrary units and thus will switch off any calibration which may interfere with detecting pure RX input signal levels.

If a specific signal level should be used as reference level, then a Thru calibration may be performed at this level while the power sweep utility window is open.

The utility's **Info Tab** offers further instructions:



Note, that the warning window can be suppressed by unchecking the box at the bottom of above window.

On the **Attenuation Grid** tab the measurement signal levels are being specified via attenuation values:

#	Target Attenuation/dB	Actual Attenuation/dB
1	0	0
2	5	5
3	10	10
4	13	12.999
5	17.5	17.499
6	20	20.002
7	30	30.001
8	40	39.991
9	50	49.968
10	100	inf
11	insert line	
12	delete line	
13	auto-fill list	
14	clear list	
15	save list	
16	load list	
18		
19		
20		
21		
22		
23		
...		

Sweeps done!

The target attenuation column can be edited. A value of 0dB denotes the maximum output power. Positive attenuation values decrease the output power, negative values are illegal.

As the attenuator hardware has a level dependent finite resolution the nominal actual attenuation value is calculated and displayed on the right column. Note, that the resolution gets poorer at higher attenuation values.

The actual attenuation values are used for further analysis and NOT the target values. The Excel-like input sheet offers a right-click menu as can be seen above. It allows to insert or delete lines, to fill the list with a set of default values (auto-fill), to erase all values (clear), to save the list to a file or load a list from a file. The file commands may also be accessed via the main menu **File**.

The **Analysis and Plot Settings** tab allows to specify measurement data analysis and plotting:

Analysis function = db(S21)

Do not normalize to reference channel (must be checked to see signal level changes)

Remove phase progression prior to averaging (should be used when above option is checked)

Analysis dll = browse and load

dll channel = 1

Plot Settings

<input checked="" type="checkbox"/> draw grid	Captions	Guideline
<input checked="" type="checkbox"/> draw data points	Re: Level/dB	<input type="checkbox"/> draw guide line to Ref() grid
<input checked="" type="checkbox"/> draw data lines	Im: none	Ref(x) = 17.0448261671+1*x
<input type="checkbox"/> Mem 1	Real Color: <input type="color"/>	Imag. Color: <input type="color"/>
<input type="checkbox"/> Mem 2	Real Color: <input type="color"/>	Imag. Color: <input type="color"/>
<input type="checkbox"/> Mem 3	Real Color: <input type="color"/>	Imag. Color: <input type="color"/>
<input type="checkbox"/> Mem 4	Real Color: <input type="color"/>	Imag. Color: <input type="color"/>

Compiler messages
no Errors

VNWA set to maximum output level.

The **Analysis function** field allows to specify a user defined function for extracting the data of interest from the measurement data. The syntax is identical to the custom trace syntax.

In above example, dB(S21) denotes the calculation of the relative level in dB from the measured S21 data. The analysis result will be averaged over the full zero span frequency sweep.

The analysis function is evaluated with complex calculus and may thus also contain an imaginary part. If "**Complex Result**" is activated, the prepared plot will contain a second y-axis for the imaginary part. The result of above dB-function will be solely real, though.

Note, that the expression `att_db` can be used in the analysis function to access the current actual attenuation value from the attenuation grid.

In order to measure RX input levels with changing TX output levels, normalization must be switched off with the according check-box. As the changing TX level will also change the internal reference signal accordingly, all changes would cancel out if normalization would not be deactivated.

On the other hand, switching normalization off means that absolute phases can no longer be measured. The measured RX input signal will experience a phase progression over the zero frequency span sweep points. Mathematically removing this phase progression will improve the level detection accuracy and sensitivity. This is the purpose of the "**Remove phase progression**" check-box.

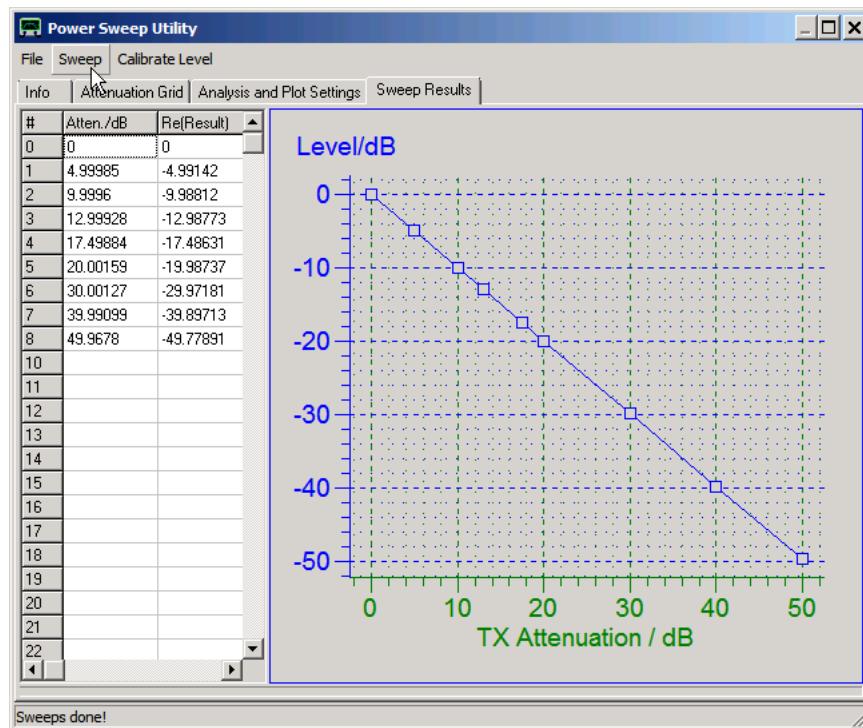
Instead of using the VNWA RX to detect signal levels, any external measurement system that can be read out by the PC can be attached via a dll software interface. This so called **analysis dll** can be loaded with the "**browse and load**" button. This facility may be useful to calibrate external power sensors. A software template is shown on the next page "Measure - Power Sweep: Analysis DLL". A "**dll channel**" number is being communicated to the analysis dll. This allows to access multiple sensors with one and the same dll.

The "**Plot Settings**" section allows to configure the plot on the sweeps result tab. The drawing of a grid, data points and data lines connecting the data points can be switched on or off separately. Captions for the left (**Real**) and right (**Imaginary**) y-axes can be freely assigned.

The "**guide line**" is a custom definable function to be plotted on top of the data points and curve. Again, the syntax is identical to that of the custom traces . Here, the guide line was used to draw a straight line automatically obtained by linear regression to the data points, see below.

Measured data sets can be stored to one of four memory spaces (**Mem1 ... Mem4**) and plotted in custom defined colors. Note, that these memory spaces are independent of the VNWA main Mem fields. Also, the saved data survives program termination and relaunch. Note, that the measured data (but not the data saved in the Mem fields) is erased, when the target grid attenuation values or the analysis function are being edited.

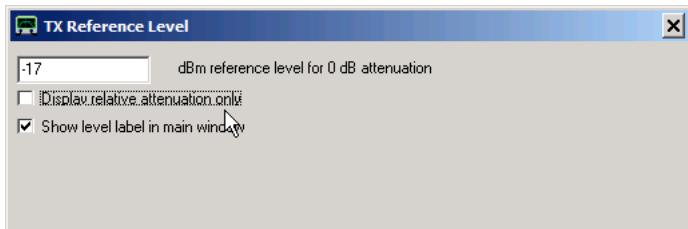
After invoking a power sweep by selecting main menu item **Sweep**, the measured data points and the plot are being found on the "Sweep Results" tab:



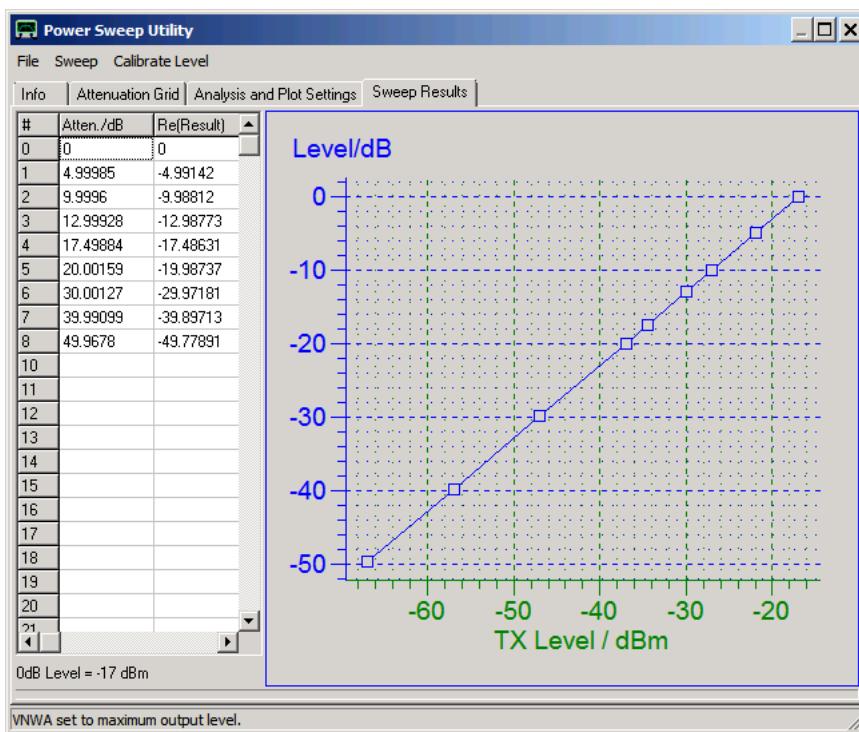
The above measurement was performed on a coax cable connecting the VNWA TX port with the RX port. Prior to the above measurement, a thru calibration had been performed at 0dB attenuation. Therefore, a level of 0dB corresponds

with the 0dB attenuation value. Clearly, the measured RX level (blue) corresponds exactly with the TX level (green, here in terms of TX attenuation) as should be for a coax cable. The good correspondence can also be observed with the numbers in the result table.

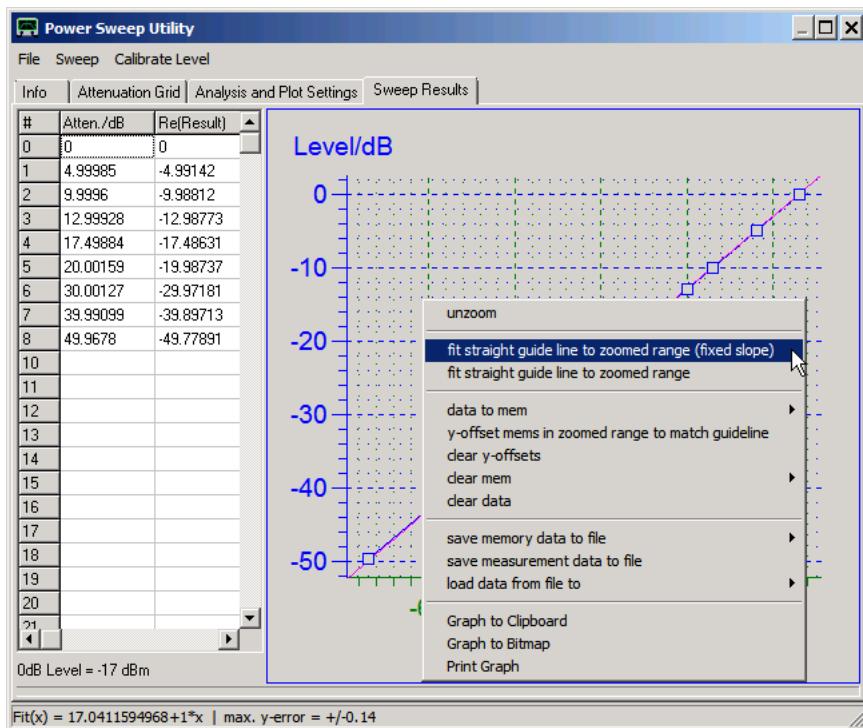
Clicking the main menu "Calibrate Level" allows to specify the absolute output level of the TX port (or the level after a booster amplifier):



If "Display relative attenuation only" is deactivated, the absolute level is used on the x-axis instead of the attenuation value:

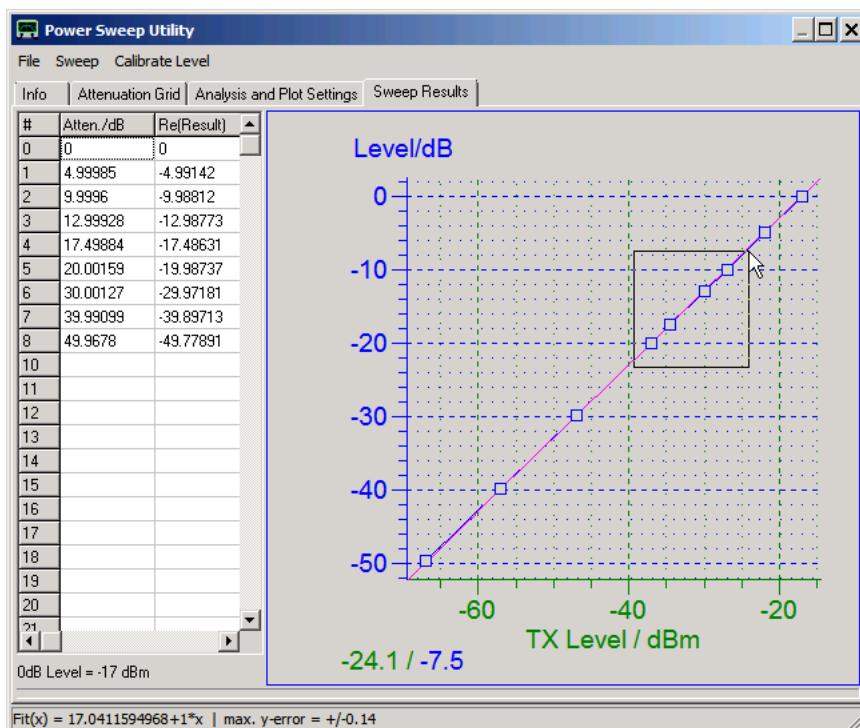


The plot offers a right-click menu offering access to several functions:

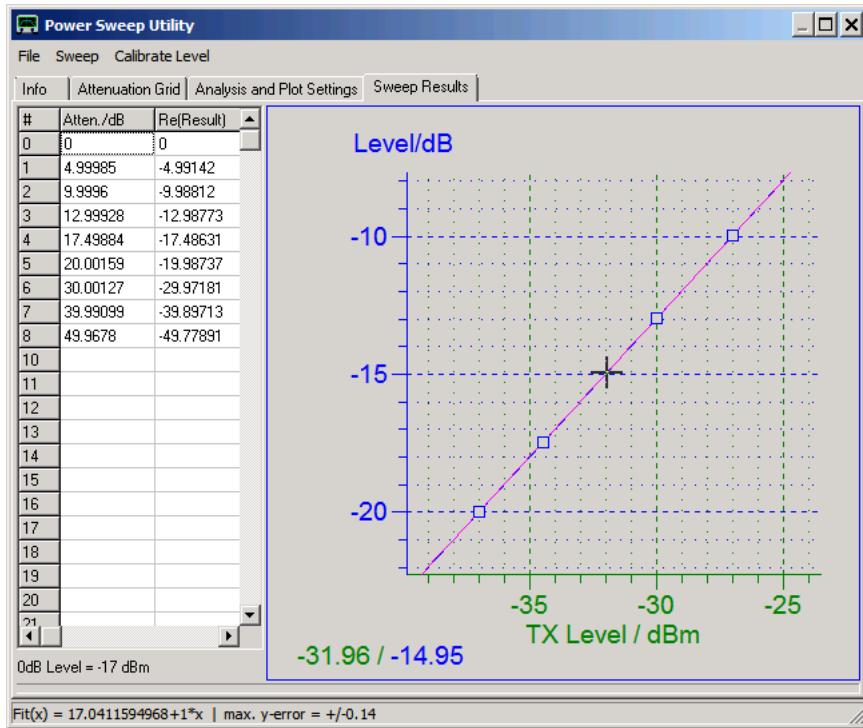


- unzoom after box zoom (see below)
- Fitting a straight line through the data points either with fixed slope (+1 or -1 chosen automatically) or with variable slope. The fit result will be found in the guide line input field in the analysis and plot settings tab. Only visible points inside the zoomed range will be used for the fit. Note the pink fitted guide line in above plot. Also note the fit result in the status line below the table and graph.
- Measured data can be saved to mem fields, mem fields or measurement data can be erased or mem traces can be automatically shifted vertically, so they best match the guideline. The latter is useful to compare compression of test objects with different amplification values.
- Data can be saved to files.
- The graph can be saved or printed.

A **box zoom** is performed by holding down the ctrl-key while drawing the box with left mouse key pressed:



The left mouse key must be released before the ctrl-key. Then a zoomed version of the graph is being shown:



Note, that pressing the left mouse key without pressing the ctrl-key will produce a **cursor** cross. Note the cursor position values on the lower left below the plot.

MEASURE - POWER SWEEP: ANALYSIS DLL

The following Delphi/Pascal code defines a simple example dll with name Powersweep_Example_DLL.dll:

```
library Powersweep_Example_DLL;

uses
  Windows,
  SysUtils,
  Classes,
  Forms,
  ShellAPI;

$R *.res

procedure _Init; export; cdecl;
begin
$IFDEF DEBUG_CON
FreeConsole;
AllocConsole;
writeln('SwitchDLL function "Init" called.');
$ENDIF
end;

procedure _Close; export; cdecl;
begin
$IFDEF DEBUG_CON
FreeConsole;
$ENDIF
end;

procedure _Process(parameter: integer; var real, imag: double); export; cdecl;
begin
$IFDEF DEBUG_CON
writeln('SwitchDLL function "Process" called.');
writeln('Parameter=',parameter);
$ENDIF
real:=4;
imag:=5;
end;

exports _Init;
exports _Close;
exports _Process;

begin
end.
```

►►► **Note:** The leading underscores in above procedure declarations (e.g. `_Init` instead of `Init`) are required for C-language compatibility.

The procedure **Init** initializes a console window and writes a short message to it. It is called only once when the dll is being loaded.

The procedure **Process** writes the passed data to the console and passes the simulated measurement results *real* and *imag* back to the VNWA software. It is called for every power measurement.

The procedure **Close** releases the console again. It is called once when the dll is unloaded.

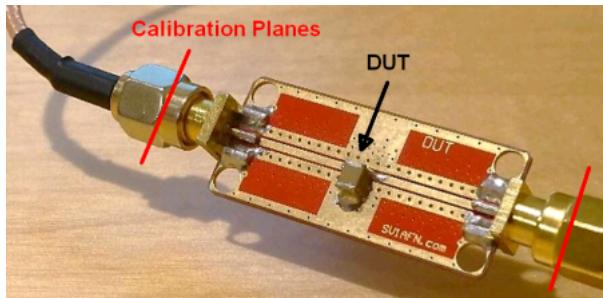
MEASURE - PORT EXTENSIONS

Here, you can add **delays** to your measured data.

From VNWA software version 36.7.9.7 on this menu also allows for more sophisticated ways to manipulate measurement data, i.e. **embed or de-embed networks**.

Port extensions vs. Embedding / Deembedding

Very often, the VNWA calibration cannot be performed at the exact position of the DUT to be analyzed. Consider the DUT being a surface mount component sitting on a pcb with microstrips connecting to SMA pcb edge connectors:



If the microstrips on the pcb are well matched to 50 Ohms and if their loss is small, their main effect will be to introduce a signal delay and thus a phase shift. In this case, the calibration planes may be shifted from the connectors to the DUT position by using port extensions with positive delay values. In this sense, the microstrips are effectively de-embedded from the measurement result by the port extensions. The measurement result looks like if the microstrips weren't there. But this correction is only approximate, e.g. the loss introduced by the microstrips is not taken into account.

Now, consider the more extreme case of an antenna sitting on a very high antenna mast:



Normally it is not feasible to measure on the mast top, so one can only measure into the cable end at the bottom of the mast. The cable may have a considerable length and thus a considerable loss and it may not even be 50 Ohms. In this case, removing the cable effect by a mere phase shift through a port extension will not yield a satisfactory result. If the cable properties are known, then the full mathematical power of de-embedding can be used to "subtract" the cable influence from the measurement to yield the data of the antenna alone. This is demonstrated in below example "de-embedding a transmission line".

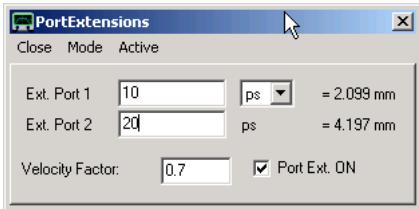
De-embedding means mathematically removing some circuitry from a chain of circuits, e.g. the cable from an antenna with cable. Also the opposite can be done, which is called embedding. Consider measuring a high impedance bandpass filter with the 50 Ohms VNWA. The measurement result will show poor match and a high passband ripple. Actually, you are interested how the filter behaves in a high impedance environment.

You could find this out by using the VNWA matching tool, but not in real time. Alternatively, you can mathematically embed your measurement in between two impedance transformers and thus get the result updated in real time while you sweep. This will be demonstrated in below example "Embedding Impedance Transformers"".

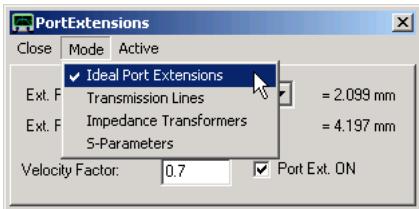
The most general approach is to define two distinct sets of 2-port S-parameters (=S-matrices) for an input and output network, that can be used to embed the measurement in between or to be de-embedded (=removed) from a measurement. This is demonstrated below in section "Using S-Parameters for Embedding/De-Embedding".

Using Port Extensions

The port extensions menu can be opened via main menu "Measure-Port Extensions" or via keyboard shortcut P.



Make sure that you have selected **Ideal Port Extensions** mode:

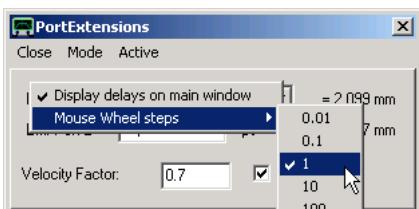


►►► **Example:** You perform a calibration at the end of your test cable. But between your test cable and you DUT, there might be another low loss 50 Ohms transmission line (e.g. a stripline on the test PCB). If you want to know the reflection coefficient of your DUT without the additional transmission line, you can simply enter the signal delay caused by the additional line into the "**Ext. Port 1**" field and check the box "**Ext. on**". The delay of the transmission line can easily be found by creating a temporary short circuit at the DUT and tune "Ext. Port 1" such that the measured reflection coefficient shows up at the short circuit point in the Smith chart.

►►► **Note:** Entered delays are **one way delays**. Reflection data will be corrected by 2x the delay, as reflected signals travel through the delaying transmission line twice (forth and back!). Transmission data will only be corrected by 1x the delay as a transmitted signal will pass the delaying transmission line only once.

►►► **Note:** **Positive delays move the calibration planes further away from the VNWA**, negative delays move them closer to the VNWA.

►►► **Hint:** You can also tune the delays with the **mouse-wheel**. Use the right-click menu to select the **mouse wheel steps**:

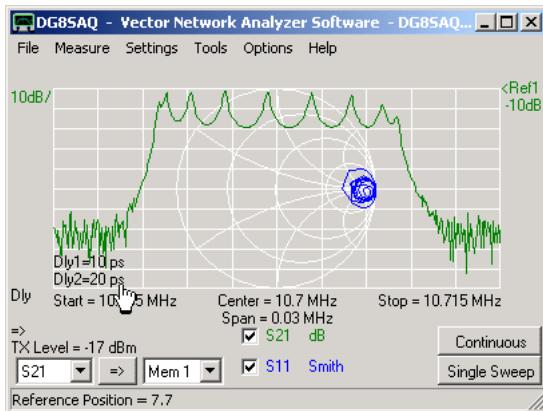


►►► **Note:** Delays for forward and reverse measurements can be chosen independently.

►►► **Note:** The mechanical lengths of the delays are being displayed for info. The velocity factor is only used to calculate the mechanical lengths of the delays. It has no influence on the measured data. Note, that a velocity factor of 0.7 is appropriate for coaxial adapters, while for coaxial cables the velocity factor is typically lower.

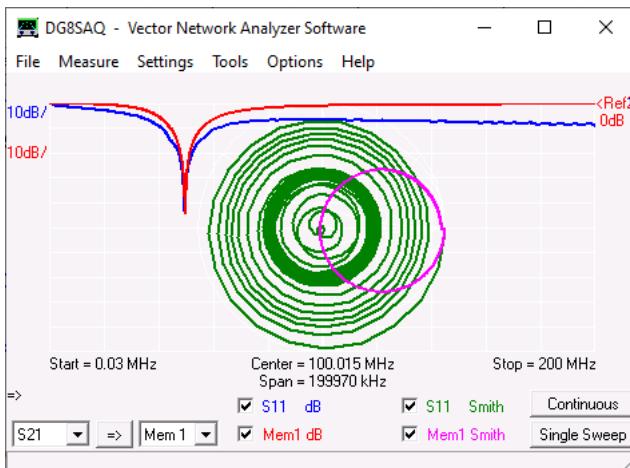
►►► **Note:** All delays also influence the phases of the transmission measurements. The effects of port extensions can only be monitored by looking at phase data or by monitoring data in the Smith chart.

►►► **Hint:** Right-clicking the port extension window will allow to activate an info text label on the main screen displaying the port 1 and 2 delays:



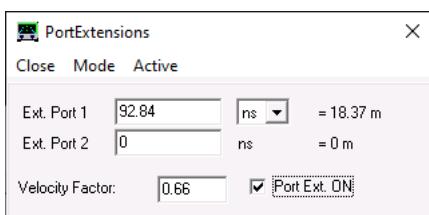
De-Embedding a transmission line

The following screenshot shows a reflect measurement (S11) of an antenna (dummy, R+L+C series circuit) connected through 18.37m of real RG58C/U coaxial cable to the VNWA TX port, which is also the position of the calibration/reference plane (blue, green):

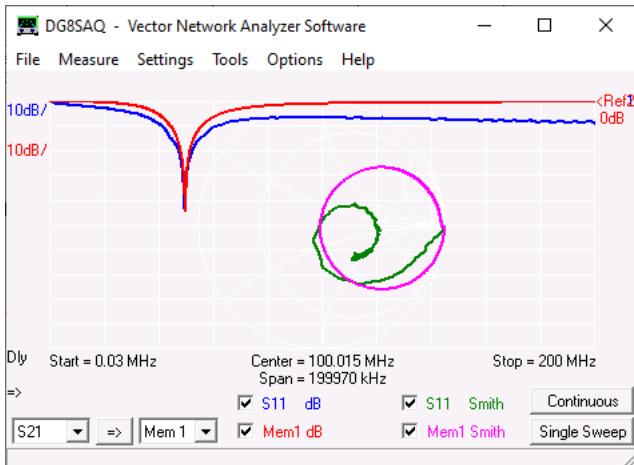


For comparison, the antenna dummy has also been measured without the cable and the result stored to Mem1 (red, pink). The effects of the cable delay and loss are clearly visible.

Now, let's try to remove the effect of the cable by using a simple port extension. We set the velocity factor to that of the cable (=0,66) and tune the port 1 delay for the real physical length of 18.37m:

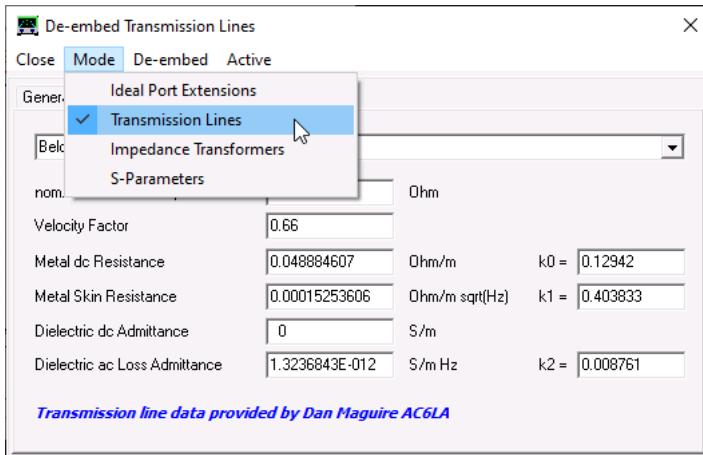


After switching port extensions ON, the Smith chart diagram changes:



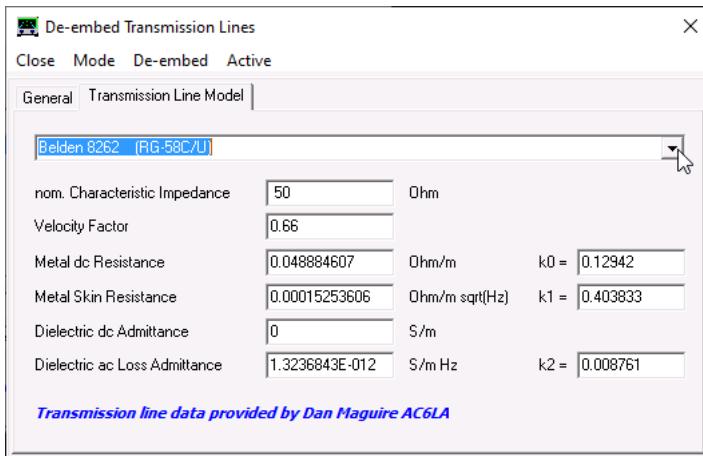
The result is not really convincing, as neither the cable loss nor the cable dispersion is taken into account by this correction.

Now, let's switch the port extensions menu to transmission line mode:

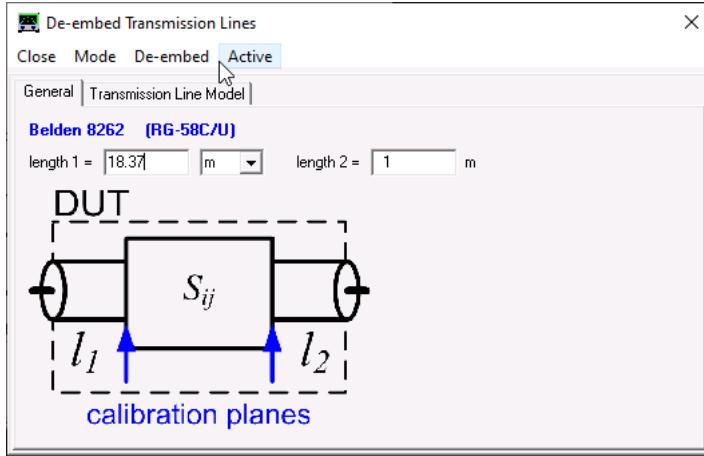


Note, that the menu window title changes when the mode setting is changed.

On the "Transmission Line Model" tab we select our cable type RG58C/U. The cable parameters will be fetched automatically from a data base kindly provided by Dan Maguire:



On the "General" tab enter the physical length of 18.37m of our cable and make sure **De-embedding** is selected and is **Active**:

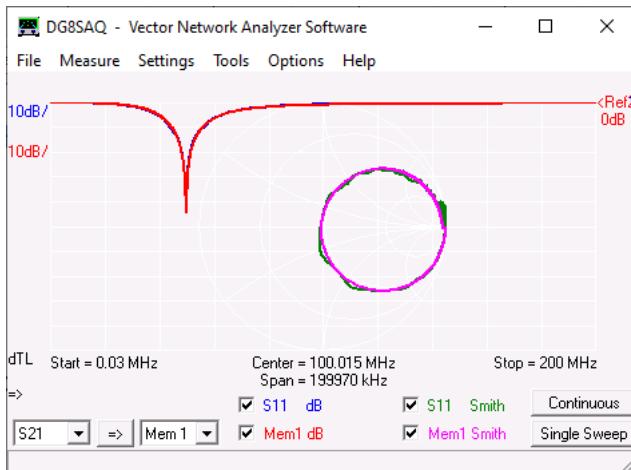


Since we only have a one-port measurement, the output length (2) setting is irrelevant.

Note that the drawing in above window changes when toggling between embedding and de-embedding.

DUT denotes what is physically there, in our case a cable + antenna. The blue calibration plane arrows show the positions where we want the planes to be, i.e. we only want to see the antenna, but not the cable.

With above settings the de-embedding result looks fully satisfactory:

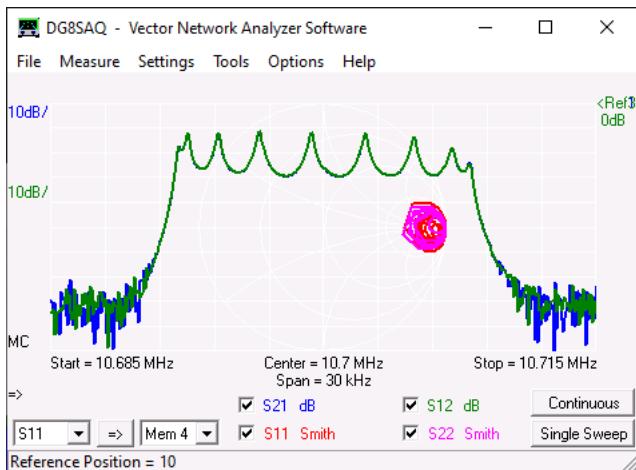


This is even more surprising as the RG58C/U cable used in this experiment was not manufactured by Belden and I did not tune any parameters.

Note the dTL label bottom left in above screenshot. This stands for "de-embed Transmission Line". Depending on the selected mode, this label changes.

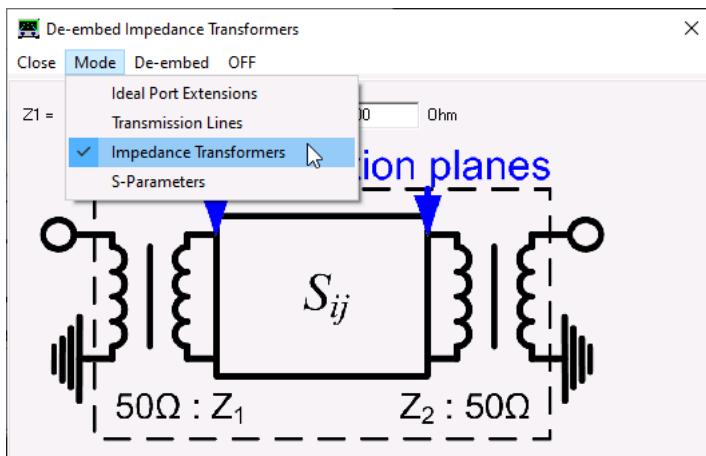
Embedding Impedance Transformers

Next, we look at a full 2-port measurement of an unmatched high impedance monolithic crystal filter (MCF):

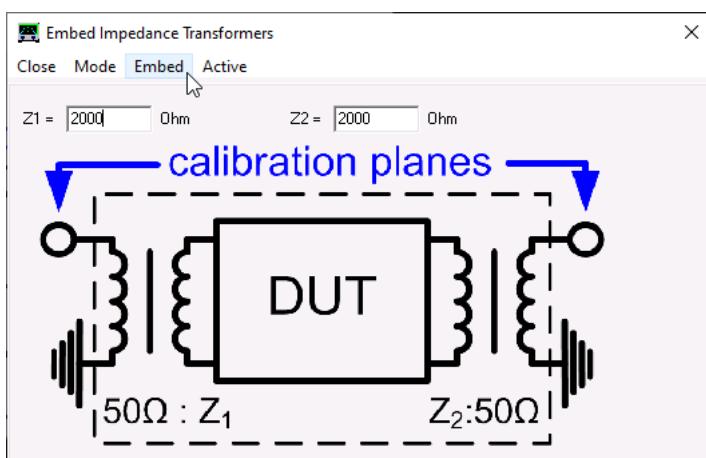


These kinds of filters are designed to operate in a very high impedance environment (here about 2kOhms) and thus show a huge passband ripple when measured in a 50 Ohm environment. Thus, we want to embed the measurement it between two ideal impedance transformers.

We switch the port extensions mode to "Impedance Transformers":

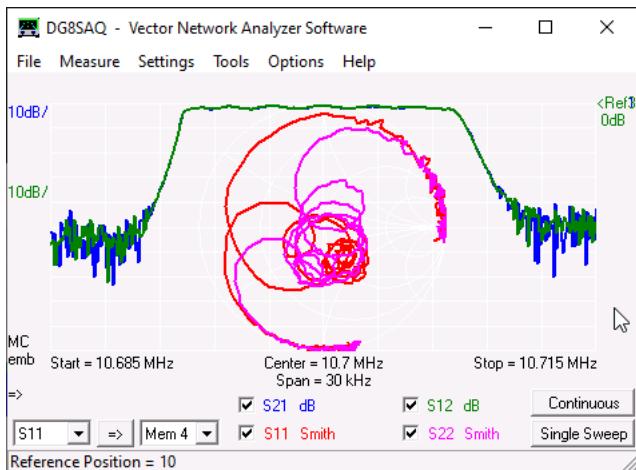


We enter the desired input and output impedances of 2000 Ohms, switch to **Embed** and to **Active**:



Note, that the blue arrows in above diagram now indicate that the virtual transformers are included in our measurement data.

And indeed, now we see the filter with a flat passband as if it was measured with a 2000 Ohm VNWA:



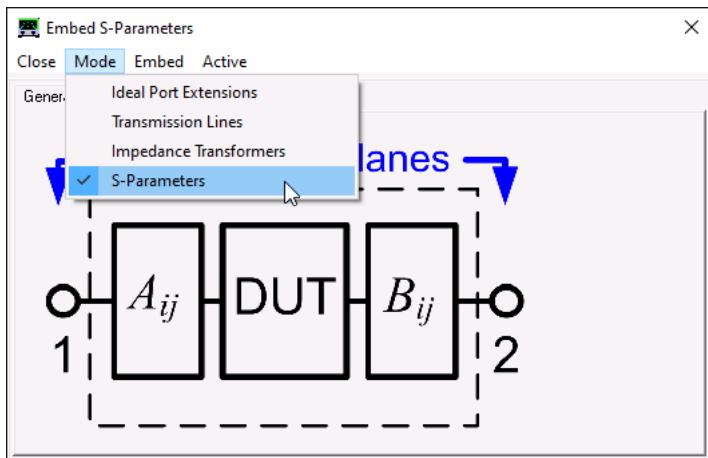
Note, that 2-port embedding/de-embedding requires valid full 2-port S-parameters. These can either be produced by loading an s2p-file or by doing a full 2-port measurement with a valid full 2-port calibration.

When using an automatic S-parameter testset and a full 2-port calibration, these corrections will be computed in real-time while continuously sweeping.

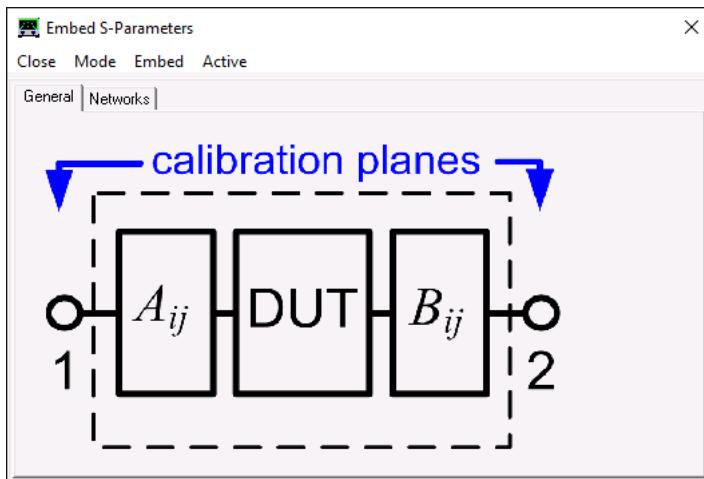
If the software detects that the data is no valid full 2-port data (either not all S-parameters measured or no valid 2-port calibration present), approximate 1-port corrections will be used as fallback. But note, that especially transmission data can become very wrong in this case.

Using S-Parameters for Embedding/De-Embedding

Embedding/de-embedding also can be performed with arbitrary 2-ports defined by mathematical expressions. To do so, select mode "**S-Parameters**".



Example 1



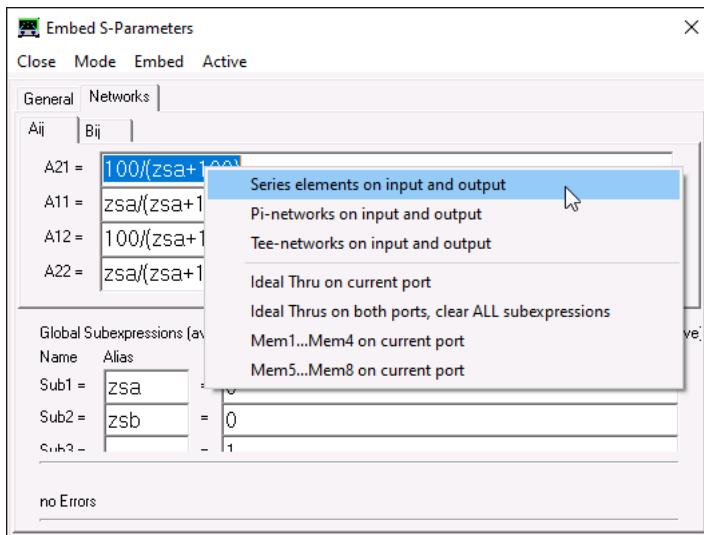
In tab Networks you can specify the input S-matrix A and the output S-matrix B to be embedded or de-embedded by means of math expressions. The syntax is identical to VNWA custom traces. To simplify matrix expressions, a set of 10 sub-expressions can be used, which are common to both A and B matrices. Matrix expressions may also use mem1...mem10 memory fields thus enabling to embed or de-embed measured data stored in the mem* data fields.

Name	Alias	Expression
Sub1		= 1
Sub2		= 1
Sub3		- 1

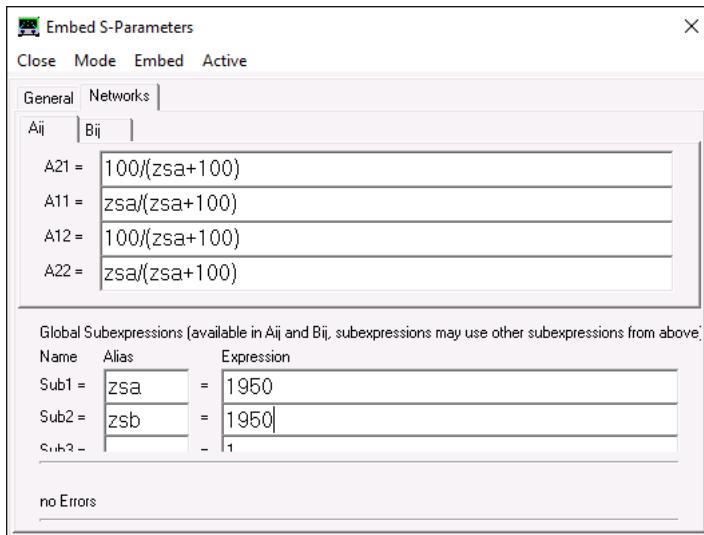
no Errors

Some special case math expressions already have already been hard-coded into the software. Select one of them by right-clicking one of the matrix fields.

Say, we want to see how the MCF from the previous section behaves if placed in between two series resistors of 1950 Ohms thus also providing a 2000 Ohm impedance level at the filter terminals. We select "Series elements on input and output":

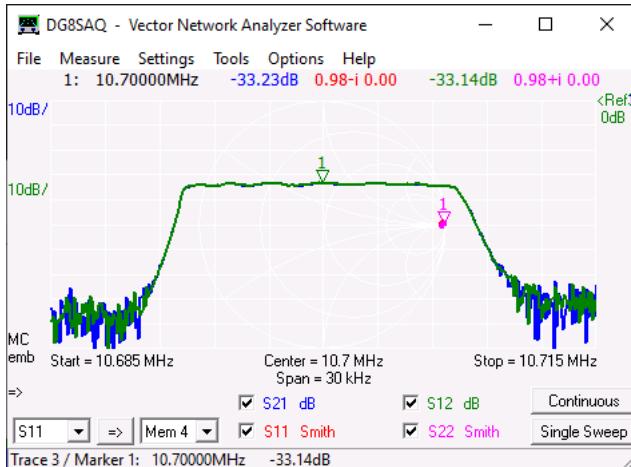


Next, we specify the series resistors zsa and zsb to be 1950 Ohms:



►►► **Note:** We could also specify complex impedances instead of resistors if we wanted, e.g. we would use the expression $j*w*1e-9$ for a 1nH series inductor.

Now, we activate embedding and obtain a flat passband again as expected:

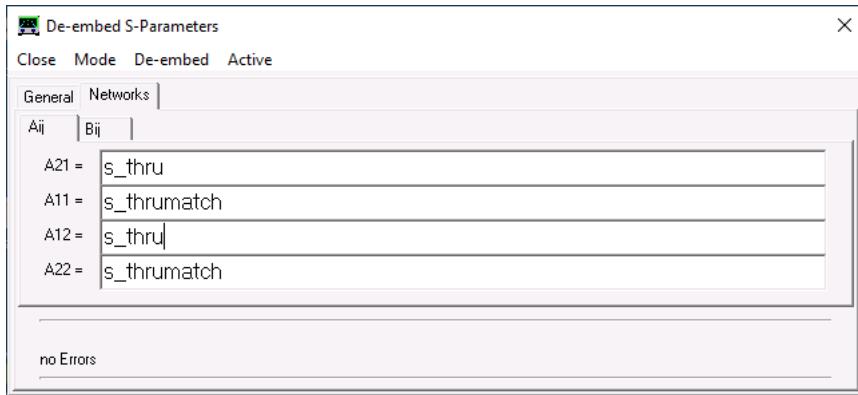


Note, that the insertion loss has dramatically increased, though. We now also see the loss introduced by the 1950 Ohms

resistors. Moreover, S11 and S22 show a near open circuit, because we basically see the 1950 Ohms resistors in series with the high filter input and output impedances.

Example 2

Another useful application has to do with calibration. Say, you have a female calibration kit consisting of female S,O,L and T-standards. So, after a SOL-calibration you have a calibrated **male** test port for reflect measurements. What to do, if you need a calibrated **female** test port? Well, simply use the female Thru standard to convert the male port to a female one. But then, you must de-embed the S-matrix of the Thru standard from the measurement result to move the calibration plane from the VNWA side of the Thru barrel to the DUT side, or in other words, to remove (=de-embed) the effect of the thru barrel from the measurement data. The S-matrix of the Thru barrel is known to VNWA as it is stored in the cal-kit model description. You can simply reference it by expressions s_thru (S12=S21) and s_thrumatch (S11=S22). Thus, the de-embedding settings for this would look like this:



Note, that the same calculation has been performed by means of custom traces in chapter "Magi-Cal as Permanent Calibrator". The new embedding/de-embedding facility greatly simplifies this job, though.

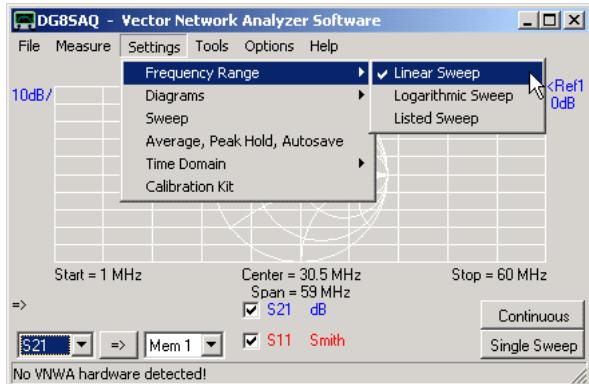
SETTINGS

The VNWA main menu "**Settings**" offers the following functions:

- Frequency Range**
- Diagrams**
- Sweep (VNWA)**
- Sweep (N2PK-VNA)**
- Power Levels**
- Average, Peak Hold, Autosave**
- Time Domain**
- Calibration Kit**



SETTINGS - FREQUENCY RANGE



This menu item allows to set the sweep mode (linear, logarithmic, listed sweep) and the sweep frequency range.

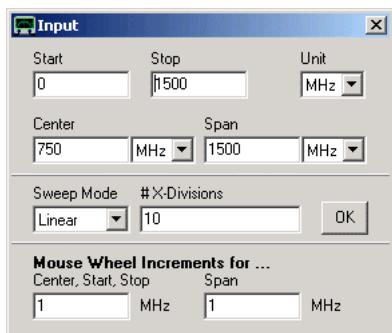
Selecting "Linear Sweep" or "Logarithmic Sweep" will change the sweep mode accordingly and will activate the **frequency "Input"** window.

Selecting "Listed Sweep" will select the listed sweep mode and open the frequency list editor.

The frequency "Input" window:

The frequency "Input" window can be activated in various ways:

- via main menu "Settings"- "Frequency Range" (linear or logarithmic sweep mode)
- by double-clicking on one of the frequency labels in the main window (linear or logarithmic sweep mode)
- by pressing the keyboard shortcut key "f" (linear or logarithmic sweep mode)



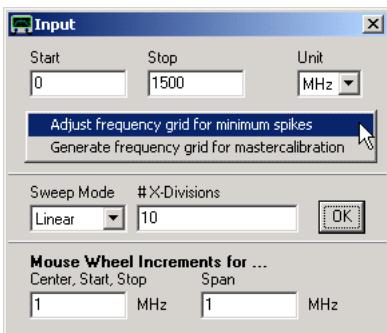
Enter a start and stop frequency or alternatively a center frequency and a frequency span.

Also, the number of vertical grid lines and the sweep mode (linear, logarithmic, listed sweep) can be changed here.

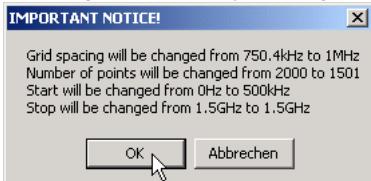
►► **Hint:** Note, that the start, stop, center frequency and the frequency span can also be changed with the **mouse wheel** by holding the mouse pointer over the corresponding main window frequency label below the grid and turning the mouse wheel. The increment values can be set here in the frequency "Input" window.

Avoiding interference spikes

Note, that while VNWA2 has quite an odd clock frequency of 35.9 MHz, the VNWA3 clock frequency is exactly 12 MHz. Since very sensitive measurements will reveal some degree of undesired interference at multiples of the clock frequency, it might be desirable to modify the frequency grid such that these frequencies are being maximally avoided. This can be done automatically for the linear sweep mode by right-clicking the frequency input window anywhere:

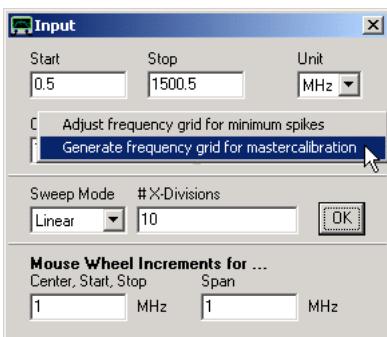


Pressing ok will modify the original frequency grid:



Generating an optimum frequency grid for a mastercalibration

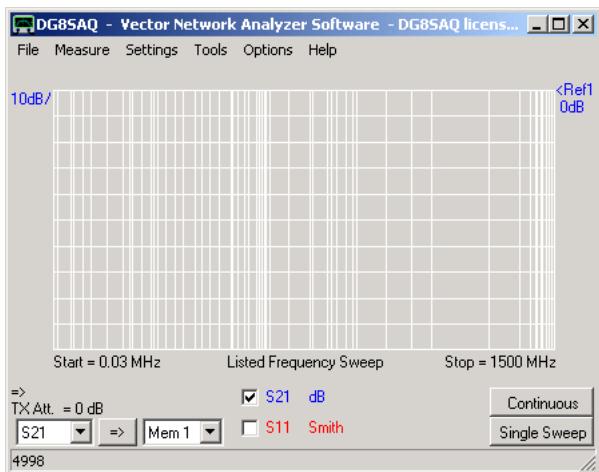
When generating a mastercalibration with auto-clock multipliers in use, special care has to be taken to properly include the frequencies, where the multipliers change, in the frequency grid. Thus, ideally a listed sweep grid is needed. An appropriate frequency list with the chosen start and stop frequency and approximately the chosen number of points can automatically be generated with the menu item "**Generate frequency grid for mastercalibration**", which is opened by right-clicking the frequency input window anywhere:



A notification of the used parameters follows:



After pressing ok, the sweep mode is changed to "Freq. List". After closing the frequency input window, the frequency list window opens to show the frequency segments. After closing it, the odd but optimized frequency grid will be shown and used in the main window:



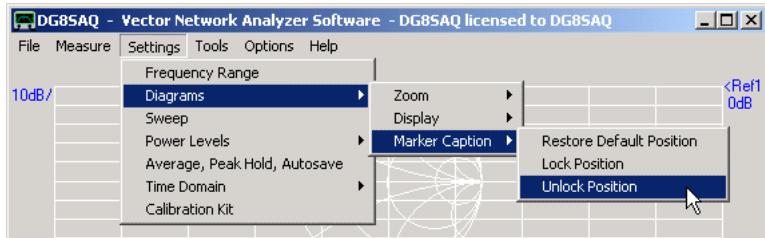
SETTINGS - DIAGRAMS

The VNWA main menu "**Settings - Diagrams**" offers the following functions:

Zoom

Display

Marker Caption

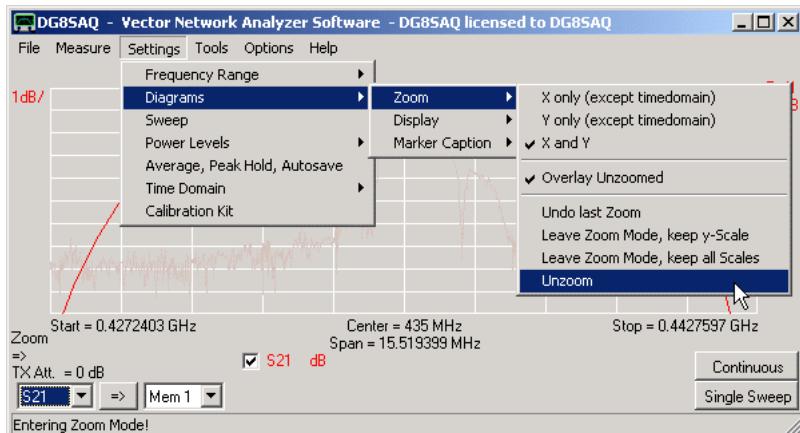


SETTINGS - DIAGRAMS - ZOOM

The **zoom** settings allow to control **box-zooms**, which can be accomplished by drawing a box with the mouse keeping the left mouse button pressed around the trace feature of interest.

►►► **Note:** Zooming into Smith charts is not implemented.

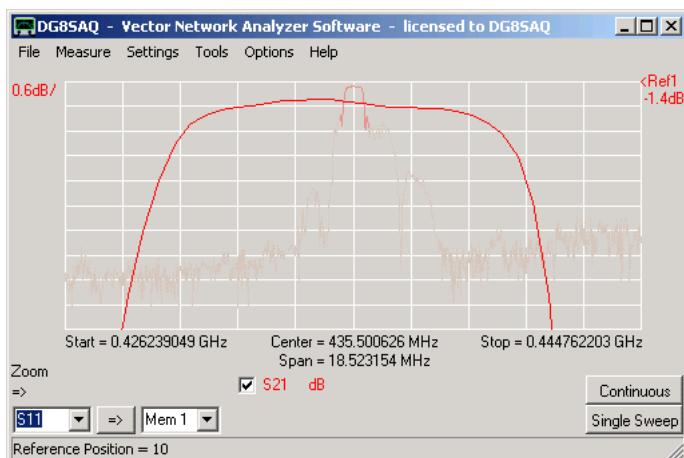
►►► **Note:** The zoom menu items below the separator line are only visible in zoom mode, i.e. when a box zoom has been performed. **Zoom mode is indicated by the "Zoom-label"** on the lower left of the main window (see below screenshot).



Choices:

- **X only ...** = only zoom the frequency axis, leave all y-axes unchanged (not applicable in time domain mode).
- **Y only ...** = only zoom vertically, leave the frequency axis unchanged (not applicable in time domain mode).
- **X and Y** = zoom frequency axis and y-axis

- **Overlay Unzoomed** = If selected, a greyed version of the unzoomed data is displayed with the zoom range highlighted together with the zoomed trace:



The following choices are only visible in zoom mode:

- **undo last zoom** = restore the x- and y-scales of before the last zoom
- **leave zoom-mode keep y-scale** = restore the full frequency span but keep the zoomed vertical scale
- **leave zoom-mode keep all scales** = keep the zoomed frequency span and the zoomed vertical scale. The data outside the last zoom box is lost, the visible data is interpolated to the full number of data points grid. A sweep after this will only sweep the visible frequency span.
- **unzoom** = restore the original x- and y-scales of before all consecutive zooms

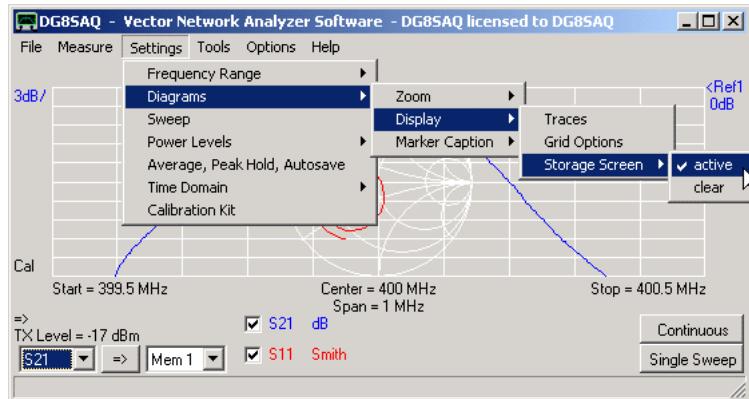
►►► **Hint:** You can also unzoom by right-clicking the main graphics grid.

►►► **Hint:** You can also unzoom by right-clicking the zoom label on the main window.

SETTINGS - DIAGRAMS - DISPLAY

The VNWA main menu "Settings - Diagrams - Display" offers the following functions:

- Traces**
- Grid Options**
- Storage Screen**
- Transparent Screen**



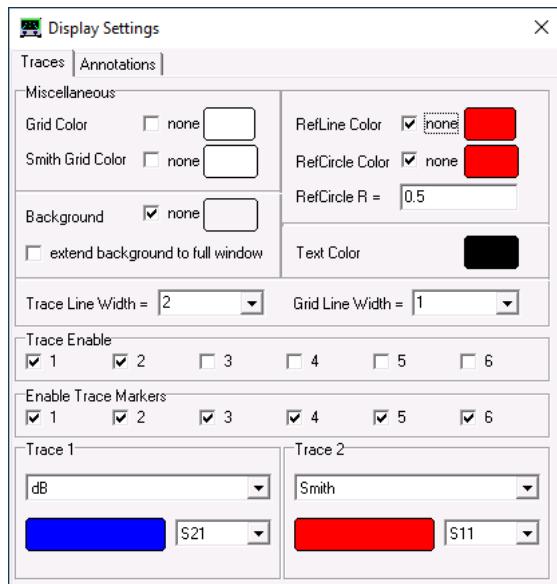
SETTINGS - DIAGRAMS - DISPLAY - TRACES

The "traces" settings allow to **control most displayed features on the main window**.

On clicking the menu item "**Settings - Diagrams - Display - Traces**", the display settings window will open.

Traces

The Traces tab allows to configure grids, traces and markers:



In the upper part of the miscellaneous panel the displayed **colors** can be configured. If "none" is selected, the feature is either not displayed or the standard windows color (grey) is used. To change a color, click onto the appropriate color field and select the desired color from the popping up color select menu.

►►► **Hint:** Two horizontal freely movable reference lines can be displayed. If these are not needed, check "RefLine Color" "none". These lines can either be positioned in the y-scale menu (accessed by double-clicking the y-scale labels (D,E) or by dragging them up or down with the mouse. Switching on the cursor when moving the lines with the mouse will simplify placing the lines to a specific level.

►►► **Hint:** A user definable reference circle can be displayed in the Smith chart. If it is not needed, check "RefCircle Color" "none". The circle diameter can be specified by the "RefCircle R" parameter. R = 1 puts the circle at the edge of the Smith chart.

►►► **Hint:** The traces' and grids' **line widths** can be specified to improve visibility outdoors in the sunlight. The default linewidth is 1 (= 1 pixel, thinnest lines)

The "**trace enable checkboxes**" select, which traces (=curves, maximum 6) may be displayed.

When checking one of the traces, an according trace information field pops up, where the **trace type** (S21, S11, ...) and the **display type** (dB, Smith...) can be selected.

The "**enable trace markers checkboxes**" select, which traces (=curves, maximum 6) will be allowed to carry markers. If unchecked, markers will not show for the selected traces, even though markers may be activated.

Available trace types:

S21, S11, S12 , S22	=> measured data
Memory 1...4	=> stored data
Unitarity	=> $ S11 ^2 + S21 ^2$ = fraction of power not dissipated
M.3/M.4	=> Memory 3 / Memory 4
S21/M.3	=> S21 / Memory 3
S11/M.4	=> S11 / Memory 4
TimeDomain	=> inverse FFT of selected data space, see example.

Gated F-Domain	=> FFT of gated time response, see example.
Custom1...6	=> arbitrary mathematical expression containing data space variables, see example.
s11,s12,...s33	=> 3-port S-parameters

Available display types:

dB	
Re dB	=> real part shown in dB
Im dB	=> imaginary part shown in dB
smith	
VSWR	
sVSWR	=> signed VSWR, negative for reflection coefficients > 1
real part	
imaginary part	
lin. magnitude	
phase	=> range -pi ... pi
continuous phase	=> extended range such that phase function becomes continuous
-cont. phase/f	=> minus continuous phase / frequency (up to software version 36.7.6) replaced by:
-cont. phase/w	=> minus continuous phase / $2\pi \cdot$ frequency (from software version 36.7.7 on)
group delay time	
polar mag/phase	
RADAR	
dFreq	=> frequency-center of RX input signal (only applicable for zero frequency span in VNWA mode for VNWA2 and VNWA3)
dFreq/Freq	=>(frequency-center)/center of RX input signal (only applicable for zero frequency span in VNWA mode for VNWA2 and VNWA3)
Frequency	=> frequency of RX input signal (only applicable for zero frequency span in VNWA mode for VNWA2 and VNWA3)

The following display types are only visible for trace types S11 and S22. Z denotes the **impedance** and Y the admittance calculated from the reflection coefficient.

real Z	
imag Z	
Z	
C --	=> series equivalent circuit capacitance, series resistance is real Z.
L --	=> series equivalent circuit inductance, series resistance is real Z
real Y	
imag Y	
Y	
R	=> parallel equivalent circuit resistance
X	=> parallel equivalent circuit reactance
C	=> parallel equivalent circuit capacitance
L	=> parallel equivalent circuit inductance
QC	=> quality factor of capacitor
QL	=> quality factor of inductor
smith renormalized	=> smith chart renormalized to user selectable complex impedances (power wave representation), see example

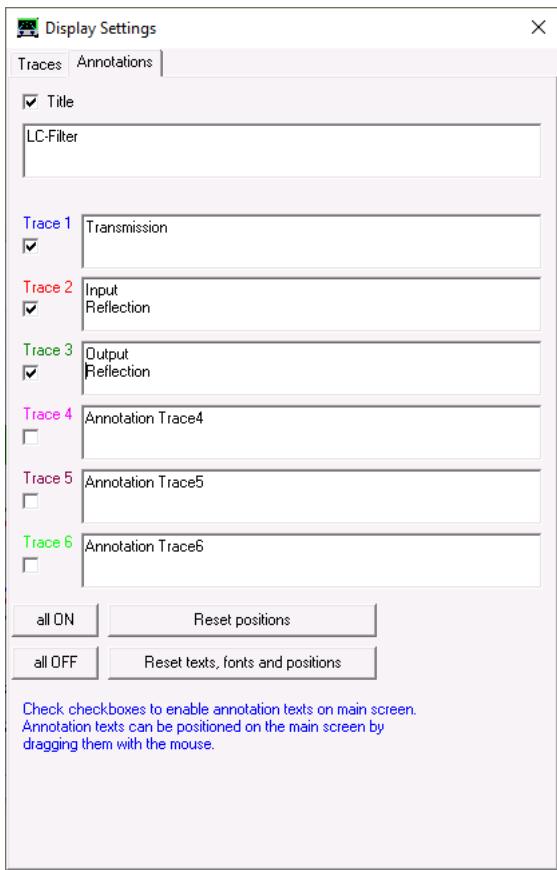
►►► **Hint:** Trace types and display types can also be changed in the main VNWA window by right-clicking the trace-type labels (K) and the display type labels (L) respectively.

►►► **Hint:** The **display settings menu** can also be accessed by double-clicking the trace select checkboxes (J), the trace-type labels (K) (except for trace type "custom", where the custom menu is invoked and for trace types "time" and "gated", where the time domain settings window is invoked.) and the display type labels (L).

►►► **Note:** Traces can be switched on or off with the trace select checkboxes (J).

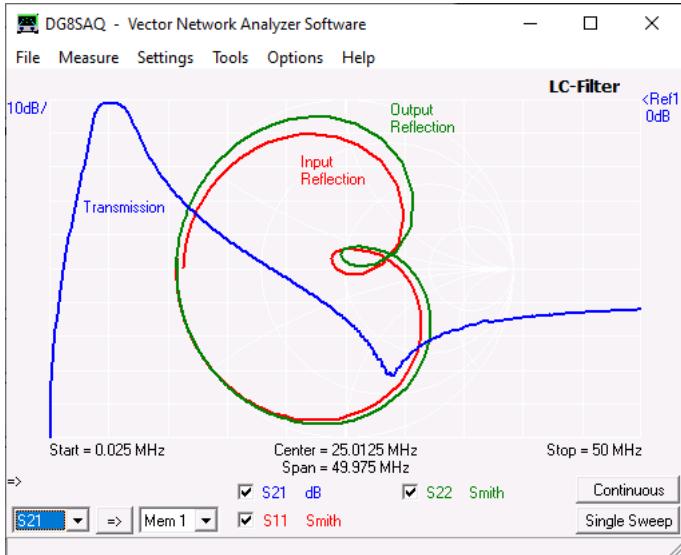
Annotations

The Annotations tab allows to configure a plot title and descriptive annotations to the displayed traces:



Note, that annotations may be multi-line.

Title and annotations will be seen on the main window:



The title and annotations may be moved by dragging them to the desired position using the mouse.

Text sizes and fonts may be changed by right-clicking one of the text labels.

Note, that title and annotations may have different fonts and sizes, but all annotations have the same font and size.

The title color is identical with the text color, the annotations share the corresponding trace colors.

Double-clicking title or any annotation will open annotations menu above.

SETTINGS - DIAGRAMS - DISPLAY - GRID OPTIONS

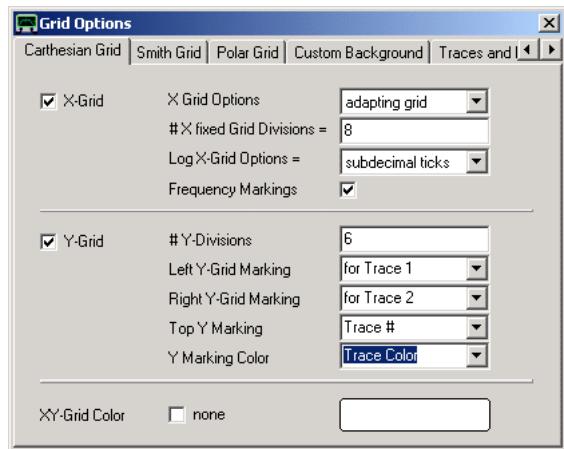
The "Grid Options" settings allow to control the appearance of the x-y grid (=Carthesian grid) and the Smith grid. Moreover, the trace and the background appearance can be controlled here.

On clicking the menu item "**Settings - Diagrams - Display - Grid Options**", the grid options window will open.

►► Hint: The same functionality can be accessed by right-clicking the grid on the main VNWA window and selecting **Grid Options**.

Many of the grid options are accessible in other windows, e.g. in the trace settings. All grid related settings are summarized here.

Carthesian Grid Options:



New grid options only accessible here are:

X Grid Options: only apply to a linear frequency grid.

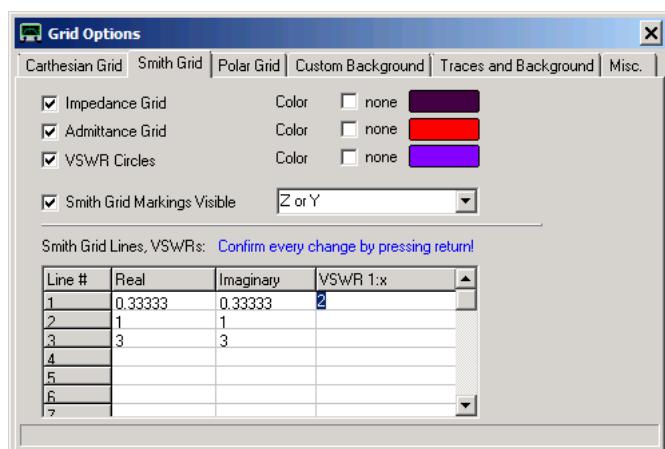
fixed grid: The vertical grid lines are placed equidistantly according to the "# X fixed Grid Divisions".

adapting grid: The vertical grid lines are placed equidistantly at easily human readable frequencies, e.g. at multiples of 100MHz.

Frequency Markings: If checked, the frequency of every vertical grid line will be displayed in the main window.

Left and Right Y-Grid Markings: Horizontal grid markings can be chosen for two traces and can be configured to show the trace number and to appear in the trace color.

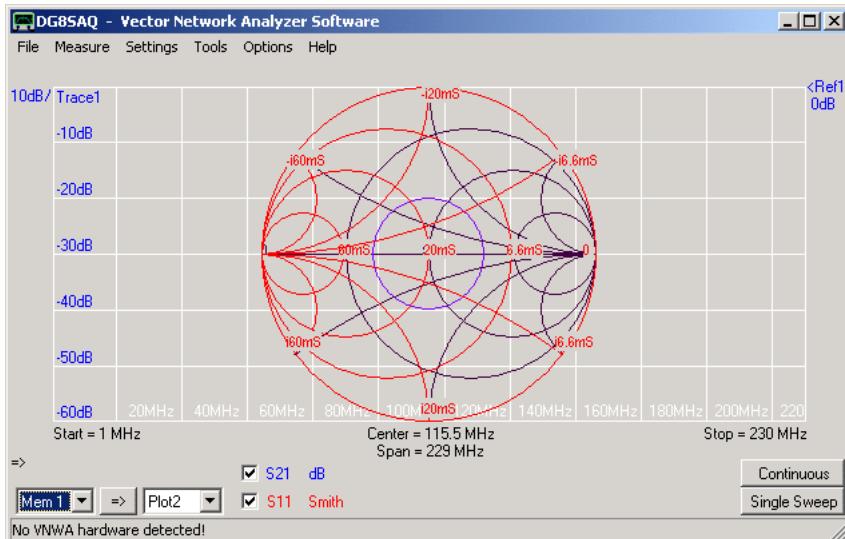
Smith Grid Options:



The Smith grid options allow to configure the Smith grid in various new ways:

- An **admittance Smith grid** (see red grid on below screen shot) can be selected in addition or alternatively to the default impedance Smith grid .
- **VSWR circles** can be added (see purple circle below).
- Note, that you can select to enter reflection coefficients or return loss instead of VSWR by clicking onto the table's VSWR title bar or the "VSWR Circles" checkbox.
- Grid and circle colors are customizable.
- All **grid lines are fully user configurable**. You can add or remove grid lines by adding or removing entries in the Excel-like table.
- **Smith grids** can be configured to show **markings** with impedance/admittance markings either normalized to 50 Ohms or absolute in Ohms/Siemens as seen below:

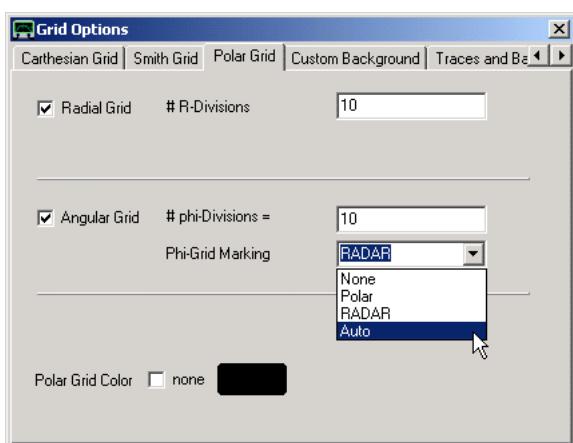
With above settings, the main window appears like this:



Note that the carthesian grid does not show right markings here, since trace 2 does not show in the carthesian grid but in the Smith grid.

Also note the adapting frequency grid in 20 MHz increments with frequency markings.

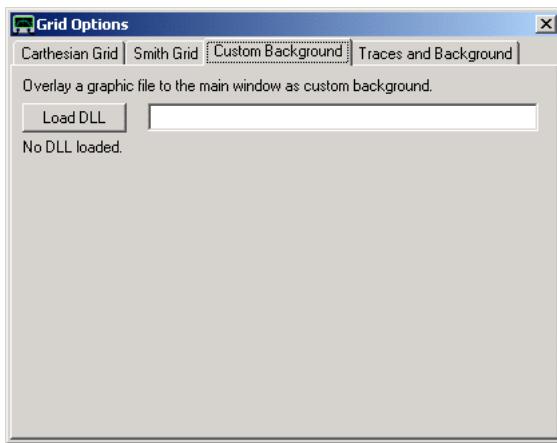
Polar Grid Options:



Here, the grids of polar and RADAR plots can be controled. Note, that while polar plots generally have the 0° orientation to the right, RADAR plots have it to the top.

Phi-Grid-Marking = Auto will automatically select the marking style according to the above rule.

Custom Background Options:



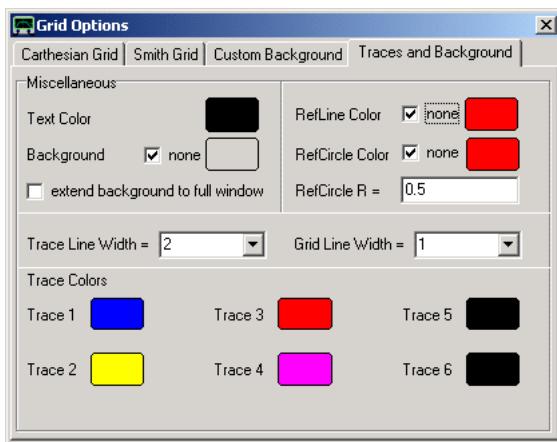
The custom background options allow to **load an image to appear as background on the VNWA main window**. Image file import is implemented in an optional external dynamic link library (dll) to avoid incompatibilities with older Windows versions (Windows98 and Windows2000 do not support GDI, which is used to read and display *.svg-files). Two different dlls are provided for reading image files:

- svg_dll.dll** supports import of scalable vector graphics files (*.svg)
- gfx_dll.dll** supports import of most pixel graphics files like *.bmp, *.jpg, *.png... .

Before importing an image, an appropriate dll-file must be loaded by pressing the "Load DLL" button and selecting the dll-file.

Once the dll is loaded, the controls for loading and manipulating an image appear.
See page "Importing and Manipulating a Background Image" for detailed examples.

Traces and Background Options:

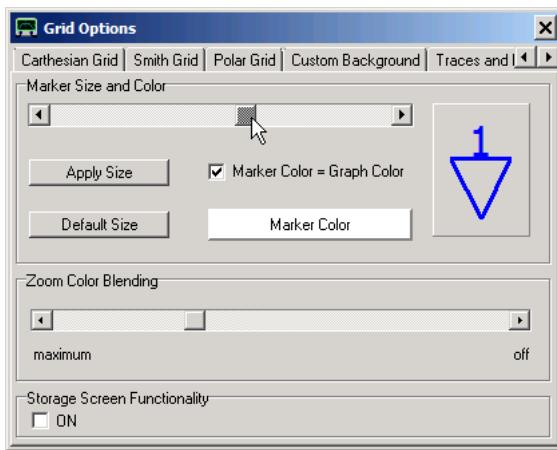


The traces and background menu allows to set the colors of traces, grid, background and reference lines and circles and do some related settings. This menu is redundant to the "Trace Options" menu.

Misc. Options:

Here some marker properties can be changed.

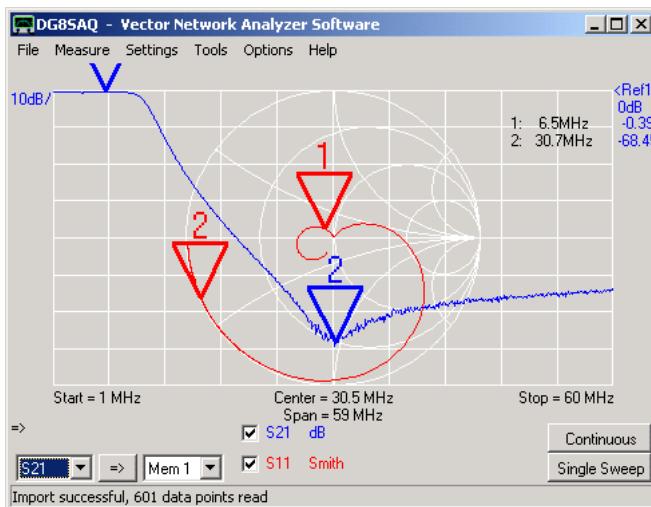
The **marker size** can be changed with a slider:



Note, that the actual markers will only change after pressing the "Apply Size" button or after closing the grid options window.

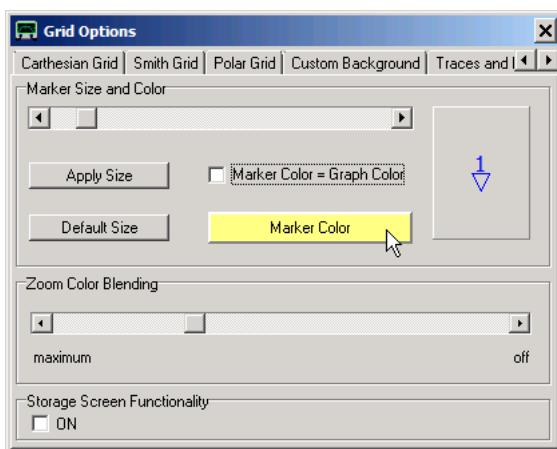
The button "Default Size" restores the default marker size.

After applying the selected marker size, the main window markers will appear in the same size as the example marker:

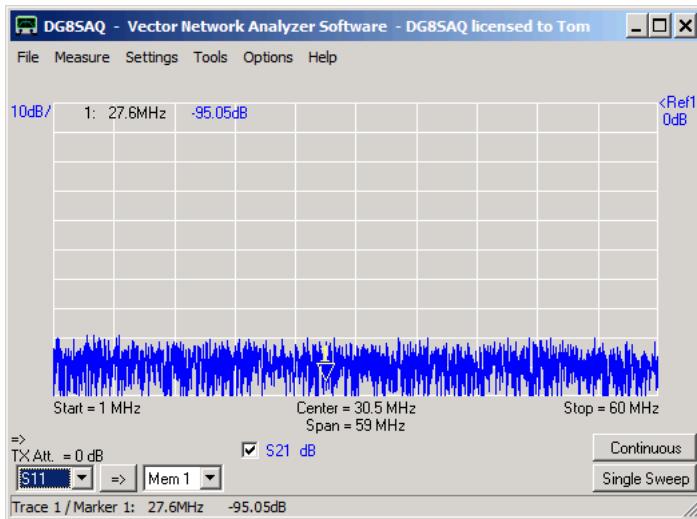


By default, the markers have the same color as the corresponding traces. Thus, sometimes it is hard to locate a marker inside a noisy trace.

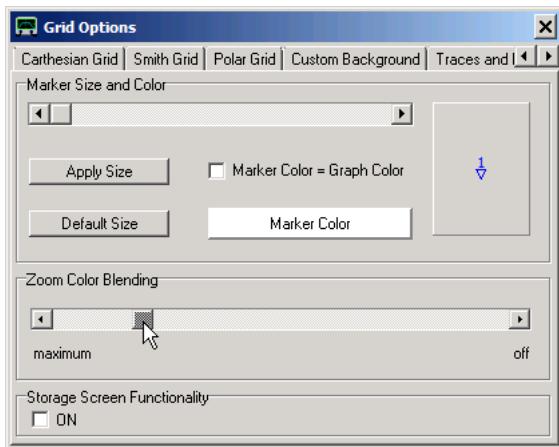
Therefore, the **color of all markers** can be changed simultaneously from the default trace color to a custom definable color by un-checking the "Marker Color=Graph Color" checkbox and selecting an appropriate color by clicking the "Marker Color" box:



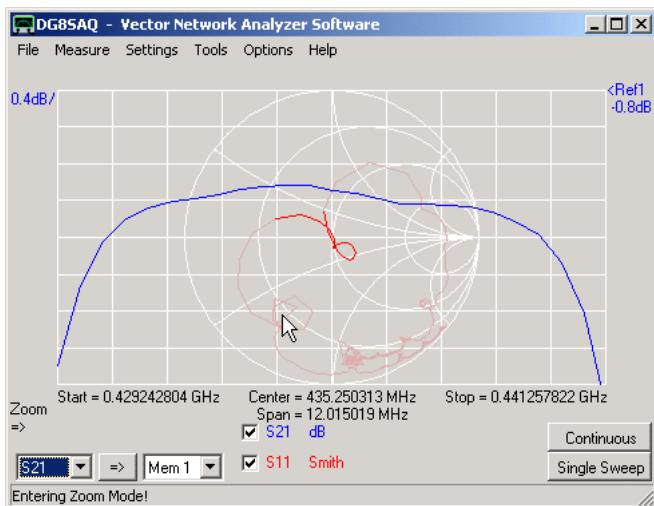
The now yellow marker is easily seen on the blue noise trace:



Zoom Color Blending:

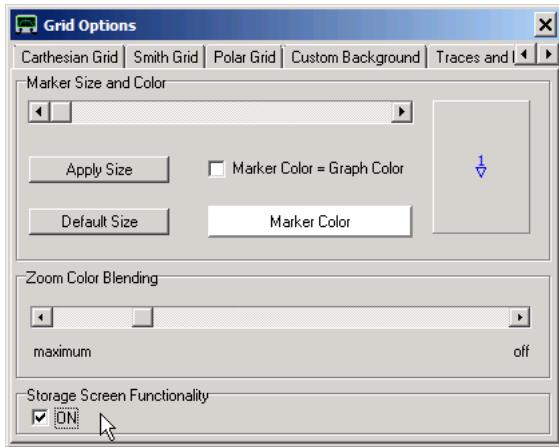


The "Zoom Color Blending" slider will change the shading of e.g. the Smith chart trace outside the zoom area (see mouse pointer):



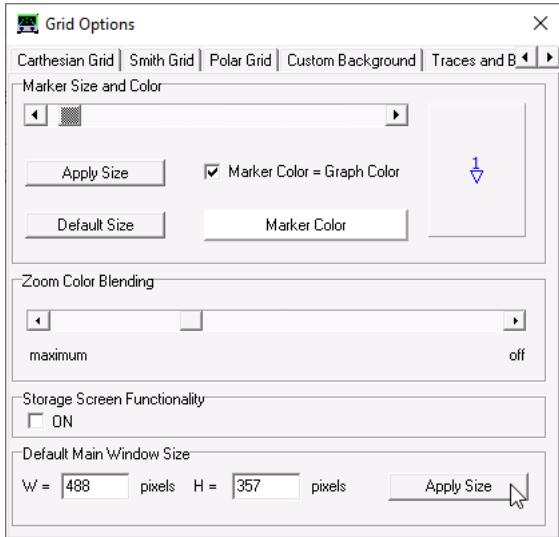
Maximum means the outside trace segments disappear while *off* means the whole trace is drawn in the same color.

Storage Screen Functionality:



From software version 35.8.q a storage screen is available, i.e. everything plotted to the main screen becomes permanent. This is useful if many consecutive sweep traces are to be compared.
See page Settings - Diagrams - Display - Storage Screen for more details.

Default Window Size Functionality:



From software version 36.7.9.8 on you can specify a default main window size, that you may want to use for documentation purposes.

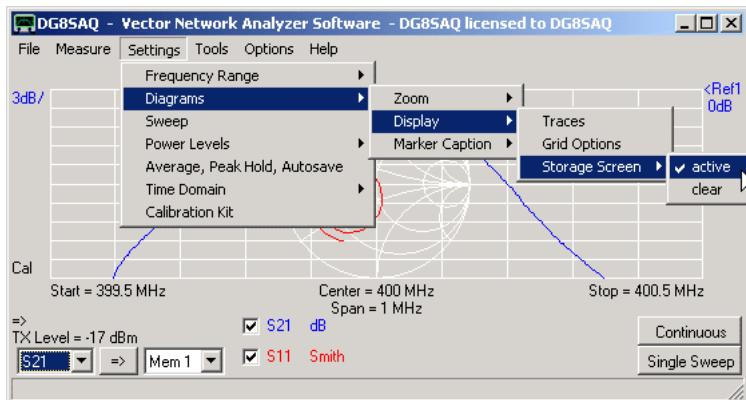
When pressing the "Apply Size" button, the main window size is set to the specified values.

Note, that you may also load the current main window size into the default size input fields by right-clicking any of the input fields and selecting the appropriate menu.

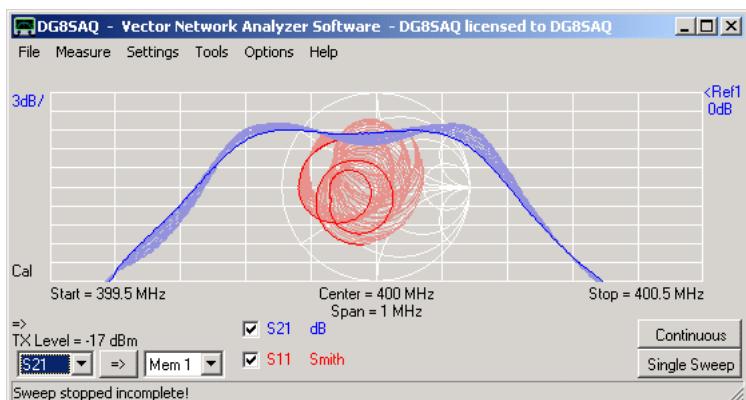
SETTINGS - DIAGRAMS - DISPLAY - STORAGE SCREEN

From software version 35.8.q a **storage screen** is available, i.e. everything plotted to the main screen becomes permanent. This is useful if many consecutive sweep traces are to be compared.

To activate/deactivate this feature click the main menu "**Settings - Diagrams - Display - Storage Screen - active**". Alternatively use the keyboard shortcut **ctrl-s**.

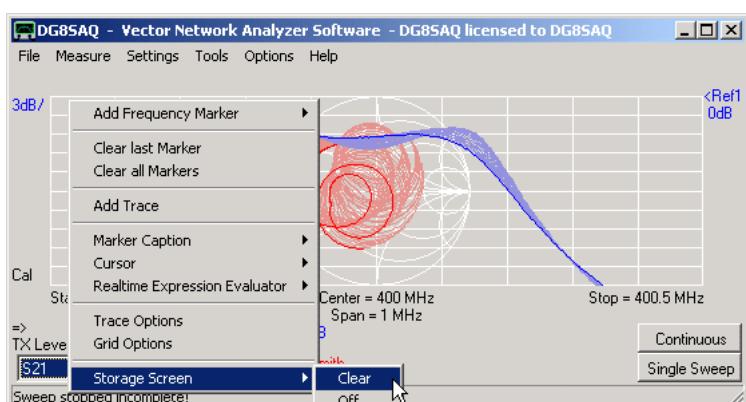


When the storage screen is activated, every sweep leaves a permanent trace on the plotting grid. The following shows a series of sweeps performed while tuning a filter:



The color pitch of the storage screen can be controlled with the Zoom Color Blending slider.

The **storage screen can be erased** by right-clicking onto the plotting grid and selecting "**Storage Screen - Clear**", via keyboard shortcut **ctrl-x** or via the main menu "**Settings - Diagrams - Display - Storage Screen - clear**", e.g.



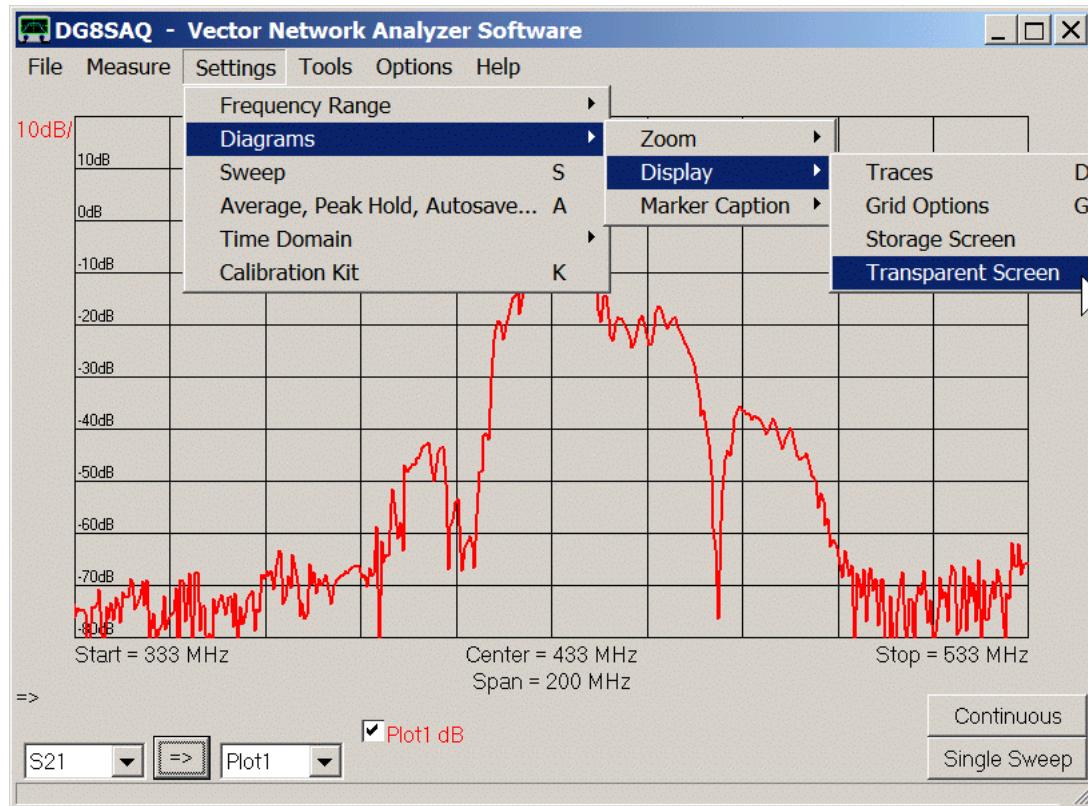
Note, that the storage screen is erased automatically, whenever the drawing grid changes size, upon a zooming action

or upon deactivation of the storage screen.

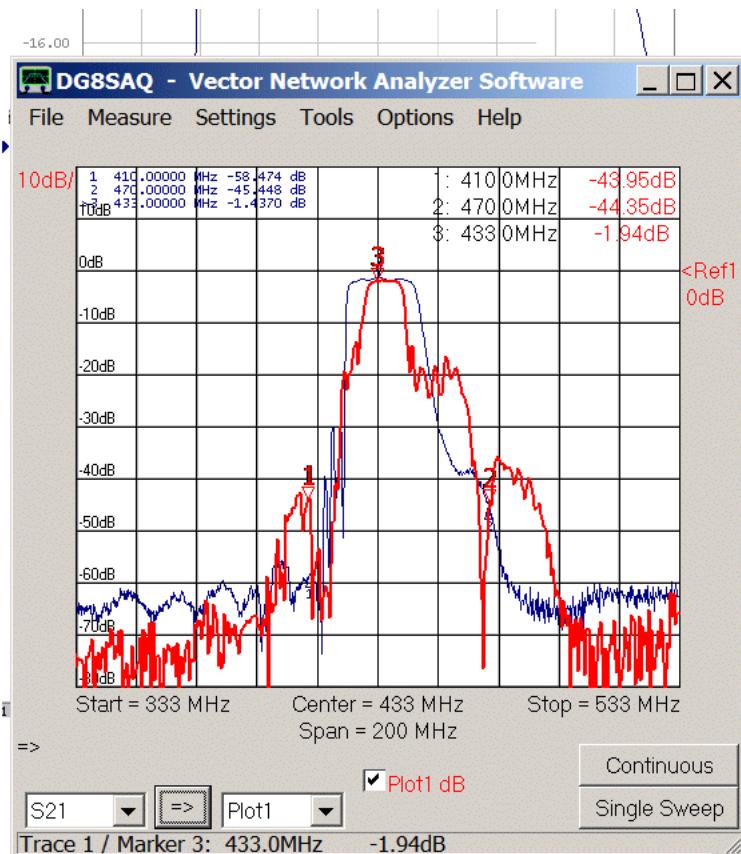
From software version 36.5.5 on the storage screen contents will be automatically saved upon program shutdown and it will be reloaded upon program restart. Also, the storage screen contents will be saved to and reloaded from saved instrument states. Moreover, the storage screen content can temporarily be saved and reloaded using the above visible storage screen popup menu.

SETTINGS - DIAGRAMS - DISPLAY - TRANSPARENT SCREEN

From software version 36.3.7.9 on, the background of the main window can be switched to **transparent** mode:



This is useful to quickly overlay a measurement or data file over a document e.g. a data sheet.
Simply move the transparent VNWA window over the document to be compared and adjust the VNWA scales to match the document's:

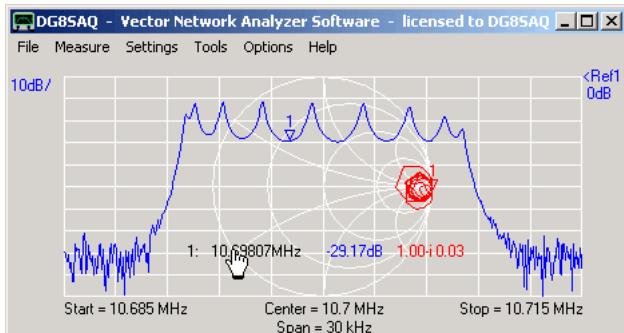


Above, the red trace and labels are VNWA data, while the blue trace and labels are on a pdf document placed under the VNWA window.

Note: Most screen capture softwares cannot capture semitransparent windows. Two softwares that can, are the Windows 7 built-in snipping tool and the freeware 7capture.

SETTINGS - DIAGRAMS - MARKER CAPTION

►►► **Note:** The marker captions can be moved on the main window by pointing onto the frequency part and dragging the captions around with the mouse, while keeping the left mouse button pressed.



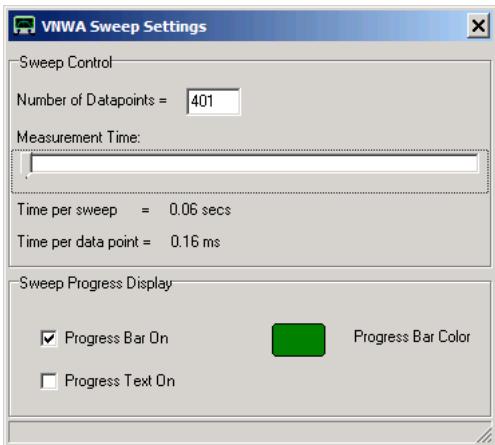
The "marker caption" settings allow to control the behavior of the marker captions on the main window.



- | | |
|------------------------------------|---|
| restore default
movable. | = places the marker captions to the default position in the upper left corner and makes it movable. |
| lock position | = locks the marker caption position such that it cannot be moved with the mouse |
| unlock position | = makes marker caption movable by the mouse after "lock position" again |

SETTINGS - SWEEP (VNWA)

The "sweep settings" allow to specify the **number of measured data points** as well as the **measuring time per data point**. From these values, a best guess **sweep time** is calculated, but as Windows is often doing unpredictable things, usually the sweep time is a bit longer in LPT mode. The prediction for USB mode is accurate.



►►► **Note:** Except for the fastest (<=1ms) time settings, **50% of the specified time are reserved for the DUT to settle** to the new frequency. The remaining 50% are used for measurement. In the <=1ms cases, the DUT settling time is specified in the setup window by the # of presamples.

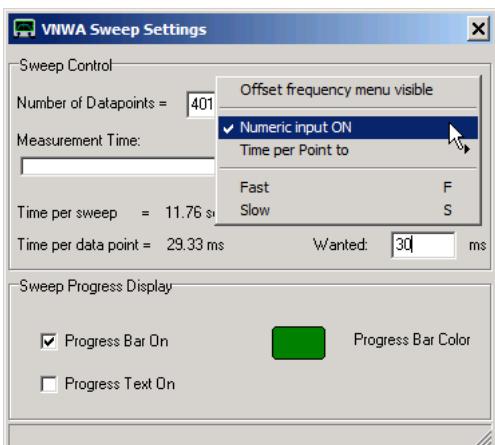
►►► **Note:** Sweep rates below 1ms / frequency point are only available in USB mode.

►►► **Note:** The sample rate slider will block on the left at the fastest possible sweep rate. If you want to sweep faster you need to increase the IF by decreasing the #samples per IF period and/or increase the sample rate.

Two kinds of **sweep progress indicators** can be switched on or off, namely a graphical progress bar appearing at the lower edge of the main VNWA window grid and a progress text appearing in the status line at the very bottom of the VNWA main window.

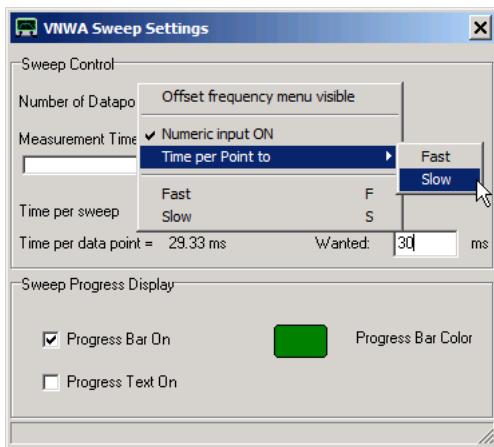
Advanced speed control:

Right-clicking the sweep control panel allows to activate a numerical input field for the time per data point:



In above example, 30ms have been entered into the "Wanted" data field. The software selects the nearest possible value, in above case 29.33ms.

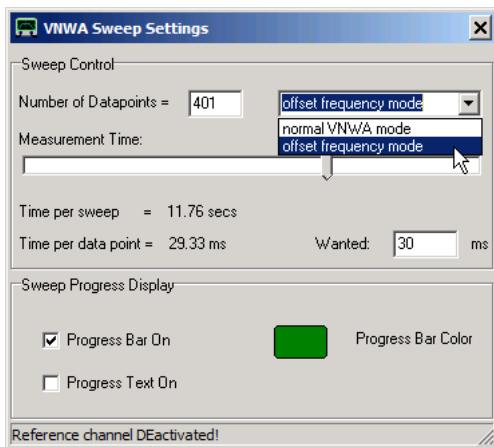
In the same way, two default speed values can be defined as "Fast" or "Slow" speeds:



Selecting "Time per Point to > Fast/Slow" will copy the currently selected time per point to a fast/slow memory, from which it can be restored when selecting "Fast" or "Slow" via above menu or by means of the keyboard shortcuts F or S respectively.

This is useful for quickly switching between two predefined measurement speeds.

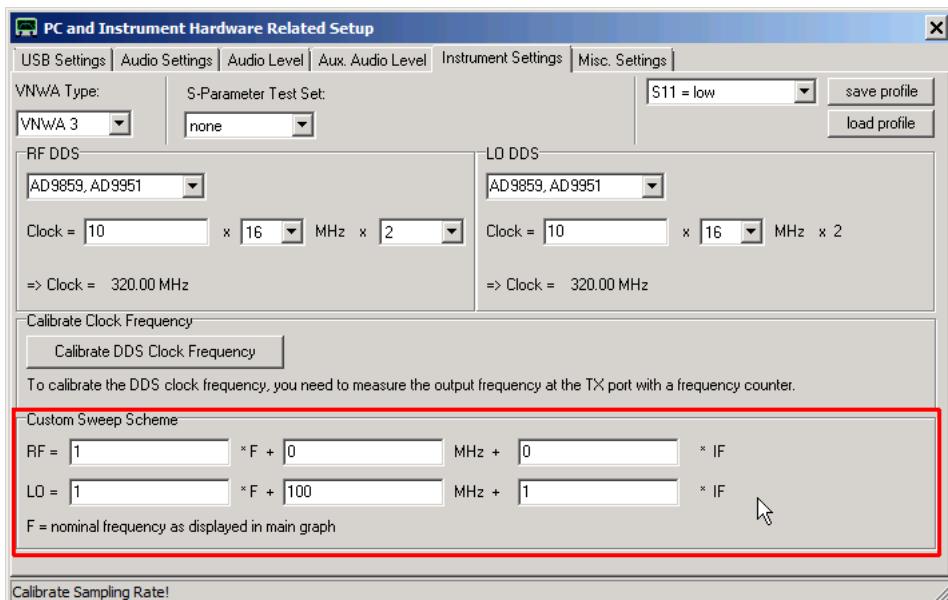
The right-click menu also allows to activate the **offset frequency menu**:



In **normal VNWA mode** the TX output frequency always tracks the RX frequency such that the RX always sees the TX signal if there is a connection between TX and RX.

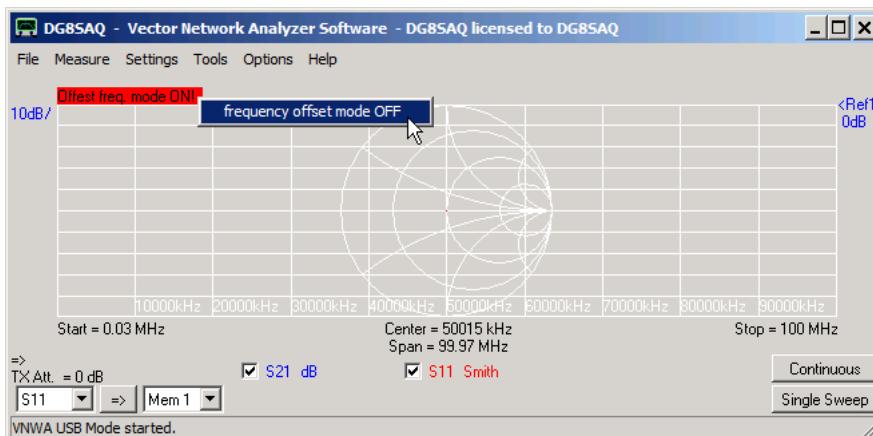
In **offset frequency mode**, the transmit and receive frequencies can have a frequency offset to each other.

The frequency offset can be custom defined in the VNWA main menu Options - Setup - Misc. Settings:



In above example, the RD oscillator which is fed to the TX port, tracks the displayed frequency. The receiver's local oscillator (LO) is offset by the intermediate frequency (IF) + 100MHz. This feature is particularly useful for performing compression measurements on mixers with the VNWA Power Sweep utility. Note, that the DUT mixer requires an LO signal of 100MHz which is locked to the VNWA clock. It is easiest to derive both the external mixer LO and the VNWA clock from an external multi-synthesizer. A VNWA3EC with external clock input is mandatory for such measurements. Note, that normalization with the reference signal is switched off in offset mode, as no internal reference signal can be produced since RF and LO do no longer mix into the instrument's IF bandwidth.

Offset mode is indicated by a red offset mode label in the VNWA main window:

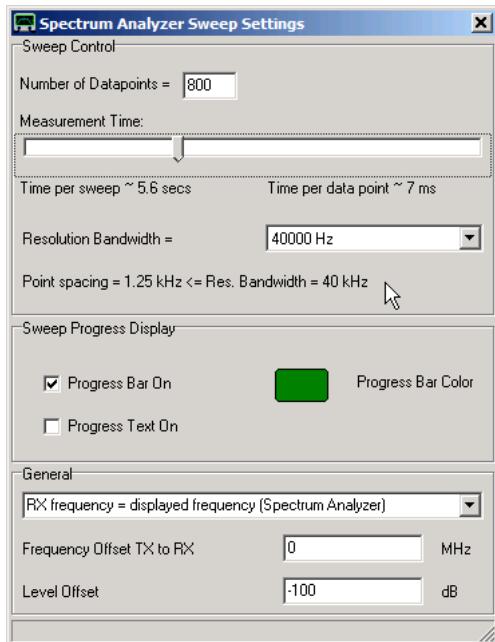


Offset mode can be deactivated by

- right-clicking the red offset mode label (see above)
- or in sweep settings via the offset mode menu.

SETTINGS - SWEEP (SPECTRUM ANALYZER - SA)

In SA-mode (=spectrum analyzer mode) the "sweep settings" allow to specify various settings.



1. The number of measured data points.

►►► **Warning:** Note, that in SA mode (unless used with tracking generator) the **frequency point spacing must be smaller than the resolution bandwidth!** If this is not obeyed, spectrum gets lost, e.g. there might be spectral lines but you won't see them. If settings are inappropriate, the **status text** (left of the mouse arrow in above screen shot) will turn red.

2. The measurement time per data point: From these values, a best guess **sweep time** is calculated, but as Windows is often doing unpredictable things, usually the sweep time is a bit longer in LPT mode. The prediction for USB mode is accurate.

Note, that the minimum sweep time depends on the resolution bandwidth. The slider will lock at the minimum allowed measurement time.

Note, that measurement times below 1ms / frequency point are only available in USB mode.

3. The resolution bandwidth.

►►► **Warning:** If the resolution bandwidth is smaller than the frequency point spacing, then the status text below the entry field turns red (see 1.).

Note, that changing the resolution bandwidth to lower values might automatically increase the measurement time setting if necessary.

4. Two kinds of sweep progress indicators can be switched on or off, namely a graphical progress bar appearing at the lower edge of the main VNWA window grid and a progress text appearing in the status line at the very bottom of the VNWA main window.

5. A frequency axis option: Three options are available:

a) RX frequency = displayed frequency (Spectrum Analyzer)

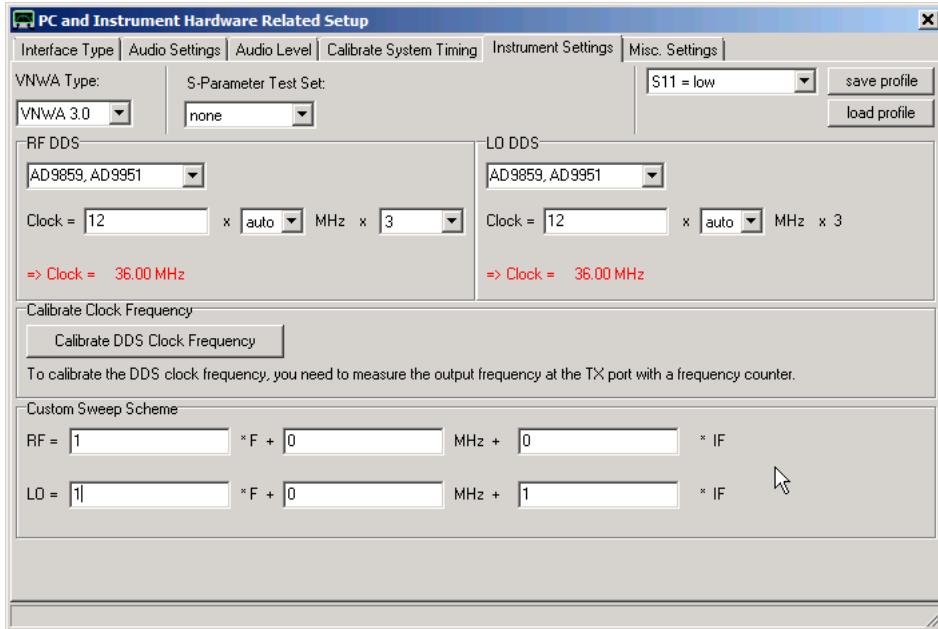
This option will set the main window frequency axis to the receive frequency of the RX such, that spectral lines are shown properly centered to their center frequency. Note, that the TX frequency is offset by the VNWA IF in this case.

b) TX frequency = displayed frequency (Tracking Generator)

This option will set the main window frequency axis to the transmit frequency of the TX such, that in tracking generator mode the measured frequency response is shown properly. Note, that the RX frequency is offset by the VNWA IF in this case.

c) custom frequency scheme (see Setup - Instrument Settings) This option allows the user to customize the frequency axis to e.g. account for external frequency converters or frequency multipliers. The customization is specified

in the "custom sweep scheme" input field in "Setup"-Instrument Settings":



RF denotes the frequency of the RF DDS.

LO denotes the frequency of the LO DDS.

F denotes the frequency axis.

IF denotes the RX intermediate frequency.

The above example shows a situation, where the TX frequency (=RF) is displayed and the LO of the RX is offset by the IF above the TX.

Note, that the above shown "custom sweep scheme" input field (next to the mouse pointer) is only visible if option c) (custom frequency scheme) is activated.

6. A frequency offset to detune the TX signal by the specified offset from the RX frequency. The displayed frequency axis shows the RX frequency. Say, you display a frequency range of 1...10 MHz and the specified offset is 10 MHz. Then the TX sweeps like a **tracking generator** the range of 11...20 MHz. This feature is useful to measure transfer functions of **converters and mixers** using an additional external local oscillator. Note, that the frequency of the **external LO** must be identical with the selected frequency offset to within the resolution bandwidth. Note, that todate no calibration can be performed if the offset is not zero. For zero offset, a regular through calibration is possible, but then you might rather use the VNWA mode to measure transmissions.

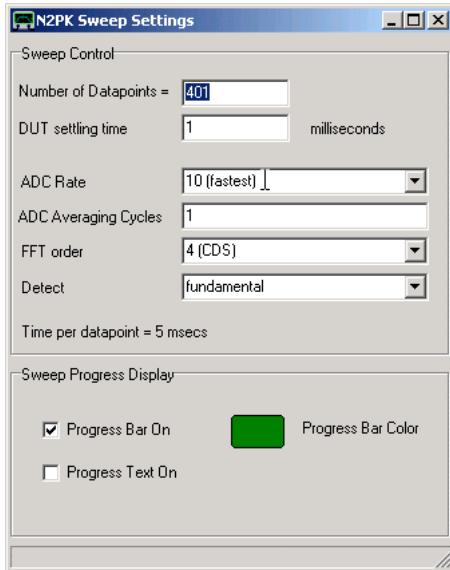
7. A level offset to adapt the displayed amplitude levels e.g. to show dBm. The offset level must todate be determined with an external oscillator with known output level, as it depends on the sound card sensitivity.

SETTINGS - SWEEP (N2PK-VNA)

The "sweep settings" allow specification of the **number of measured data points** as well as the **measurement timing**.

Moreover, the **analysis mode** (normal, CDS, higher order FFT and harmonic detection) is specified here. From these values, the time per data point and the sweep time are calculated and displayed.

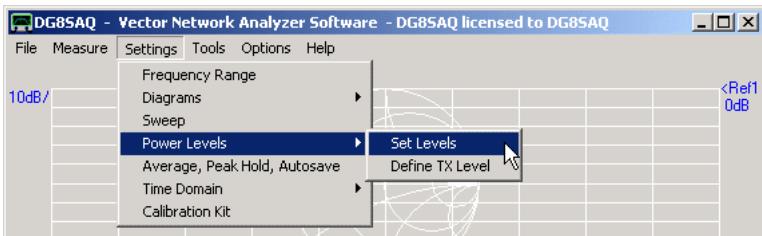
Finally, two kinds of **progress indicators** can be switched on or off, namely a graphical progress bar appearing at the lower edge of the main VNWA window grid and a progress text appearing in the status line at the very bottom of the VNWA main window.



SETTINGS - POWER LEVELS

This menu item allows to **control the VNWA oscillators' signal levels**. Firmwares v4.15 (VNWA2) and v5.10 (VNWA3) are mandatory for this functionality.

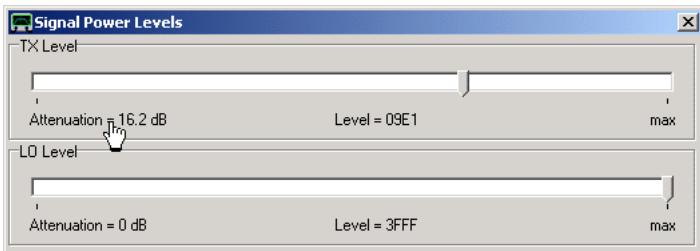
The menu item is only visible if the software/firmware combination supports level control.



Set Levels command

Invoking the "**Set Levels**" command will open the "Signal Power Levels" window.

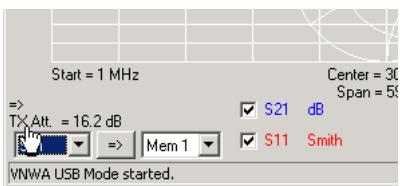
This menu item can also be accessed by the shortcut key "L" or via the main window level label context menu.



The TX and LO levels can be changed with the sliders. Note, that the LO level should normally be set to maximum as shown above.

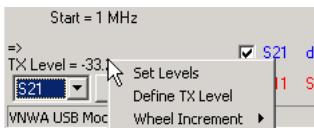
The left attenuation label below the slider (see mouse pointer above) shows the attenuation value (or absolute level if selected, see below). Note, that the level can also be controlled in very fine steps with the mouse wheel, when the mouse pointer is held over this label. Observe the "Level" label change in unity steps when the mouse wheel is turned. The "Level" label shows the DDS amplitude word in hexadecimal representation. Note, that you can scroll the levels below the slider accessible values. I have found the levels to deviate strongly from the theoretical values for attenuations higher than 50dB and therefore I have excluded the high attenuation range from the slider accessible values.

The TX level attenuation value (or absolute level if selected, see below) is also displayed in the main window level label:



Note, that you can also control the TX level by holding the mouse pointer over this label and scrolling the mouse wheel. Double-clicking this label will invoke the "Signal Power Levels" window.

Right-clicking the label will open the level context menu:

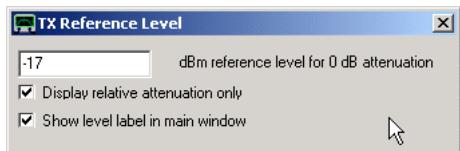


From here, you can also invoke the "Set Levels" command.

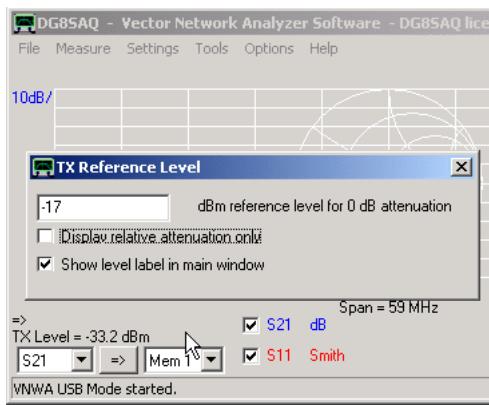
Also, you can set the desired increment for mouse wheel action on the main window power label.

Define TX level command

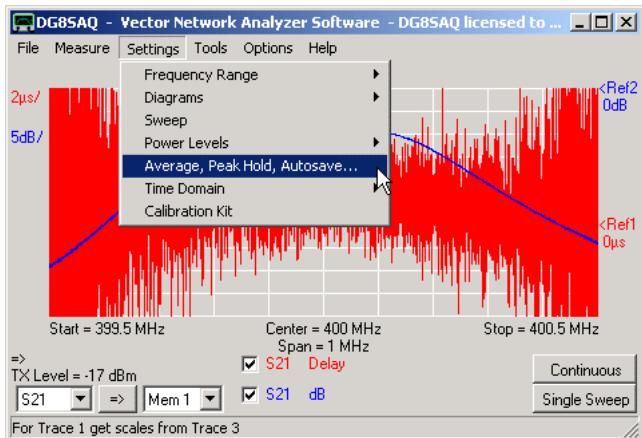
Invoking the "Define TX Level" command from the context menu or from the main menu will open the "TX Reference Level" window:



If you specify an absolute TX reference level for 0 dB attenuation and **UN**-check "Display relative attenuation only", then the main window power label will display the absolute TX level instead of an attenuation value as shown in the screenshot below. Note that below main window power label may be completely deactivated by unchecking "Show level label in main window".



SETTINGS - AVERAGE, PEAK HOLD, AUTOSAVE...



The " Average, Peak Hold, Autosave..." settings allow to

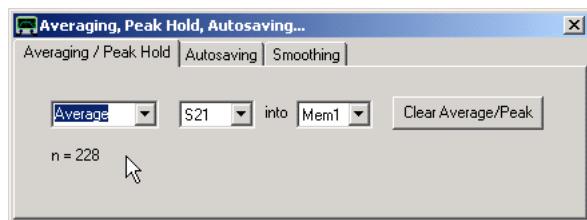
- average measurement results over any number of sweeps
- determine the maximum of any number of sweeps at each frequency point and display it (peak hold)
- autosave sweep data after a sweep to any memory space
- smooth noisy data by averaging over neighboring frequency points

SETTINGS - AVERAGE, PEAK HOLD, AUTOSAVE... - AVERAGING/PEAK HOLD

Averaging and Peak Hold can be activated with the left dropdown menu box:

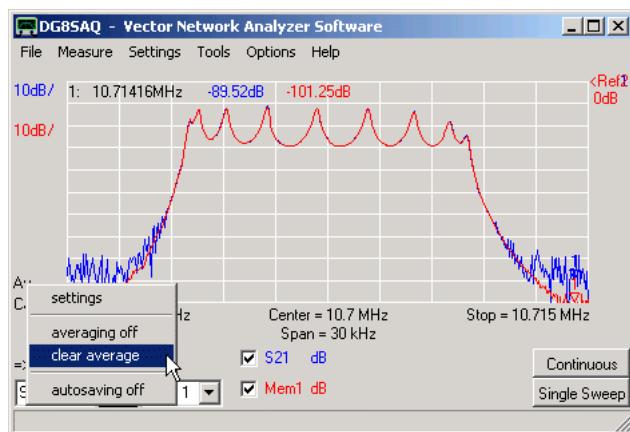


Averaging



In the above example, Mem1 accumulates the average of consecutive S21 sweeps. n = 228 means, that already 228 sweeps have been averaged up to now. This number is reset to zero when pressing the "Clear Average" button. At the same time, the the averaging memory space (Mem1 in above example) is cleared i.e. zeroed.

The effect of averaging can be seen in the following image:



Averaging effectively reduces the noise. The averaged red trace is a lot less noisy than the single sweep blue trace.

►►► **Note:** You can also access some averaging functions by right-clicking the "Av" label on the main window (partly hidden by the upper left corner of the popup menu in above screenshot).

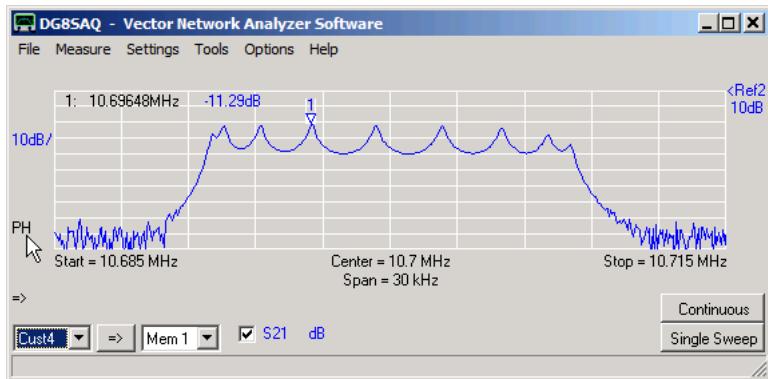
►►► **Note:** The averaging label is only visible, if the averaging function is activated.

►►► **Note:** Holding the mouse pointer over the averaging label will invoke a pop-up text with the current average number n.

Peak Hold

The peak hold function will determine after each sweep for every frequency point the bigger amplitude of the recent sweep and the stored sweep and stores this maximum value of the two. Thus over the time and consecutive sweeps, at each frequency point, the biggest ever encountered amplitude will be accumulated. This function is well known in

spectrum analyzers. It may also be useful for tuning filters. Trace selection and reset functionality is the same as for averaging. If the peak hold function is activated, this is indicated by the PH label in the main window.

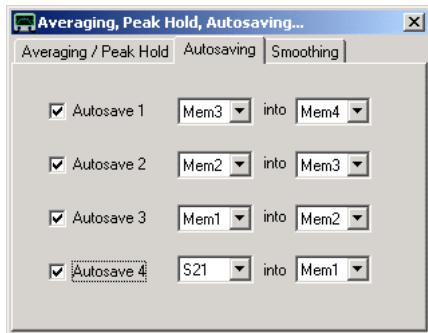


- »»» **Note:** You can also access the peak hold functions by right-clicking the "PH label" on the main window (see above).
- »»» **Note** The PH label is only visible, if the peak hold function is activated.
- »»» **Note:** Holding the mouse pointer over the PH label will invoke a pop-up text with the current sweep number n.

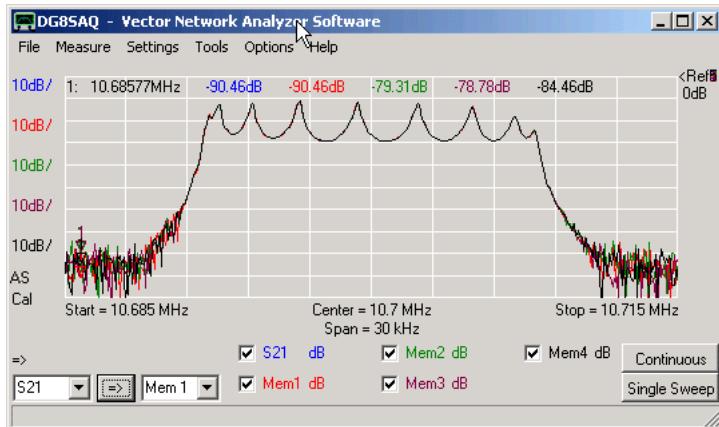
SETTINGS - AVERAGE, PEAK HOLD, AUTOSAVE... - AUTOSAVING

Autosaving

Autosaving means automatically saving data at the end of each sweep to another memory space.



The effect of the above settings is, that the last four sweeps are automatically displayed and updated after each sweep:



This feature is useful when tuning devices, where it is advantageous to see the changes from sweep to sweep.

The activated "auto saving" function is indicated by the AS label on the left side of the main window (see above).

►►► **Note:** You can also access some autosave functions by right-clicking the "AS label" on the main window (see above).

►►► **Note** The AS label is only visible, if the autosaving function is activated.

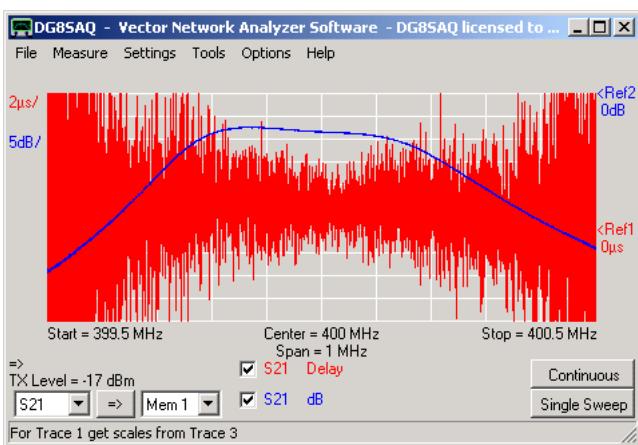
SETTINGS - AVERAGE, PEAK HOLD, AUTOSAVE... - SMOOTHING

Smoothing

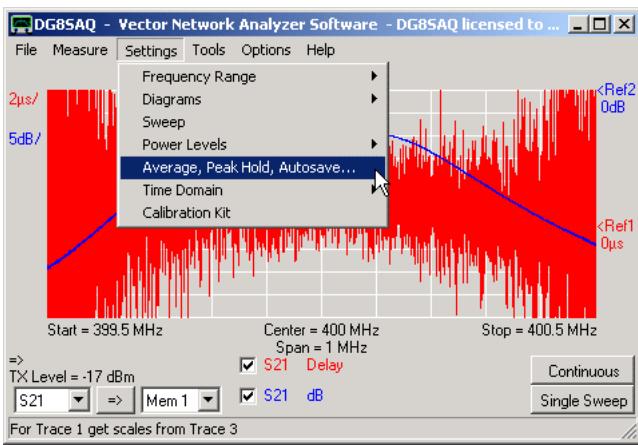
Smoothing means to average data over several neighboring frequency points. This can be used as a means to reduce noise, but care must be taken not to smooth out real features of the data.

Example:

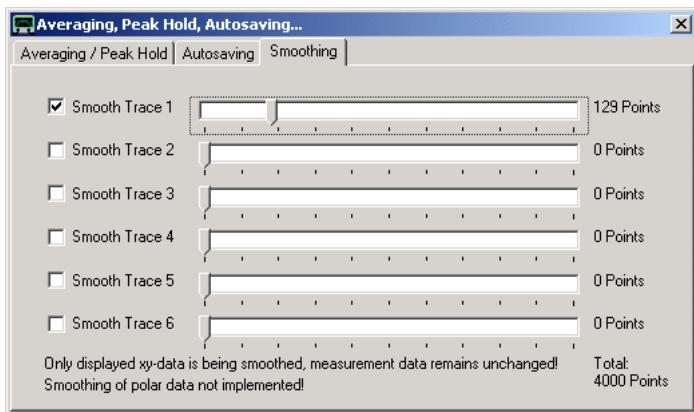
Below is a S21-measurement of a bandpass filter passband showing attenuation (blue) and group delay time (red) with 4000 frequency points. The noise level was artificially increased by an attenuator in front of the RX port, which was calibrated out. While the amplitude data is quite smooth, the group delay data is very noisy because it was obtained by numerical differentiation of the original measurement data. The higher the number of points, the larger the numerical noise in this differentiation process will be.



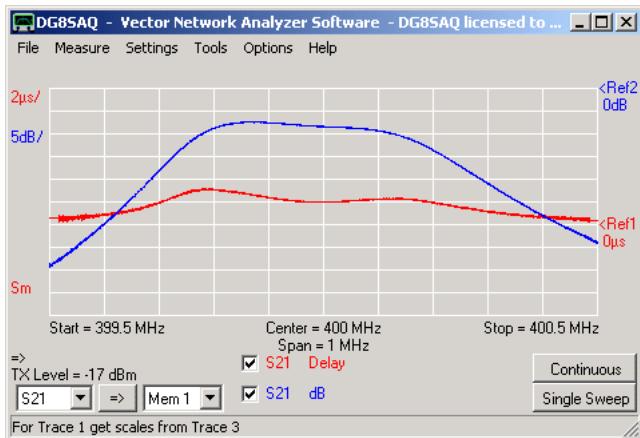
As the group delay of such a filter is a relatively smooth function and there is a vast number of points in this measurement to use for smoothing, the actual group delay can be obtained by invoking the smoothing functionality:



Select the trace to be smoothed, in this case Trace1 (=red trace). The corresponding slider adjusts the averaging width, i.e. how many frequency points are being used to calculate the average.



Move the slider and observe how the traces change:



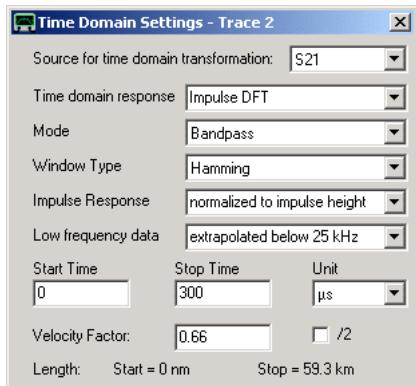
Smoothing could efficiently remove the noise from the group delay trace.

►►► **Note:** If smoothing is activated, a red **Sm** label is shown on the lower left edge of the main screen. Right-click or double-click this label to access some of the smoothing functionality.

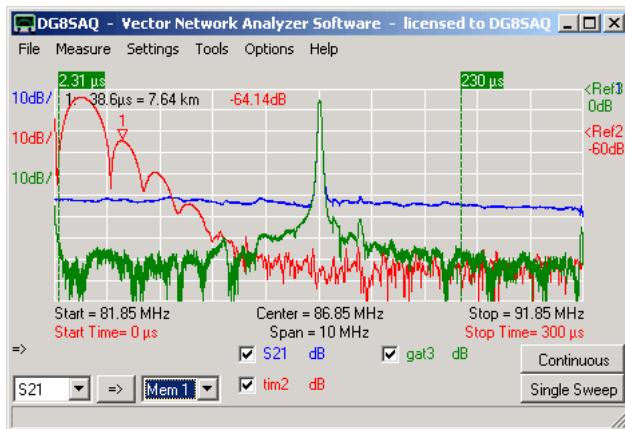
►►► **Note:** The straight stubs at both ends of the red smoothed trace above indicate the smoothing width. Frequency points too close to the range edges cannot be smoothed, therefore those points are set to the same value as the closest averaged point, which produces the straight stubs.

SETTINGS - TIME DOMAIN

The "time domain settings" allow display and analysis control of time domain data.



With the above settings, after each sweep an inverse discrete Fourier transform (DFT) of the S21 data will be calculated and displayed in a time range of 0...300 us if trace type "time" is activated (see below, red trace 2). The frequency data is weighted with a Hamming window function prior to calculating the Fourier transform in above example.



Time markers display time, distance and level. The distance is calculated from the time with the vacuum speed of light and a velocity factor. This is useful to determine lengths of coaxial cables.

The /2 checkbox allows to get proper lengths on reflection measurements, where the signal travels through the cable twice (forward and backward).

►►► **Hint:** You can access the time domain settings also by double-clicking the red start and stop time labels or the Time or Gate label in the main window.

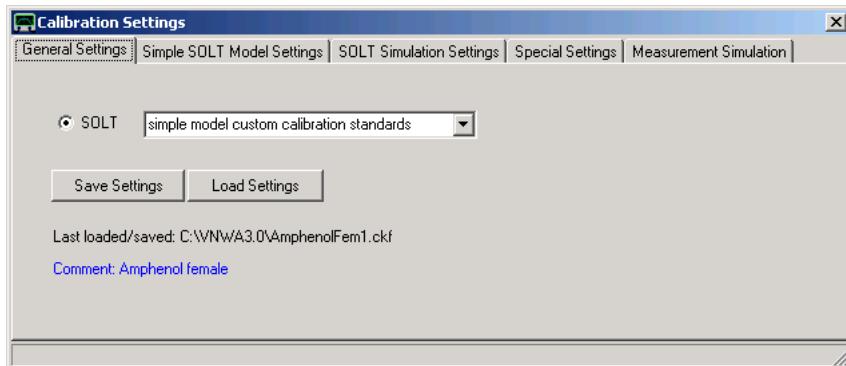
►►► **Hint:** You can separately add frequency and time markers.

►►► **Hint:** Besides the impulse response, also the **step response** of a DUT can be displayed selectable with the "**Time domain response**" dropdown menu. This is useful to determine transmission line impedances and impedance variations.

For an introduction to time domain measurements and practical examples see here.

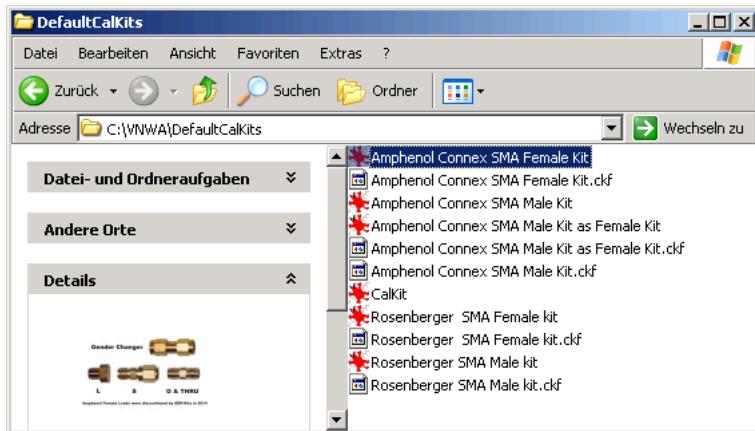
SETTINGS - CALIBRATION KIT

The **calibration kit settings** allow to specify the parameters of the user calibration kit (physical open, short, load, thru standards) used to calibrate the VNWA. The software needs to know the electrical properties of the calibration standards in order to compensate measurement errors e.g. due to test cables by using calibration measurements.



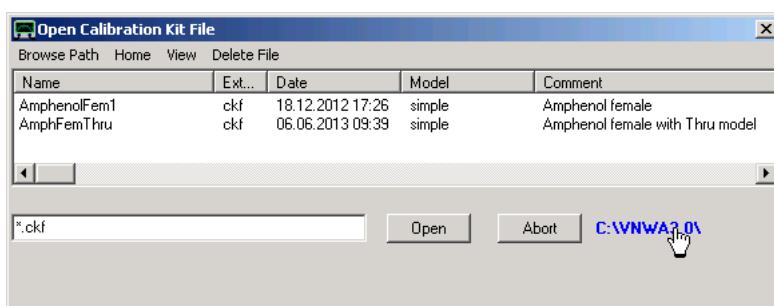
See page "Calibration Standard Setup" for details on calibration kit models and corresponding settings.
See page "Extracting calibration kit parameters" for a simple method to obtain model parameters of an unknown calibration kit.

From software version 36.7 on, the VNWA installer also installs a set of default calibration kit model files prepared by Kurt Poulsen OZ7OU corresponding to the calibration kits supplied by SDR-Kits:



These model files (*.ckf) and corresponding descriptive image files (*.jpg, optional) can be loaded into the VNWA software by means of the "**Load Settings**" button.

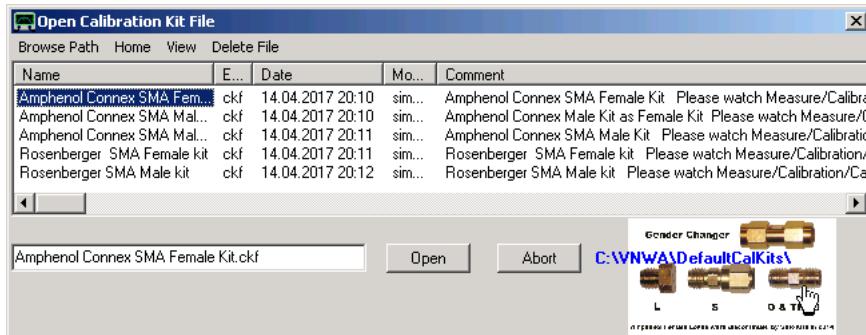
Upon pressing "**Load Settings**" the calibration kit file manager will open and display all available cal kit files with their properties:



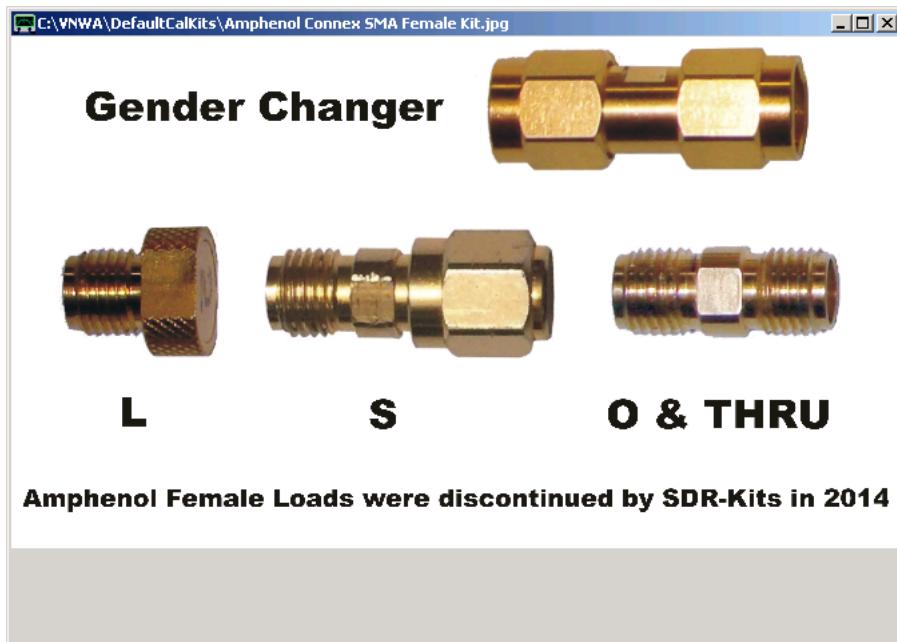
►►► **Hint:** You can also browse and change the file path by clicking the blue path label on the bottom right.

In order to locate the default calibration kit files, browse for the VNWA installation folder (normally c:\VNWA\l) and select the subfolder .\DefaultCalKits therein.

When highlighting one of the models by clicking on it, the corresponding descriptive image will be shown on the bottom right:



The descriptive image may be enlarged by double-clicking it:



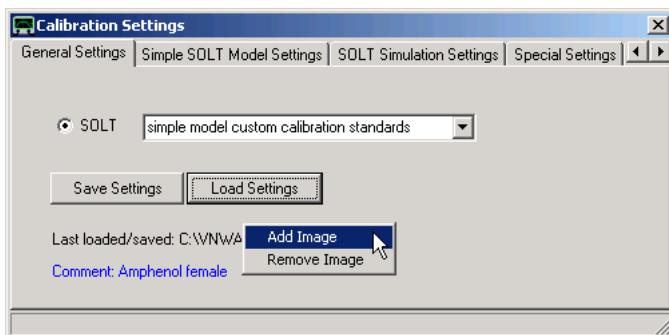
A cal kit can be loaded from the calibration kit file manager by double-clicking the corresponding file name in the list or by selecting the file and pressing the Open button or the return keyboard key.

If a calibration kit model file includes a descriptive image, it will be shown in the calibration kit settings menu together with the model comment after loading:



►►► **Hint:** You may have to increase the window size to see the image.

You may remove the image or load a different one by right-clicking the window area:

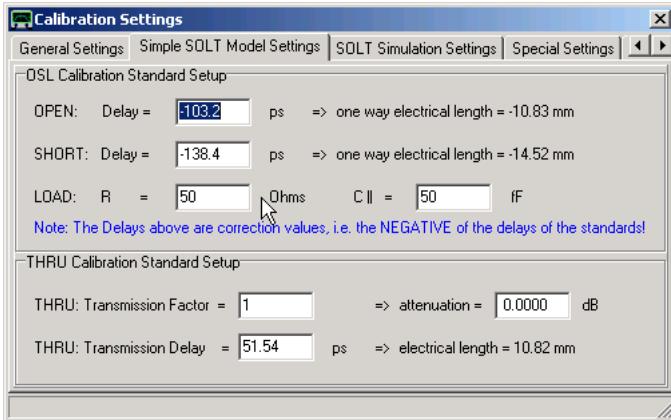


When adding an image, it is highly recommended to save the model data together with the just added image by means of the Save Settings button.

The image will be saved under the same name with extension .jpg as the model data for later automatic loading.

►►► **IMPORTANT:** When using default cal kit models, make sure to change to load resistance to the exact value of your load standard and save the modified model to a directory different from the ..\DefaultCalKits folder, as the latter will be overwritten with every software update you install!

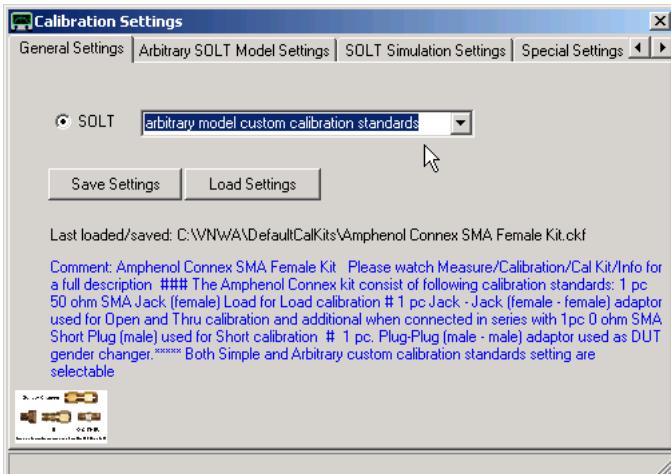
Below, one of the default cal kit data sets is shown:



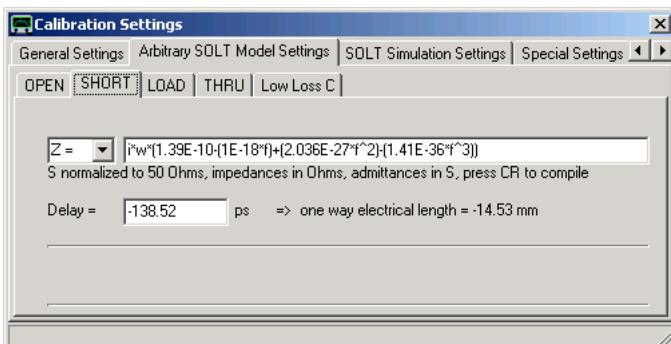
►►► **Note:** The above default data set still shows ideal 50 Ohms load resistance. You must modify this to the exact value of your load.

►►► **Note:** OSL delays are two way delays. Lengths are calculated from half the delays using the velocity factor specified in the Special Settings tab. If the calibration standard is longer than the position of the desired calibration plane indicates, a negative delay is to be entered.

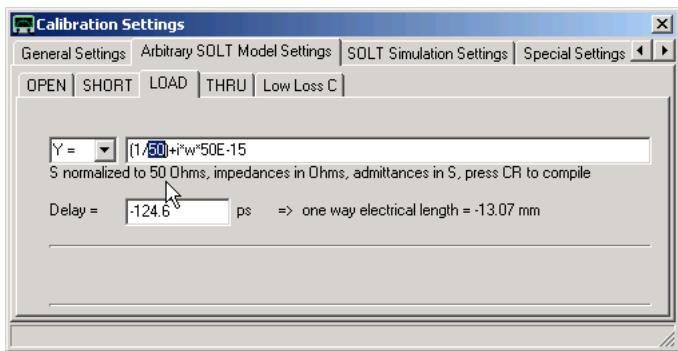
►►► **Note:** The default cal kit data sets also include more accurate arbitrary cal models:



As example, here is the load "open" model:



If you use these arbitrary models, make sure to also change the "load" model here to reflect your exact load dc resistance (ideal 50 Ohms value highlighted below):



TOOLS

The VNWA main menu "Tools" offers the following functions:

Matching Tool
Restore Unmatched
Copy Display Data to S2P Buffer

Crystal Analyzer
3-Port Analyzer

Specification Tester

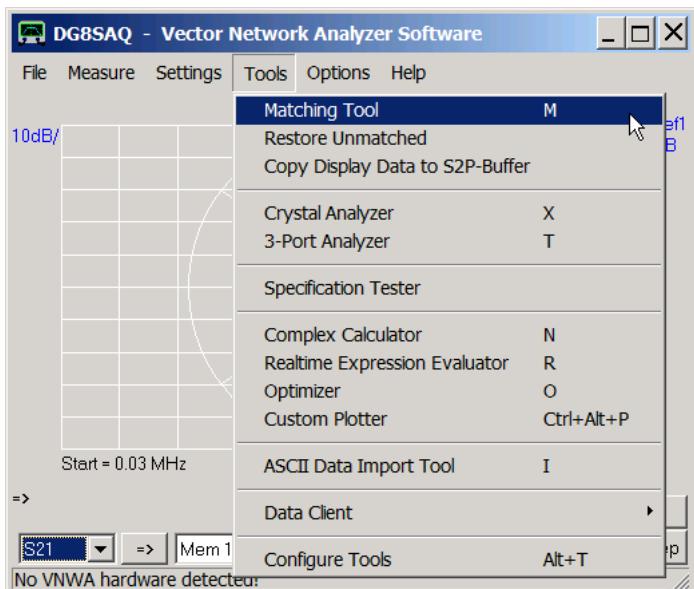
Complex Calculator
Realtime Expression Evaluator
Optimizer
Custom Plotter

Virtual Keyboard (32 bit Windows versions only)

ASCII Data Import Tool

Data Client

Configure Tools



TOOLS - MATCHING TOOL

The VNWA's built-in ***matching tool*** allows to **simulate the 2 port DUT's transfer characteristics under arbitrary impedance termination conditions** and to **calculate matching networks**.

This is useful for the application of non 50 Ohms filters like crystal filters, where matching networks need to be optimized.

►►► **Note:** The original S-parameters normalized to 50 Ohms are stored in a separate internal S2P-buffer and remain untouched during the simulation. Simulation results will be stored and displayed via the display memory spaces (S11, S21, S12, S22), which will change in turn. The original 50 Ohms S-parameters can be restored from the S2P-buffer into the display buffers by the main menu command "Tools"- "Restore-Unmatched".

►►► **Note:** The matching tool can only be invoked, if the internal S2P-buffer contains valid S-parameters. You can copy the display memory spaces (S11, S21, S12, S22) into the internal 2-port buffer and thus make them valid S-parameters with the main menu command "Tools"- "Copy Display Data to S2P Buffer".

A detailed example is found [here](#).

TOOLS - RESTORE UNMATCHED

"Restore Unmatched" copies S-parameters from the S2P-buffer into the display buffers (S11, S21, S12, S22) .

►►► **Note:** The "restore unmatched" can only be invoked, if the internal S2P-buffer contains valid S-parameters.

TOOLS - COPY DISPLAY BUFFER DATA TO S2P BUFFER

"Copy Display Buffer Data to S2P Buffer" copies the display memory spaces (S11, S21, S12, S22) into the internal S2P-buffer and thus makes them valid S-parameters, which can be used in conjunction with the built-in matching tool.

TOOLS - CRYSTAL ANALYZER

The "**crystal analyzer**" tool allows to extract equivalent circuit model parameters from the measured reflection coefficient of a crystal resonator or similar resonator (SAW, ceramic,...)

See page "Crystal Analyzer: Extracting Resonator Model Parameters" for detailed examples.

TOOLS - 3-PORT ANALYZER

Three port RF devices like baluns or SAW-filters with e.g. balanced outputs confront the user with a complex characterization task:

On the one hand, one would like to know e.g. the insertion attenuation from single ended input to differential outputs and the complex differential output impedance (**differential mode**). On the other hand, the **common mode attenuation** is of interest.

The 3-Port analyzer tool allows to perform both characterizations on a set of imported or measured 3-port S-parameters.

A step by step example of performing a 3-port analysis is found here.

TOOLS - SPECIFICATION TESTER

The **Specification Tester** tool is available from VNWA version 36.3.7.9 on.

It allows to measure components and compare them to a custom defined specification.

Measurement results can be logged to a file. Pass/Fail indication will be given acoustically and optically.

The Specification tester tool consists of 3 tabs:

Spec Syntax Tab:

```
Specification Tool
File Measurement Custom
Evaluator Specification Optimizer Settings Log Comments Spec Syntax
Source | Command | Parameter1 | Parameter2 | Lower Limit | Upper Limit
SOURCE = S11, S21, S12, S22, Mem1..4, Cust1..6
PARAMETERS:
frequencies in MHz or gains in dB
COMMANDS:
INMIN: min. insertion gain between frequencies P1 and P2, unit dB
INMAX: max. insertion gain between frequencies P1 and P2, unit dB
ARIPP: amplitude ripple between frequencies P1 and P2, unit dB
DRMAX: max. gain between frequencies P1 and P2 relative to last il*, unit dB
DAMAX: max. gain between frequencies P1 and P2, unit dB
VALUE: value at frequency P1, unit dB
VALPH: phase value at frequency P1, unit Degrees
DELAY: average group delay between frequencies P1 and P2, unit ns
GDMAX: max. group delay ripple between frequencies P1 and P2, unit ns
FCENT: center frequency at rel. P1-dB-level, unit MHz
FFMCE: center frequency at rel. P1-dB-level, unit ppm
BANDW: bandwidth at rel. P1-dB-level, unit MHz
FCABS: center frequency at abs. P1-dB-level, unit MHz
PFMAB: center frequency at abs. P1-dB-level, unit ppm
BANDA: bandwidth at abs. P1-dB-level, unit MHz
AZMIN: min abs. impedance between frequencies P1 and P2, unit Ohms
AZMAX: max abs. impedance between frequencies P1 and P2, unit Ohms
RZMIN: min re(impedance) between frequencies P1 and P2, unit Ohms
RZMAX: max re(impedance) between frequencies P1 and P2, unit Ohms
RZAVG: average re(impedance) between frequencies P1 and P2, unit Ohms
IZMIN: min im(impedance) between frequencies P1 and P2, unit Ohms
```

This is a quick reference of the spec syntax.

Specification Tab:

#	src	cmd	param1	param2	min	max
1	s21	fcent	-1.5	0	900	910
2	s21	inmax	890	915	-4	0
3	s21	drmax	935	960	-999	-15
4	s21	aripp	890	915	0	1.5
5						
6						
7						
8						
9						
10						
11						
12						
13						

Component 8 passed.

Here up to 20 spec items can be defined.

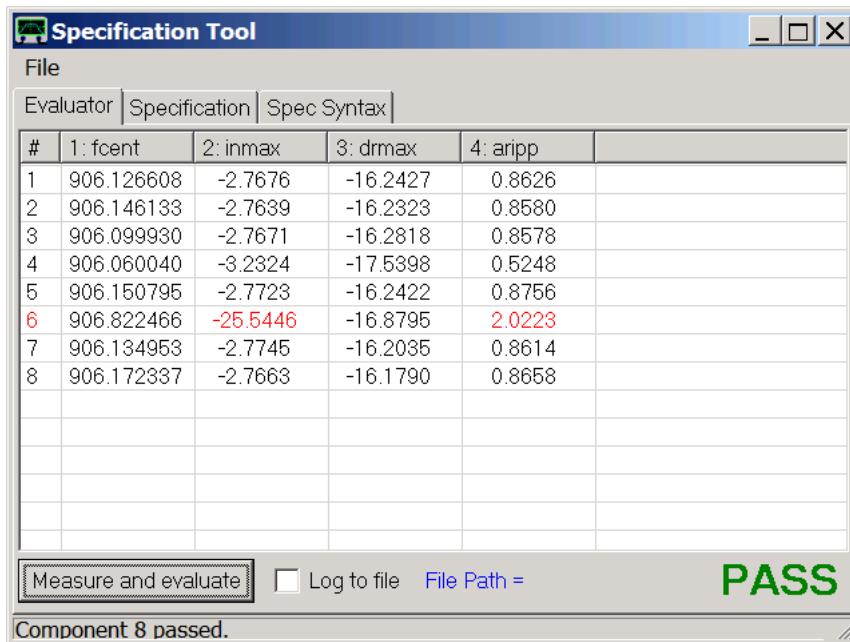
- The # -column denotes the spec reference number, which will show in the measurement log.
- The src -column denotes the source to be tested. Allowed choices are S11, S21, S12, S22, Mem1...Mem4, Cust1...Cust6.
- The cmd -column denotes the spec command as described in the Syntax tab.
- The param* -columns denote the parameters of the spec command as described in the Syntax tab.
- The min and max column denote the lower and the upper specification limit.

Example:

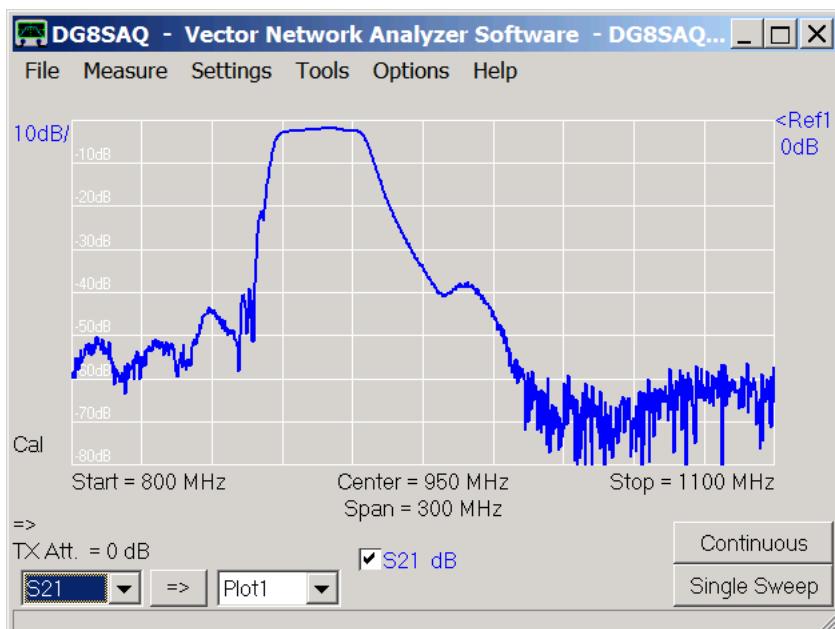
The four entries in above example all test S21 for:

- the -1.5dB center frequency, good is 900MHz...910MHz
- the maximum insertion gain (=negative attenuation) in the GSM RX band 890MHz...960MHz, good is anything between -4dB and 0dB.
- the attenuation in the GSM TX band 935MHz...960MHz relative to the previously evaluated maximum insertion gain, good is anything between -999dB and -15dB.
- the amplitude ripple in the GSM RX band 890MHz...960MHz, good is anything between 0dB and 1.5dB.

Evaluator Tab:



Upon pressing the **Measure and evaluate** button, VNWA performs a single measurement, e.g.



After completion of the sweep, the measurement is being automatically evaluated with respect to the entered specification. The results will be displayed in the **Evaluator** table, and, if **Log to file** is activated, the same results will also be automatically logged to a text file. To define a log file path, click onto the blue **File Path** label. The log file name will automatically be generated from the current time and date.

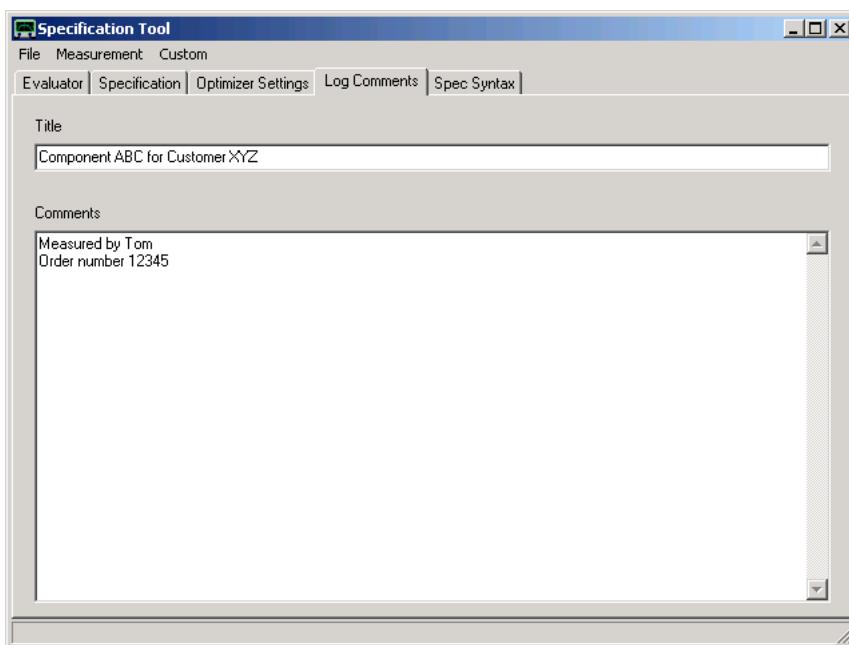
Pass/Fail information of the last measurement will be given optically on the lower right (green PASS or red FAIL label) and acoustically by two different sounds.

In above example, 8 components have been tested, of which component #6 has failed. Red entries in the table denote failures.

- ▶▶ **Hint:** The Evaluator table will be cleared when the Specification table is being accessed.
- ▶▶ **Hint:** Specifications can be loaded from or saved to a file via the **File** menu on the top left.

Log Comments:

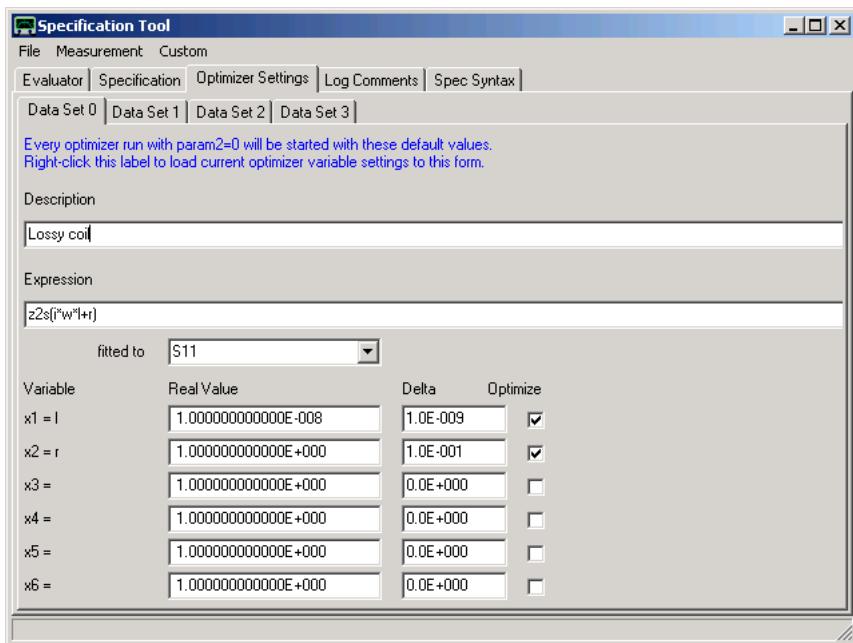
From VNWA version 36.7.5 on the user can specify a title and a comment to be inserted into the log file for later reference:



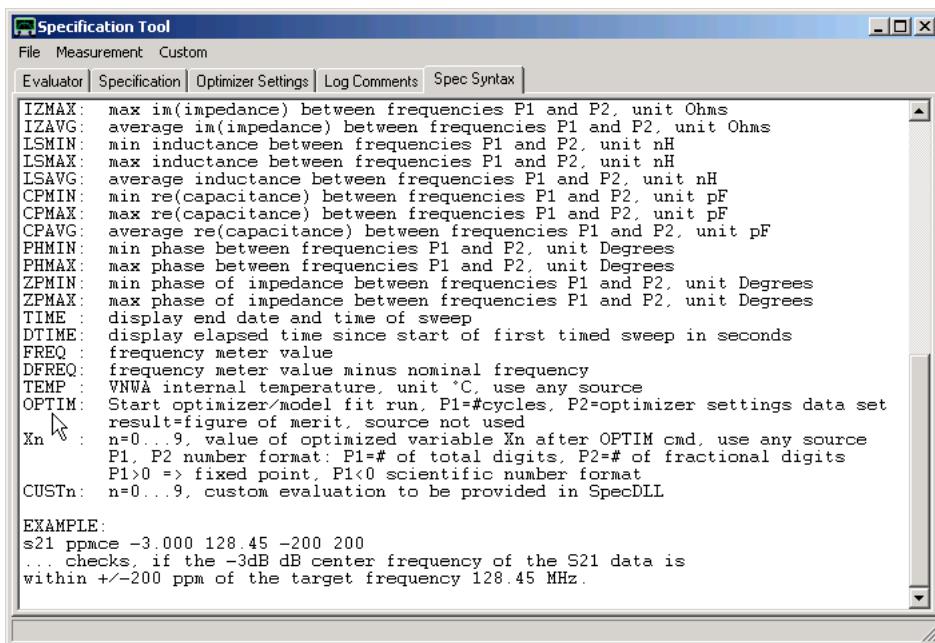
Optimizer Settings:

From VNWA version 36.7.5 on the user can invoke up to four different optimizer runs from within the specification tester in order to extract model parameters.

The four sets of optimizer parameters may be specified in the *Optimizer Settings* tab:



The corresponding syntax can be found in the *Spec Syntax* tab:

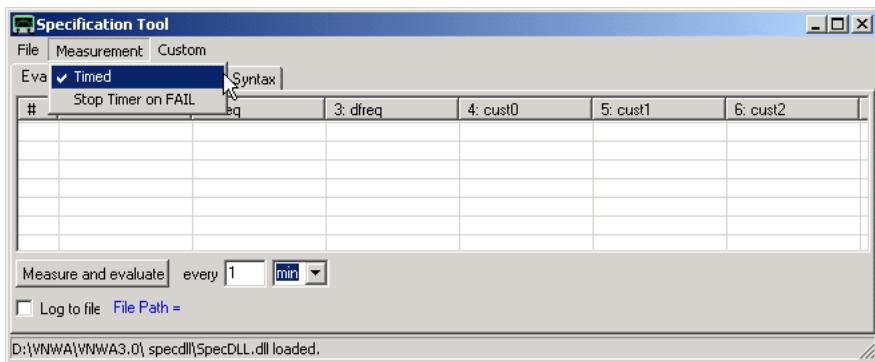


Relevant spec items are:

OPTIM invoke optimizer
X<n> read optimizer variable x<n>

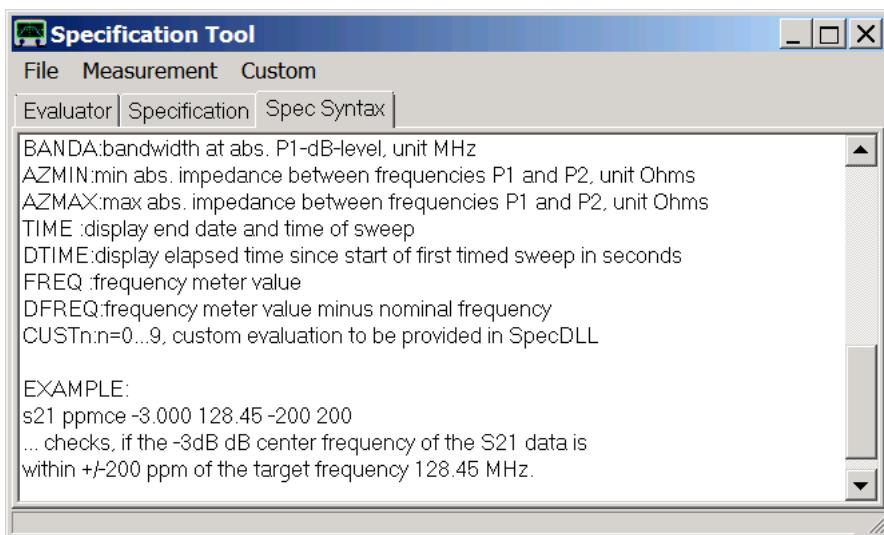
Timed Measurements:

From VNWA36.5.0 on measurements can be launched periodically by a custom definable timer. Timer measurements are useful for monitoring the same device over a long time e.g. under varying environmental conditions like temperature. This can also be used for automatic spectrum monitoring in Spectrum Analyzer mode.



With above settings, when pressing "Measure and evaluate", the button will remain in the pressed position and a measurement will automatically be performed every 1 minute and be checked against the user defined spec. The timed measurement process continues until the "Measure and evaluate" button is pressed again. As seen in above menu, the software can be configured such that the timer automatically stops, when the measurement fails to comply with the user defined spec.

For timed measurements several new spec commands have been implemented:



Particularly useful for timed sweeps are:

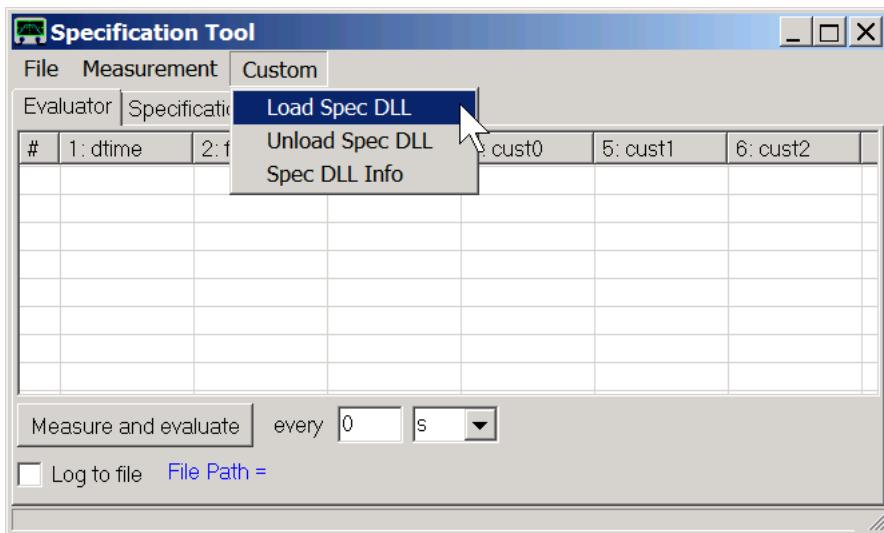
- TIME** Logs the time and date when the sweep ended.
- DTIME** Logs the time when the sweep ended in seconds after the "Measure and evaluate" button was pressed.
- DFREQ** Logs frequency meter for long time frequency stability measurements.
- CUST*** Custom data analysis as defined in a user provided external dynamic link library.

Custom defined data analysis in external dynamic link library:

By providing an external dynamic link library (dll), the user can define any conceivable analysis of the measurement data.

Moreover, the analysis value sent back to the specification tester need not be related to the measurement data at all, but could also be an environment value measured by means of the dll like environmental temperature, a voltage or a current. This is very useful for automatic temperature measurements of for e.g. measuring the C-V characteristic of a varactor diode.

The spec tester dll is loaded via the spec tester main menu:



Pascal/Delphi example dll source code:

Below example shows simple dll that accesses the measured sweep data and returns some values to the VNWA spec tester. In this example a console window is opened, which is useful during program development for outputting debug information using the writeln() command.

The VNWA spec tester accesses three functions:

_Init: is called when the dll is loaded the first time. Do one time initializations here, e.g. open the console window.
_Close: is called just before the dll is unloaded. Do cleanups here like releasing dynamic memory or closing the console window as shown in the example.
_Process: is called after every sweep when new measurement data is available. Note, that the values are returned as strings, thus any kind of data may be returned to be displayed in the table.

SOURCE CODE:

```

library SpecDLL;

uses
  Windows,
  SysUtils,
  Classes;

{$R *.res}

type
  TComplex=Record
    R: double;
    I: double;
  end;

  TFloatArray=array of double;
  TComplexArray=array of TComplex;

procedure _Init; export; cdecl;
begin
  FreeConsole;
  AllocConsole;
  writeln('UserDLL function "Init" called.');
end;

procedure _Close; export; cdecl;
begin
  FreeConsole;
end;

```

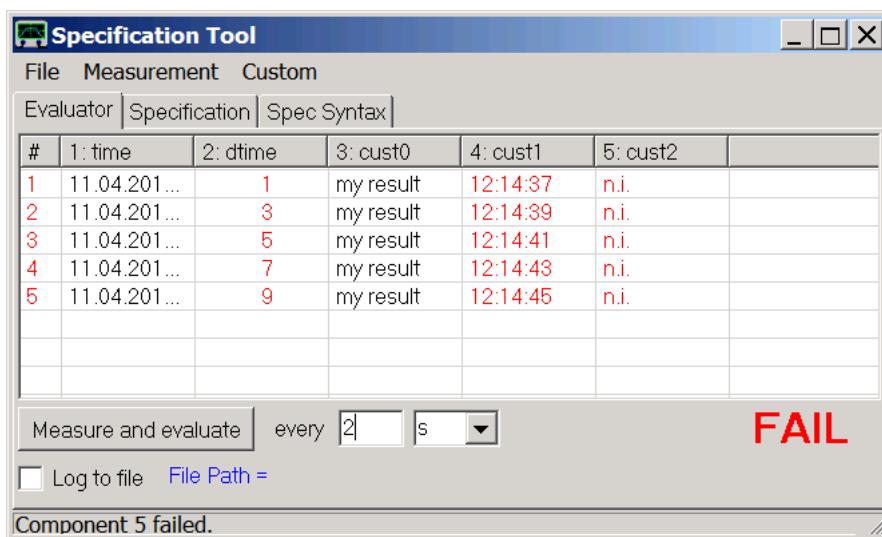
```

function _Process(Item: Integer; Data: PChar;
    freq: TFloatArray; Src: TComplexArray; Npts: Integer; dt: double;
    param1, param2, min, max: double): boolean;
    export; cdecl;
begin
writeln('UserDLL function "Process" called.');
// demonstration how to access VNWA data
writeln('Freq. range: ',freq[0],'...',freq[Npts-1], ' Hz');
writeln('Time per data point = ',dt, ' ms');
writeln('Number of data Points = ',Npts);
writeln('First data entry: ',Src[0].R, ' +* ',Src[0].I);
case Item of
  0:           //cust0
  begin
    StrPCopy(Data,'my result'); //return result of evaluation as string
    Result:=true;             //pass=true
  end;
  1:           //cust1
  begin
    StrPCopy(Data,timetostr(now)); //return anything you wish, e.g. temperature, voltage,....
    Result:=false;
  end;
else
  begin
    StrPCopy(Data,'n.i.');//cust2...cust9 not implemented yet, can be implemented as above
    Result:=false;
  end;
end;
exports _Init;
exports _Close;
exports _Process;

begin
end.

```

Below is an output of above example dll.



The screenshot shows the 'Specification Tool' application window. At the top, there's a menu bar with 'File', 'Measurement', and 'Custom'. Below the menu is a tab bar with 'Evaluator', 'Specification', and 'Spec Syntax'. The main area contains a table with five columns: '#', '1: time', '2: dtime', '3: cust0', '4: cust1', and '5: cust2'. The table has 5 rows, each containing a number from 1 to 5 followed by a timestamp and the string 'my result'. Below the table, there are input fields for 'Measure and evaluate' (set to 'every 2 s') and a 'Log to file' checkbox. A large red 'FAIL' button is prominently displayed. At the bottom, a message box says 'Component 5 failed.'

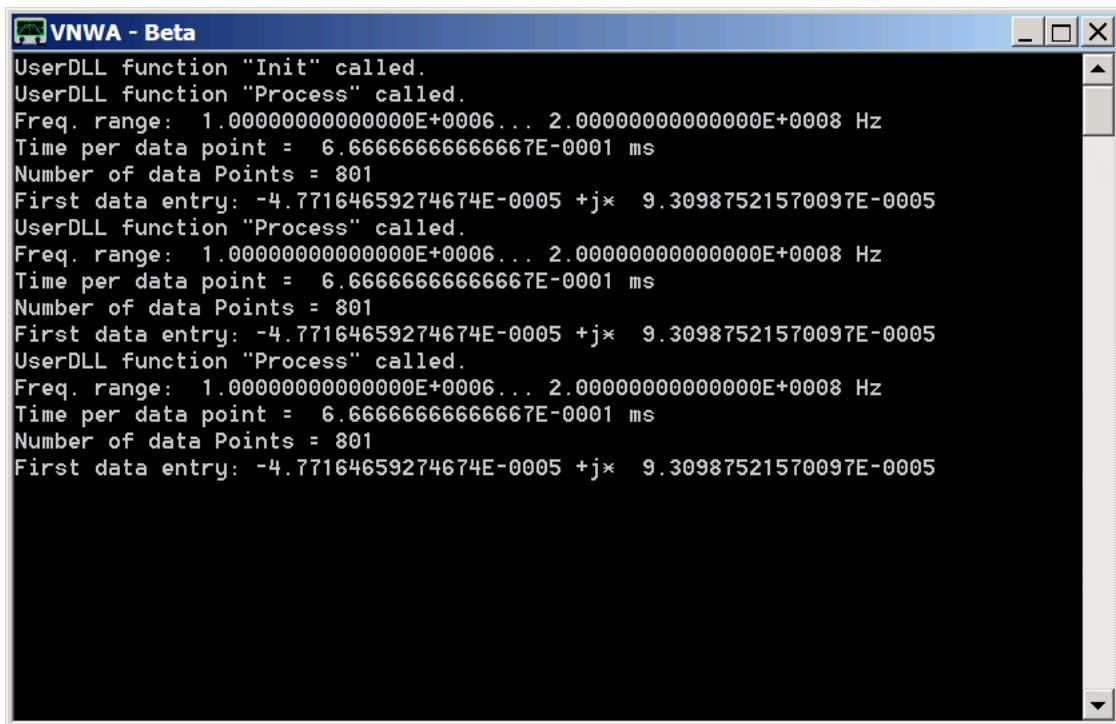
#	1: time	2: dtime	3: cust0	4: cust1	5: cust2
1	11.04.201...	1	my result	12:14:37	n.i.
2	11.04.201...	3	my result	12:14:39	n.i.
3	11.04.201...	5	my result	12:14:41	n.i.
4	11.04.201...	7	my result	12:14:43	n.i.
5	11.04.201...	9	my result	12:14:45	n.i.

Measure and evaluate every s

Log to file File Path =

Component 5 failed.

And this is the console output of above example dll after the first sweep:

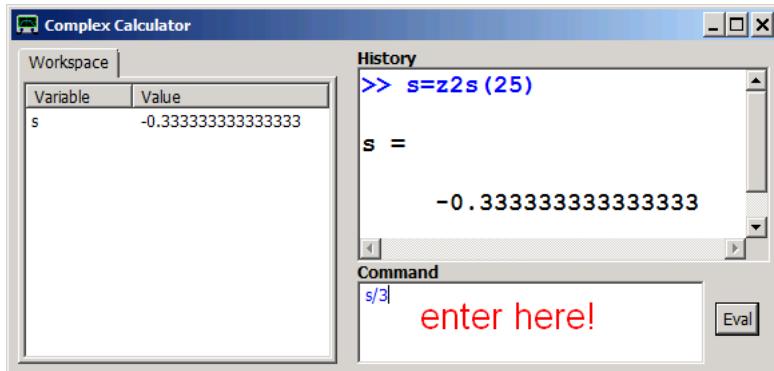


```
VNWA - Beta
UserDLL function "Init" called.
UserDLL function "Process" called.
Freq. range: 1.000000000000E+0006... 2.000000000000E+0008 Hz
Time per data point = 6.666666666667E-0001 ms
Number of data Points = 801
First data entry: -4.77164659274674E-0005 +j* 9.30987521570097E-0005
UserDLL function "Process" called.
Freq. range: 1.000000000000E+0006... 2.000000000000E+0008 Hz
Time per data point = 6.666666666667E-0001 ms
Number of data Points = 801
First data entry: -4.77164659274674E-0005 +j* 9.30987521570097E-0005
UserDLL function "Process" called.
Freq. range: 1.000000000000E+0006... 2.000000000000E+0008 Hz
Time per data point = 6.666666666667E-0001 ms
Number of data Points = 801
First data entry: -4.77164659274674E-0005 +j* 9.30987521570097E-0005
```

TOOLS - COMPLEX CALCULATOR

This is a tool for **evaluating complex expressions** written by Simon Bucheli.
Thanks for allowing me to integrate this tool into VNWA!

The tool basically works like a pocket calculator, but it can do calculus with **complex numbers**.



Expressions are being entered into the bottom right field as indicated by the red hint above.
The history of entered expressions and calculation results is shown in the top right field.
All previously defined variables and their values can be seen in the workspace on the left.

►► Hint: The tool can **access data in the VNWA data spaces** (S11,S21,...,Mem1..Mem10...) and **marker frequencies and times**.

Usage

Evaluation of simple expressions

sqrt(2)
e^(j*pi)+1

Definition of simple variables (immediate evaluation) - use "="

x = 1
a = 1; b = 2; c = 3

Definition of expression variables - use ":"=

x1 := a+b
x2 := a-b

Implemented Constants

e Euler constant = 2.71828...
i imaginary unit
j imaginary unit j=i
gamma Euler-Mascheroni-constant = 0.57721...
pi = 3.14159....
mil = 1 imperial MIL=1 inch/1000 in meters
inch = 1 imperial inch in meters
foot = 1 imperial foot in meters
yard = 1 imperial yard in meters
mile = 1 imperial mile in meters
c0 = vacuum speed of light in meters per second
kb = Boltzmann constant in Joules per Kelvin
neper = ln(10)/20, conversion Neper to dB
sa = spectrum analyzer mode offset level in linear scale

Predefined Variables

x1...x10 Optimizer variables from Optimizer Tool

Implemented Functions

abs
arcsin
arccos
arctan
arccot
arg or **Arg** *example:* `arg(exp(j*5))=5`
cos
cot
conj
ceil
deg conversion radians to degrees
db or **dB** $dB(x)=20*\log(abs(x))$
exp
floor
heaviside
im *example:* `im(2+3*j)=3*j` (unusual definition of the imaginary part)
imag *example:* `imag(2+3*j)=3` (usual definition of the imaginary part)
J0 Bessel function of the first kind of order 0
J1 Bessel function of the first kind of order 1
lb logarithm base 2
lg logarithm base 10
ln natural logarithm (base e)
log logarithm base 10
mag $mag(x)=abs(x)$
marker or **Marker** or **m** or **M** *example:* `marker(3) = frequency [Hz] of marker 3`
mval<n> *example:* `mval(1) = value of marker 1 for trace 2`
mem1 or **Mem1** *example:* `mem1(200e6) = value of data space Mem1 @ 200 MHz`
...
mem10 or **Mem10**
plot1 or **Plot1** *example:* `plot1(200e6) = value of data space Plot1 @ 200 MHz`
...
plot10 or **Plot10**
round
rad conversion degrees to radians
re *example:* `re(2+3*j)=2 = real part`
sin
sqrt
sqr
sign
s21 or **S21** *example:* `S21(10e6) = value of data space S21 @ 10 MHz`
s11 or **S11** *example:* `S11(m(2)) = value of data space S21 @ marker 2`
s12 or **S12**
s22 or **S22**
s2t converts a reflection coefficient normalized to 50 Ohms to a transmission coefficient
s2y converts a reflection coefficient normalized to 50 Ohms to a complex admittance. *example:* `s2y(0) = 0.02`
s2z converts a reflection coefficient normalized to 50 Ohms to a complex impedance. *example:* `s2z(0) = 50`
ss2t converts a shunt reflection coefficient normalized to 50 Ohms to a transmission coefficient.
t2s converts a transmission coefficient to a reflection coefficient normalized to 50 Ohms
t2ss converts a transmission coefficient to a shunt reflection coefficient normalized to 50 Ohms.
t2zs converts a transmission coefficient to a shunt impedance.
tan
time1 or **Time1** *example:* `time1(1e-6) = value of time trace 1 @ 1us`
...
time6 or **Time6** *example:* `time6(m(3)) = value of time trace 6 @ time marker #3`
y2s converts a complex admittance to a reflection coefficient normalized to 50 Ohms. *example:* `y2s(0) = 1`
z2s converts a complex impedance to a reflection coefficient normalized to 50 Ohms. *example:* `z2s(0) = -1`
zs2t converts a shunt impedance to a transmission coefficient.

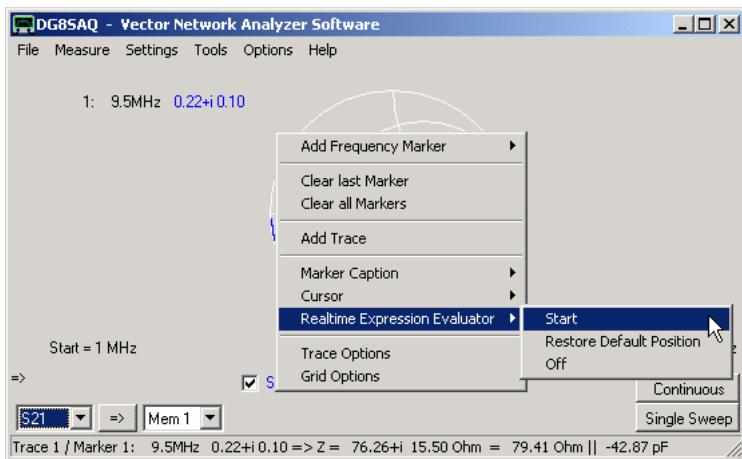
TOOLS - REALTIME EXPRESSION EVALUATOR

The **Realtime Expression Evaluator** can serve as a customizable marker.

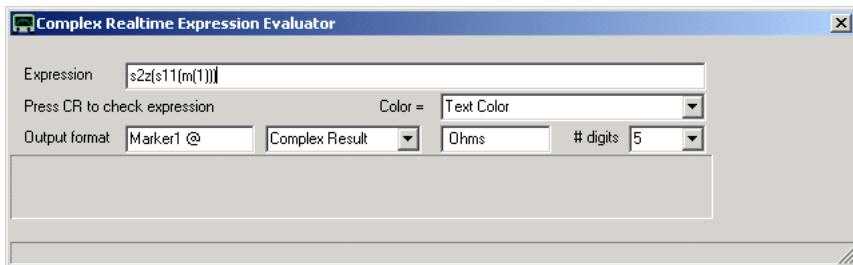
Example 1: A Smith chart marker showing impedance instead of reflection coefficient

You look at S11 in a Smith chart and you would like to have a marker showing impedance instead of the reflection coefficient:

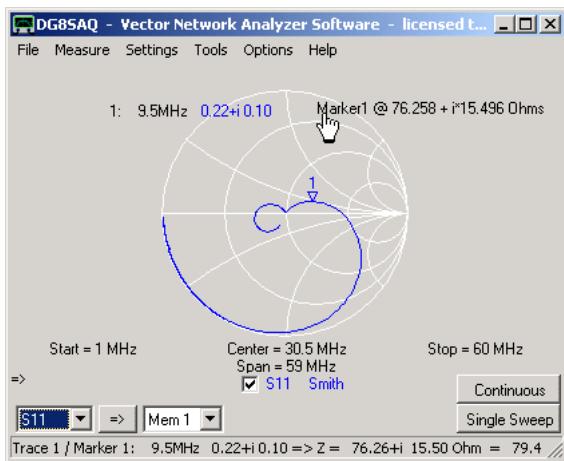
Select main menu "**Tools-Realtime Expression Evaluator**" or right-click onto the main graphics screen and select menu item "**Realtime Expression Evaluator - Start**":



The following expression evaluator window will open:



Here, you can customize a marker text to your needs. Enter the mathematical expression to be evaluated into the Expression field. In above example S11 is evaluated at the marker 1 frequency and converted to a complex impedance. Pressing enter will parse the expression. In case of expression errors, these will be displayed in the grey blank rectangle below the controls. Note, that the expression syntax is that of the complex calculator tool. With the controls below the expression, you can customize the way, the result will be displayed. Close the window when done. With above settings a new label appears on the main screen showing the impedance at marker 1 position (see mouse pointer below):

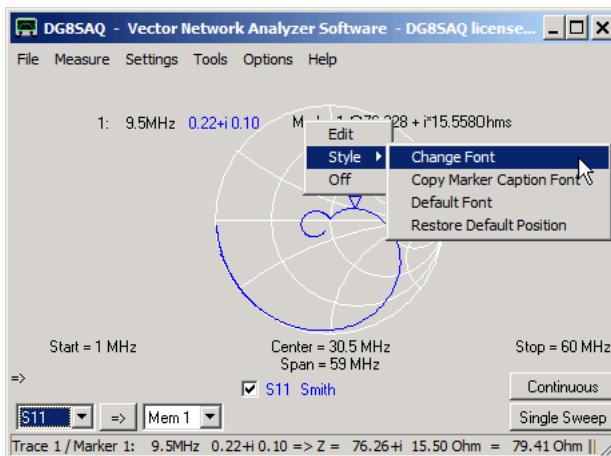


Note, that the displayed impedance values change when you move marker 1.

Note, that the default position of the label is top left, but the label can freely be moved with the mouse.

Note, that you can reopen the expression evaluator window by double-clicking the label.

Note, that some functions can also be accessed by right-clicking the label:



Edit will open the expression evaluator window.

Style will allow to customize the label:

- **Change Font** will invoke the font manager allowing to change font, style and size.

- **Copy Marker Caption Font** will transfer the font settings of the marker caption to the realtime expression evaluator label such that the two will look alike.

- **Default Font** will restore the default text font, style and size

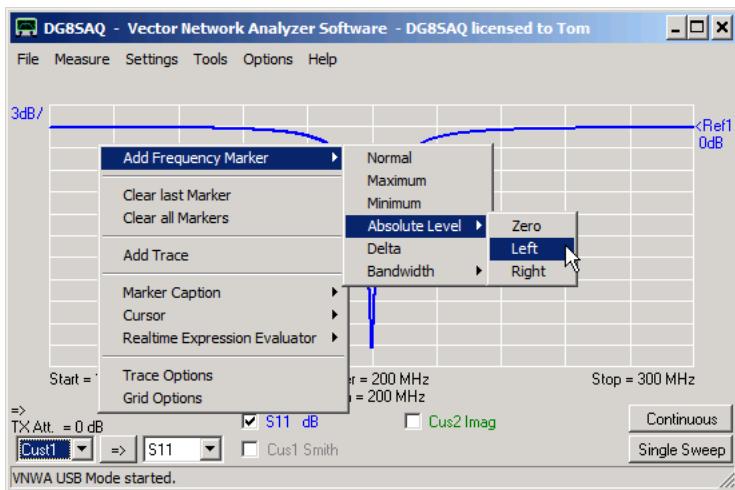
- **Restore Default Position** will move the label to the top left default position.

Off will switch the realtime expression evaluator function off.

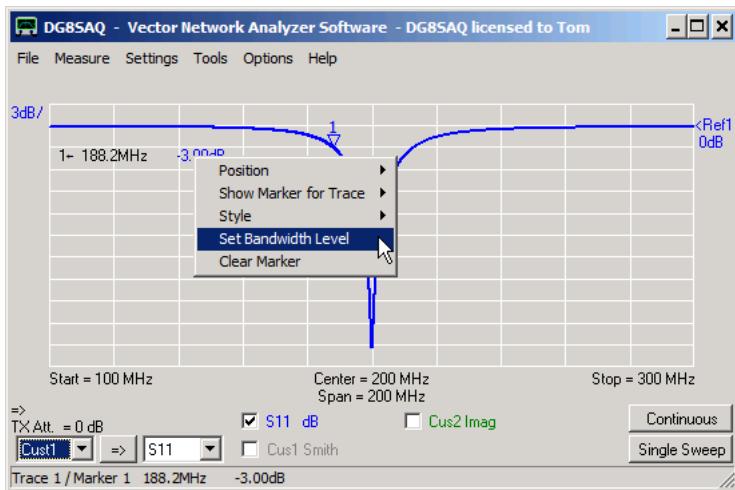
Example 2: Determining a resonator's Q-value from two bandwidth markers

This example demonstrates how the realtime expression evaluator can be used to determine the Q-value of a resonance notch by comparing the -3dB notch width to the notch frequency in real time.

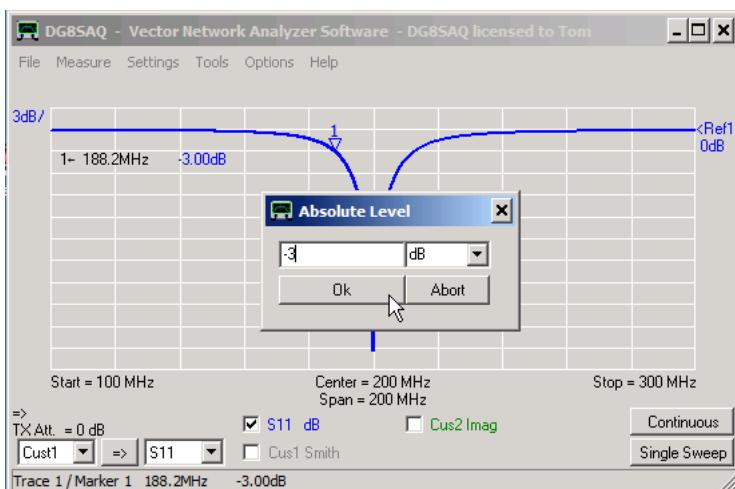
In order to determine the notch width we need two markers to the left and to the right of the notch at an absolute level of -3dB. We place the left marker by right-clicking on the main grid and selecting a left absolute level frequency marker:



We can set the marker level by either right-clicking the marker itself or the marker level label:

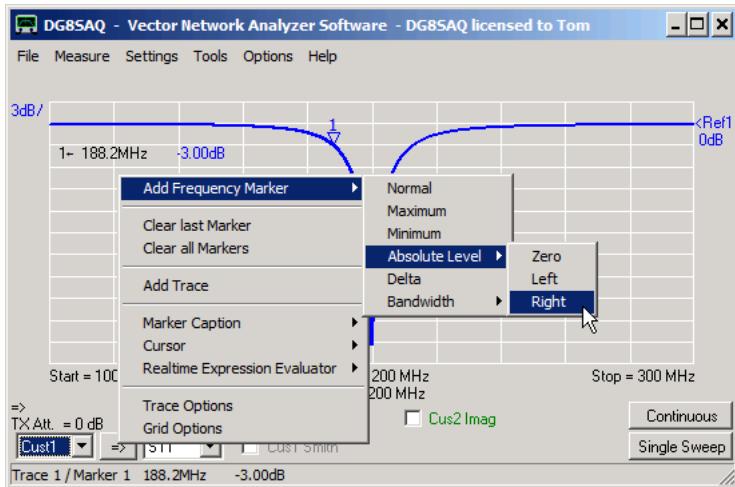


Upon selection of the "Set Bandwidth Level" menu item, the level mask pops up:

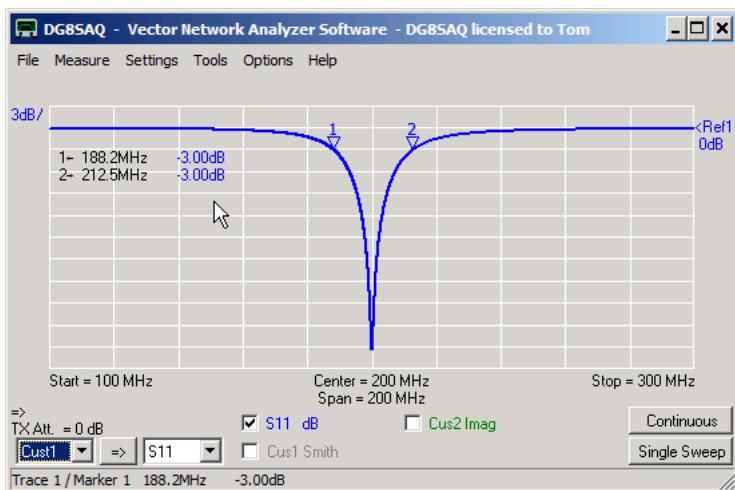


We enter -3dB and confirm.

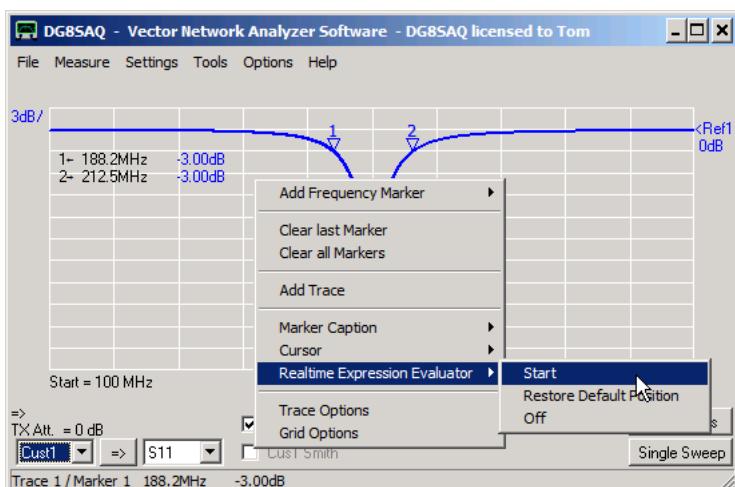
In a similar fashion we add a right absolute value frequency marker:



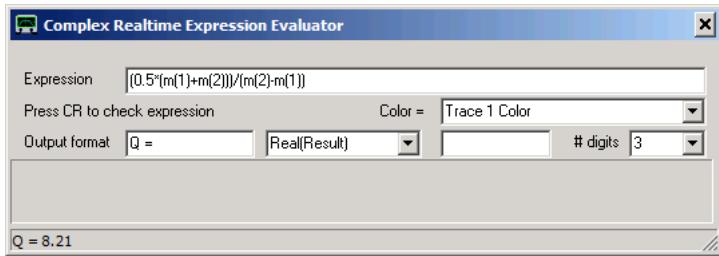
Here are the two markers we need for our analysis:



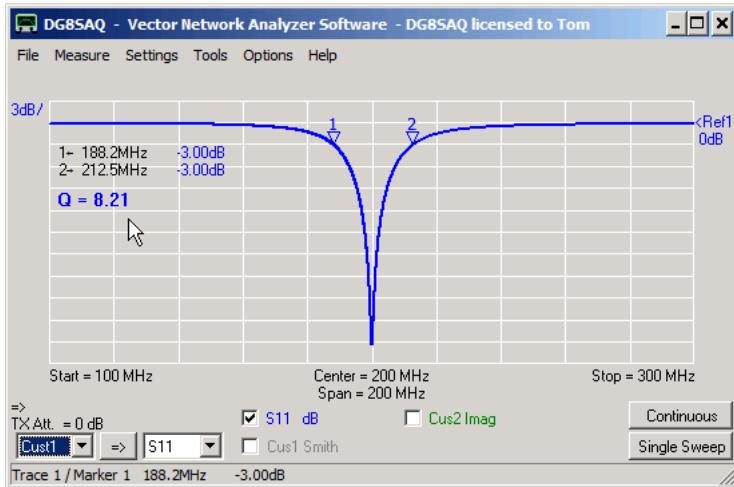
Next we start the real time expression evaluator:



The expression definition mask pops up. There, we enter the evaluation formula, where $0.5*(m1+m2)$ yields the notch center frequency computed from the two marker frequencies m(1) and m(2) and $(m2-m1)$ yields the notch width as calculated from the same marker frequencies. So, the result of the formula yields notch frequency divided by notch width, which is the Q-value.



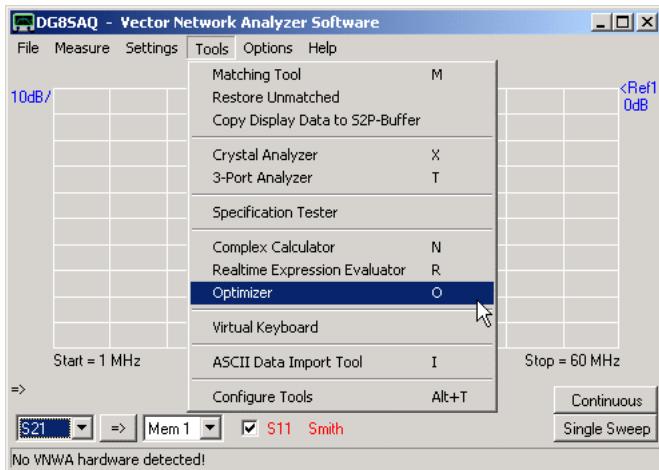
With above output format settings (display "Q=", display real part only, display no unit, 3 valid digits) we obtain the following result:



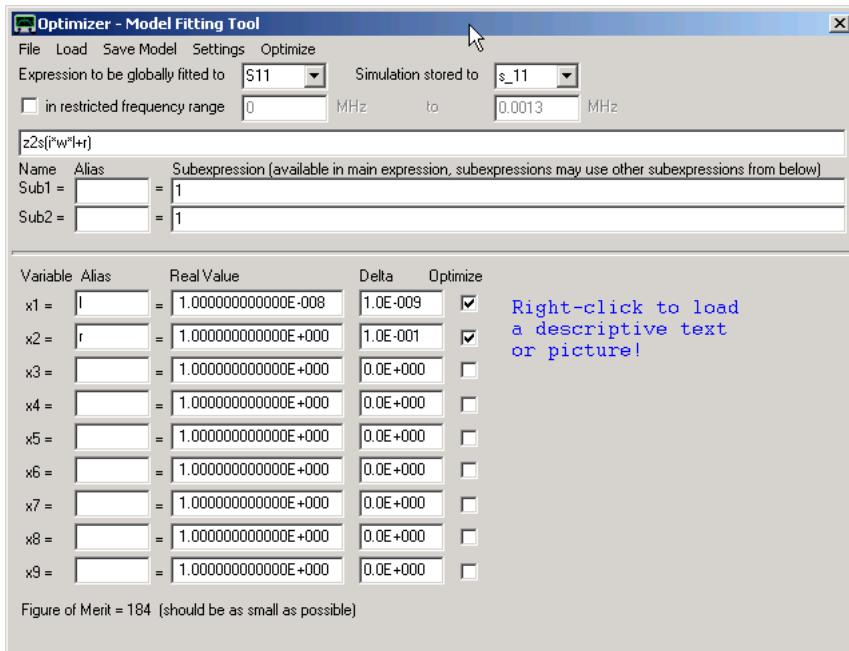
The computed Q-value of 8.21 is displayed and updated in real time as new measurement results come in. Note, that the Q-label like the marker labels can be dragged to any convenient position on the VNWA window.

TOOLS - OPTIMIZER

VNWA version 36.5.0 and up has a built-in least squares fit optimizer tool suitable for model fitting:



Upon first start, the optimizer tool presents itself like this:



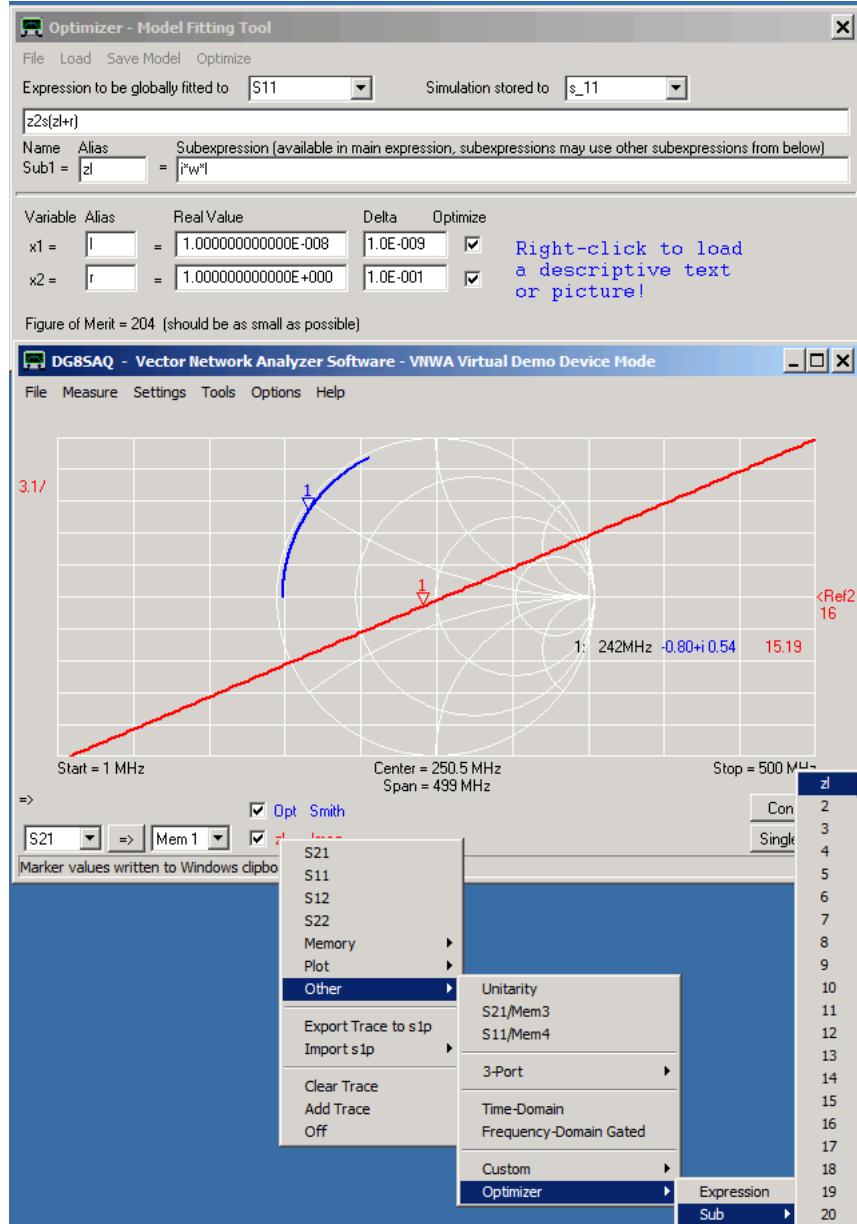
- The optimizer tool includes a complex valued function parser with syntax identical to the custom trace expressions syntax, which is used to define a fit function in the input line above the sub-expression block.
- The fit function may be split into up to twenty subexpressions which may be named descriptively using aliases for the default names Sub1...Sub20.
- All expressions may contain up to 10 real valued variables to be optimized (x1...x10), which may again be renamed descriptively.
- Initial real values have to be assigned to the variables as well as initial step sizes (Delta) for the optimizer.
- For manual optimization, the values can be changed in Delta steps using the mouse wheel. Upon changing the variables the model will be recalculated.
- The optimizer can be informed to keep a variable constant by setting the Delta value to zero or better by unchecking the Optimize checkbox.
- From software version 36.7.4 on the optimizer can be instructed to perform the optimization over a restricted frequency range only.
- A descriptive text or a jpeg or bitmap picture may be loaded by right-clicking the picture/text area. A picture may also be loaded via menu "Load"- "Picture".

- The optimizer tool also allows to save or load a model via the File menu.
- The first line in the optimizer tool informs the optimizer to fit the model to resemble S11 as much as possible with above settings.
- The model simulation is selected to be stored in 3-port data space s_11 with above settings (choices s_11, s_21, opt=separate storage).
- In menu "Settings" the optimizer may be instructed to update the main graphs after each optimizing step in real-time. This may be instructive as one can observe how the optimizer comes to the final solution but it does slow down the optimizing process considerably.
- The menu item "Optimize" initiates the optimizing process.

Note, that pressing Return on any input field will invoke recompilation and recalculation of all expressions.
 Note that unused subexpressions may be hidden by moving the separator bar up (see mouse arrow above).
 Note that subexpressions may be moved up or down using the right-click menu of the Sub*-labels.
 Note, that unused variables may be hidden by decreasing the window height.

Note:

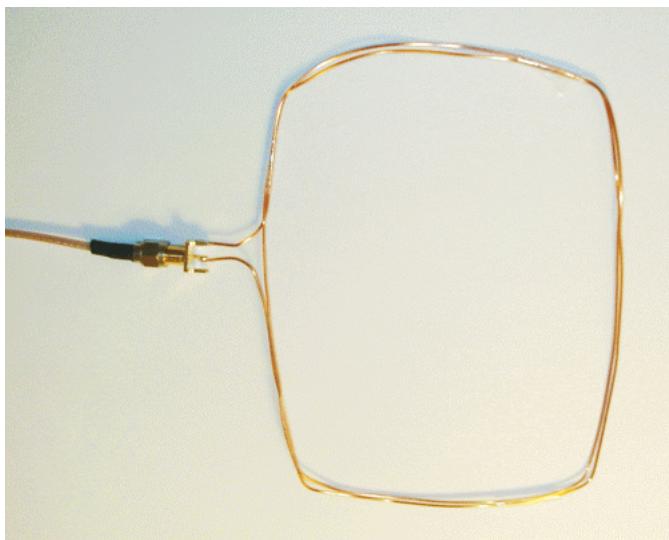
The optimizer main expression as well as all sub-expressions may be directly plotted in the main window:



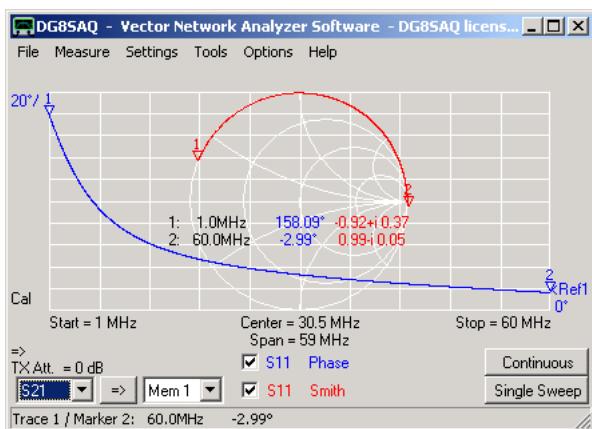
Above, we display the optimizer expression in trace 1 and sub-expression 1 with alias z1 in trace 2. Observe, that the sub-expressions are shown with their alias names if assigned.

Example

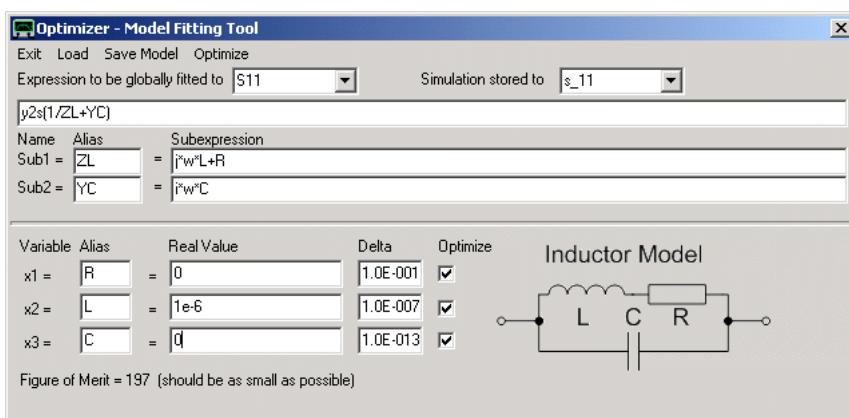
We want to find an equivalent circuit model for the following loop antenna:



The measurement reveals that the loop is basically an inductor with rather low loss which becomes slightly capacitive at high frequencies, though:

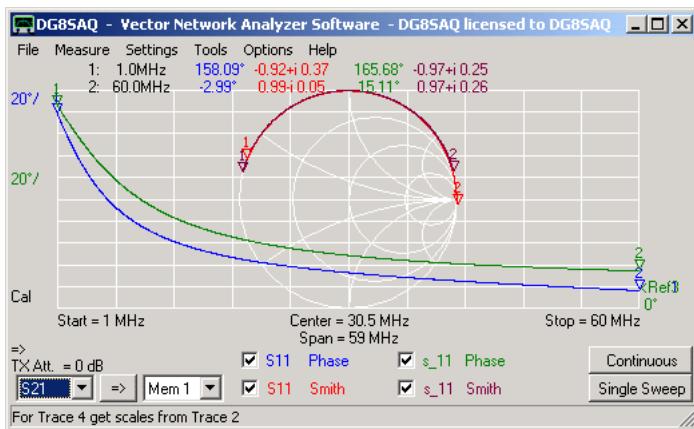


We thus try to simulate the antenna using a model consisting of an ideal inductor with a series resistor. In order to model the capacitive behavior at high frequencies, there must be a small capacitance in parallel to the lossy inductor:

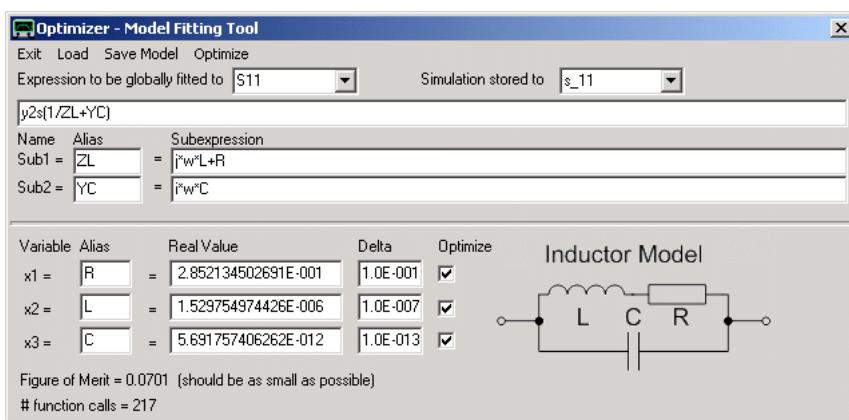


The predefined variable w denotes the angular frequency.

Since we have instructed the optimizer to store the simulation into 3-port data space s_11 , we may compare our measurement to our initial guess for the model parameters:



As the match is not perfect yet, we instruct the optimizer tool to fit our model to the S11 measurement by pressing the optimizer menu "Optimize". The optimizer then yields the following optimum variable values:



i.e. rounded to two digits

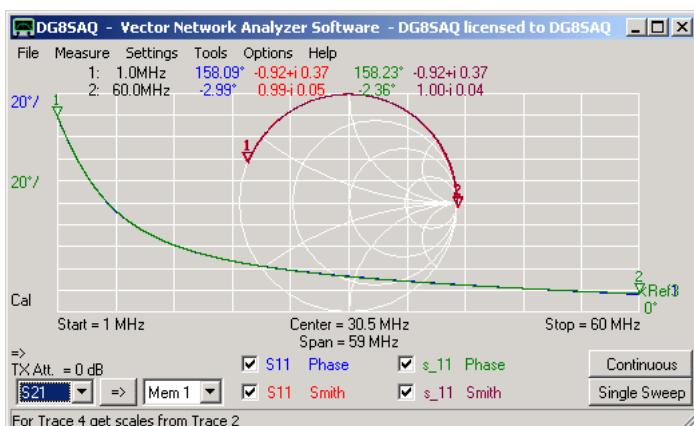
R=0.29 Ohms

L=1.5uH

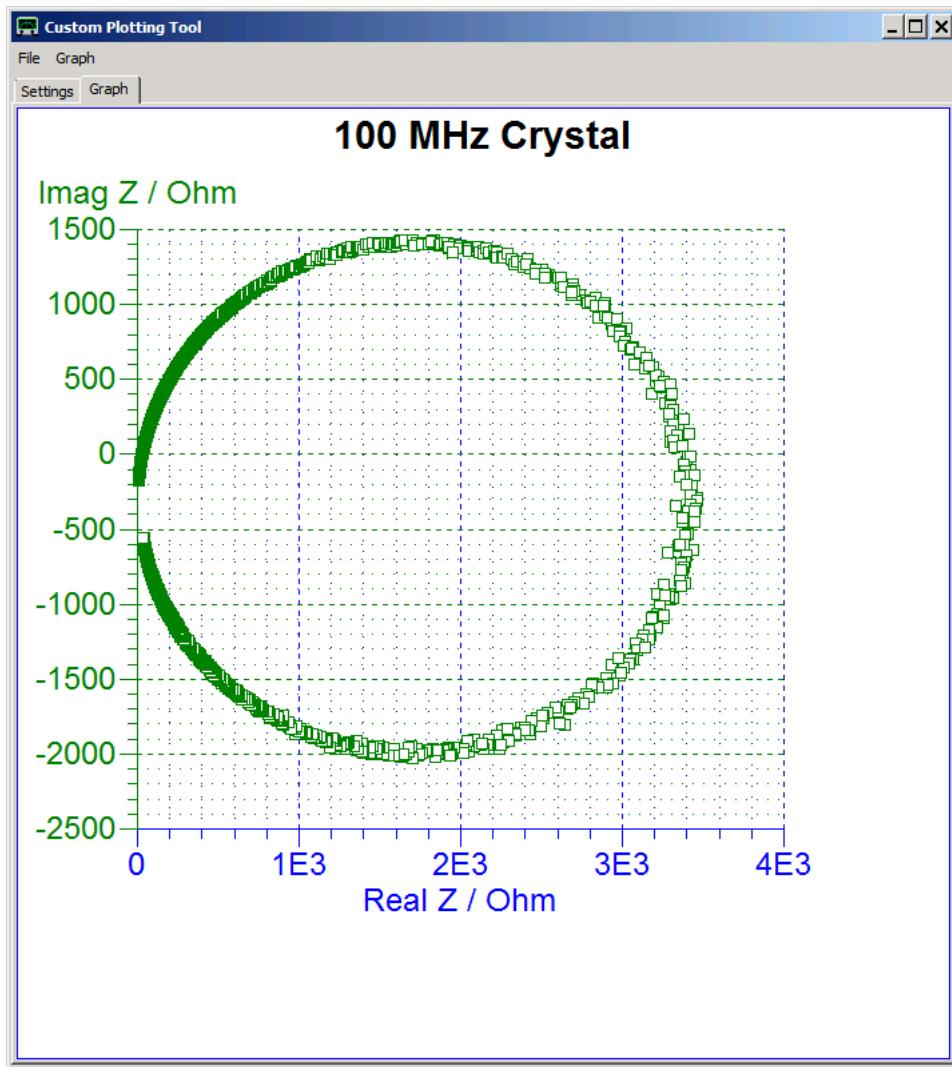
C=5.7pF

These values intuitively make sense.

Comparing the optimized model to the measurement we find an almost perfect match:



TOOLS - CUSTOM PLOTTER



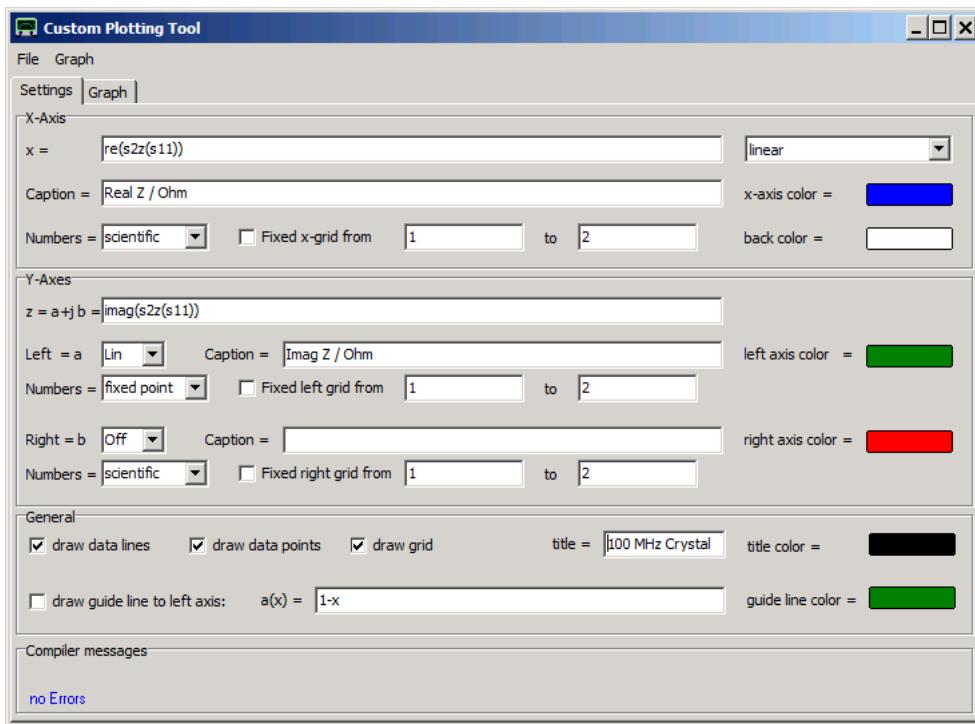
The **custom plotting tool** allows to generate high quality highly customizable plots of calculated, measured or processed data.

The x- and y axes are freely configurable with respect to displayed data, lin/log scaling and number format.

Data may be plotted with polygons, point symbols or a combination of both.

The custom plotting tool is invoked via main menu "**Tools - Custom Plotter**" or via the keyboard short-cut **ctrl+alt+P**.

The graph is configured on the **Settings** tab. The following settings produce above graph, S11 contains measured reflection data of a 100MHz crystal:



X-Axis Panel:

The point grid of the plot is identical with the point grid of the VNWA measurement settings. Thus, the number of plotted points is identical with the number of measurement points per sweep.

The x-data is specified with a custom formula. The syntax and available functions are identical with those for custom traces. Note that the calculus is performed with complex numbers.

In above example, the x axis will plot the real part of the impedance of the measured S11 data in Ohms, which is also indicated in the **Caption** field. We use a linear x-axis.

Note, that linearly measured data can also be plotted on a logarithmic scale here if desired.

The requested number format is scientific, i.e. 1000 will be displayed as 1E3. If we are unhappy with the autoscale function, we may specify an x-range and activate a fixed x-grid.

Y-Axes Panel:

We may have two independent y-axes, one on the left and one on the right. The real part of the complex custom expression for z is plotted to the left y-axis, while the imaginary part is plotted to the right.

Note, that the z-expression may contain the variable **x** in addition to the functions, variables and constants defined for custom traces.

The variable **x** denotes the value calculated with the custom-expression for the x-axis.

Example: Say, you want to plot the function $\sin(x)$ in the x-range from 0 to 10. Then you use the x-expression $x = n/n_top*10$ and the z-expression $z = \sin(x)$.

Here, n will run from 0 to n_top , which is the sweep settings number of points-1. Thus, the x-expression will run from 0 to 10 and the variable x simply accesses this value.

Both y-axes can be switched to a linear scale, a logarithmic scale or completely off independently. As for the x-axis, captions, fixed y-ranges, number formats and colors can be specified.

General Panel:

Here, the plotting style is being specified. A polygon may be drawn from data point to data point (**data lines**) and/or individual data points may be plotted as little squares.

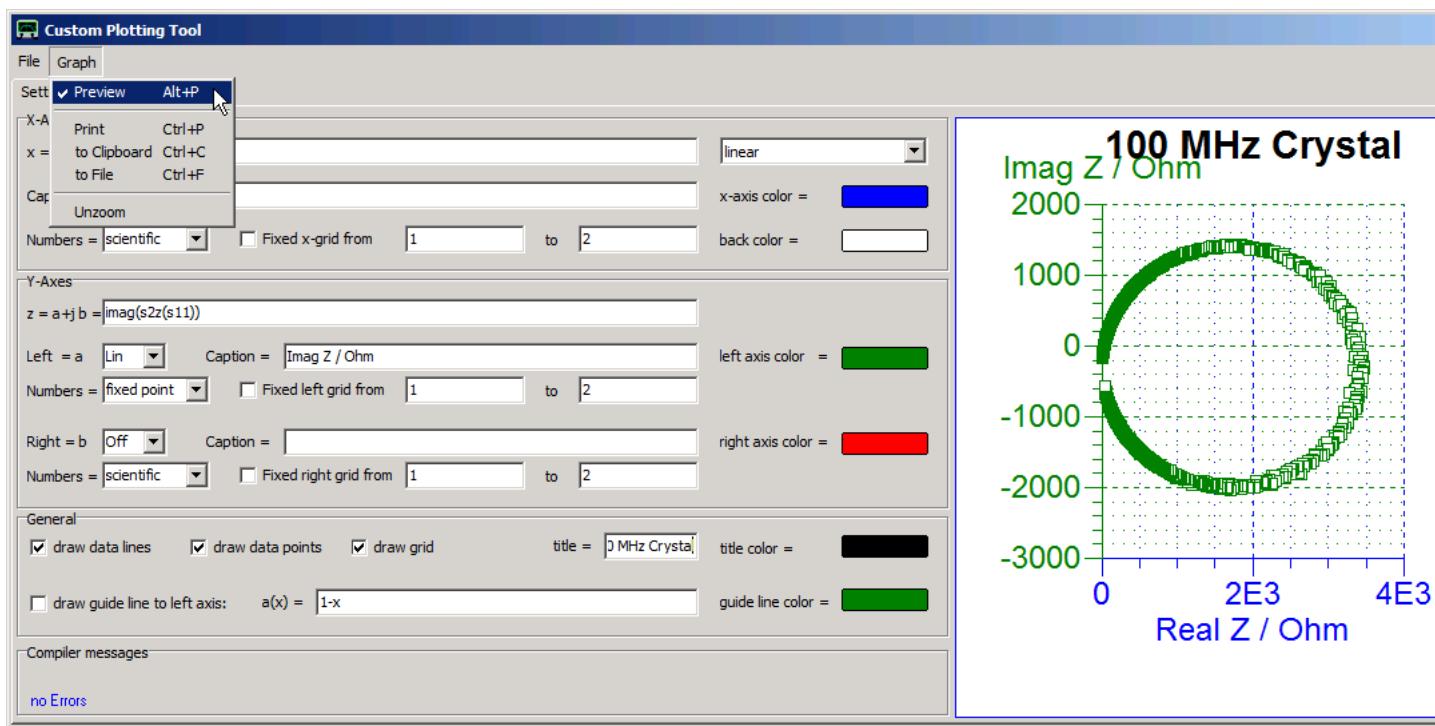
Also, the coordinate grid lines can be switched on or off here and we can give the graph a title which will appear above the graph.

Moreover, a so called guide line can be specified with another custom expression $a(x)$. This guide line can be any function of x. It will be drawn with respect to the left y-axis and it will always be drawn as a line, not a set of symbols.

Compiler messages Panel:

The custom expressions will compile upon pressing the return key. In case of a syntax error, the corresponding expression field turns from white to red and a descriptive error message will appear on the compiler messages panel.

It is very useful to observe the effects of changing settings on the graph. Therefore, there is a preview function, that produces a little copy of the graph on the right hand side of the settings panels:



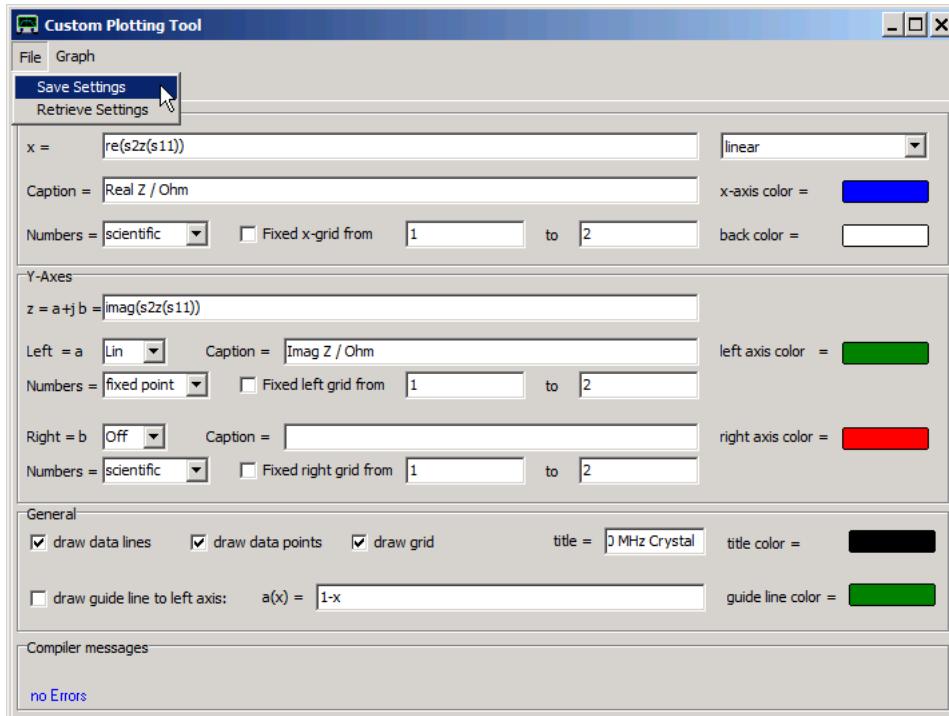
The preview mode is switched on or off via the menu "**Graph-Preview**". The Graph menu also allows to print the graph, copy it to the clipboard or a file and do an unzoom when in box zoom mode (see below).

Note that you can change size of the preview graph by dragging the boundary between the input panels and the graph with the mouse.

All the settings will be saved and restored when closing and relaunching the VNWA application. They will also be saved in an instrument state.

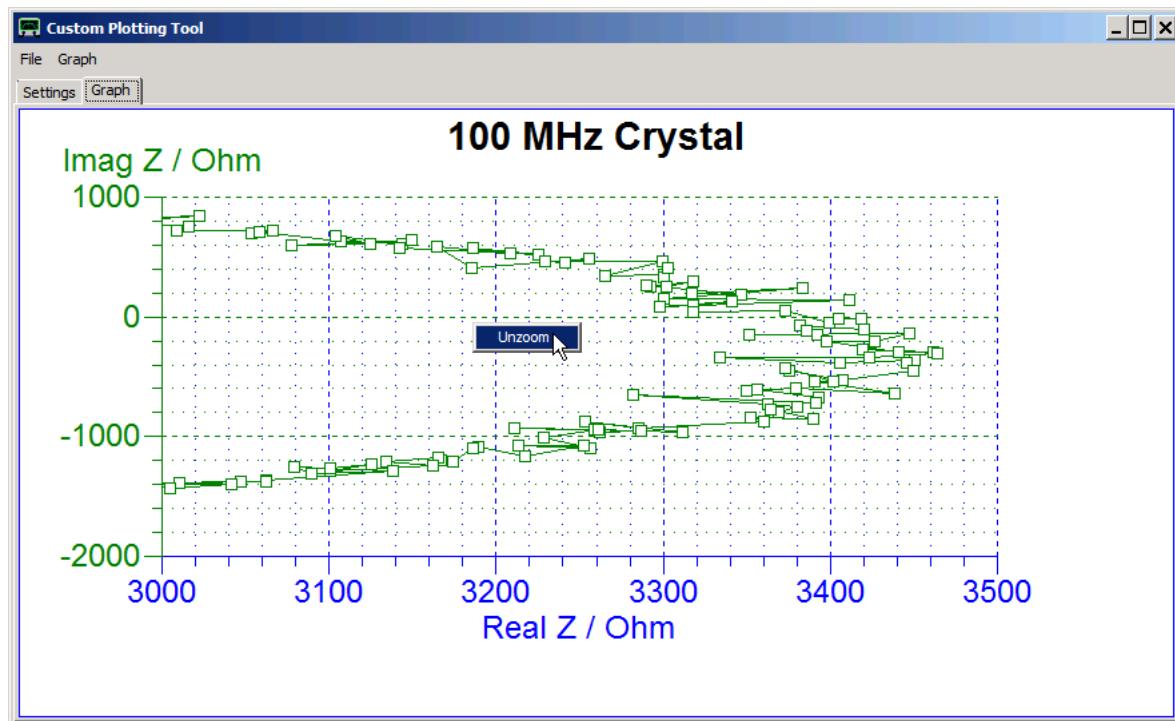
But the graph settings can also be saved into an individual graph setting file and be reloaded from there. Note, that the data points will not be saved this way.

This is done in the **File** menu:



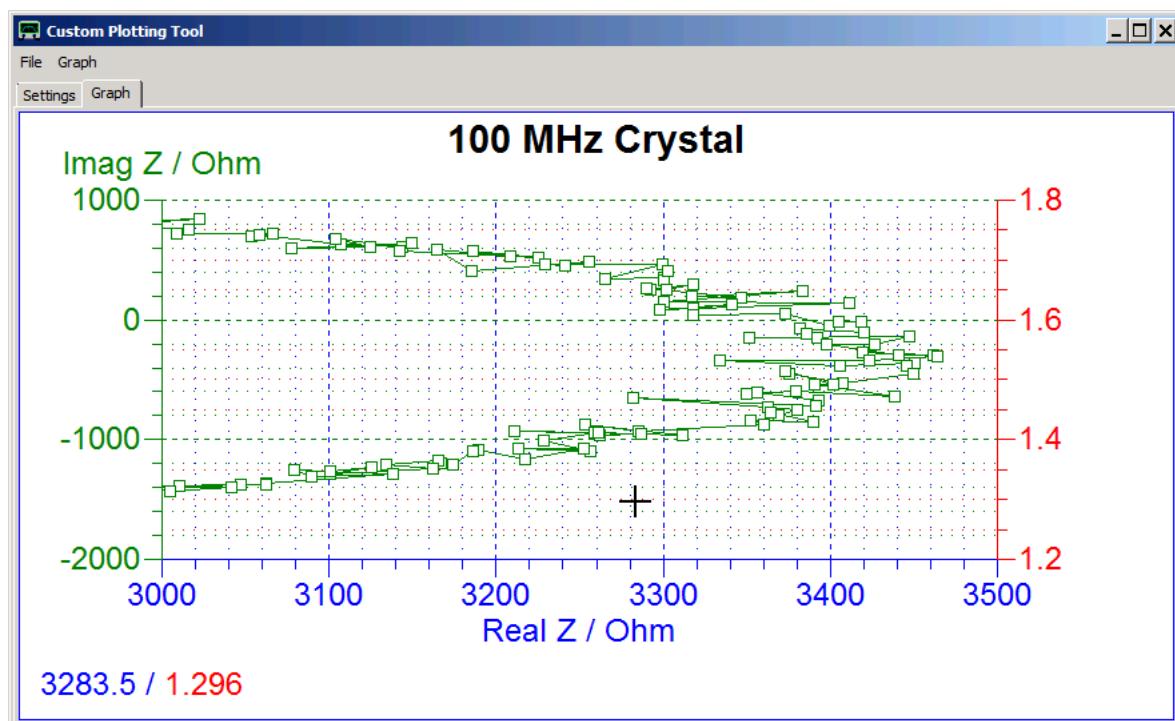
On the main plot (NOT on the preview plot) a box zoom can be performed. This is achieved by pressing the ctrl-key, then holding down the left mouse key and drawing a box around the area of interest.

First release the mouse key and only then release the ctrl-key to achieve the zoom:



To unzoom, right-click into the plot and select the unzoom menu as shown above or use the menu **Graph-unzoom**.

The main graph also offers a cursor for obtaining coordinates from the graph. When pressing the left mouse key, a little cursor cross will replace the mouse pointer and the cross coordinates will be shown on the lower left. The y-coordinate refers to the left y-axis. If the coordinates with respect to the right y-axis are desired, then while holding down the left mouse key, the right mouse key must be pressed additionally:



►► **Hint:** If different colors are used for the axes, it is very easy to see which marker coordinate corresponds to which axis.

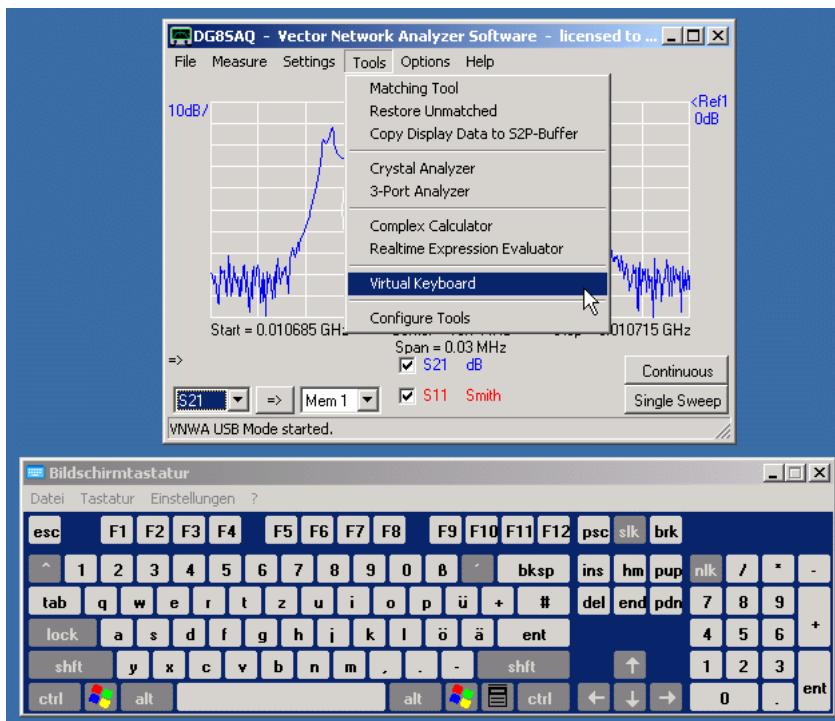
▶▶▶ **Note:** When measurement data is displayed, the graph will be updated in real-time while new data is being measured.

TOOLS - VIRTUAL KEYBOARD

32 bit Windows Systems

Windows has a built-in **on-screen or virtual keyboard**, which can be operated with the mouse or via touchscreen in case no hardware keyboard is available or desired.

For 32 bit Windows versions this keyboard can directly be invoked from within VNWA using the main menu "**Tools - Virtual Keyboard**":



64 bit Windows Systems

For 64 bit Windows versions this menu item is made invisible as Windows does not allow the 32 bit VNWA application to launch the 64 bit keyboard application.

There are two possible workarounds to still use a virtual keyboard on a 64 bit Windows machine.

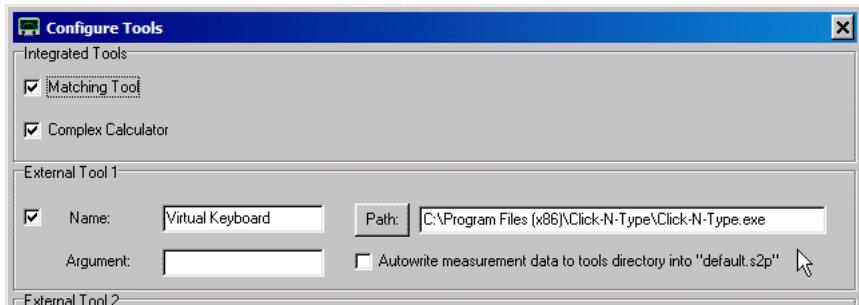
Solution 1: Launch the Windows virtual keyboard from outside VNWA

This is done by either typing "osk.exe" from a Windows command prompt (which is a challenging task without keyboard) or by having placed the osk application beforehand onto the desktop and double-clicking it.

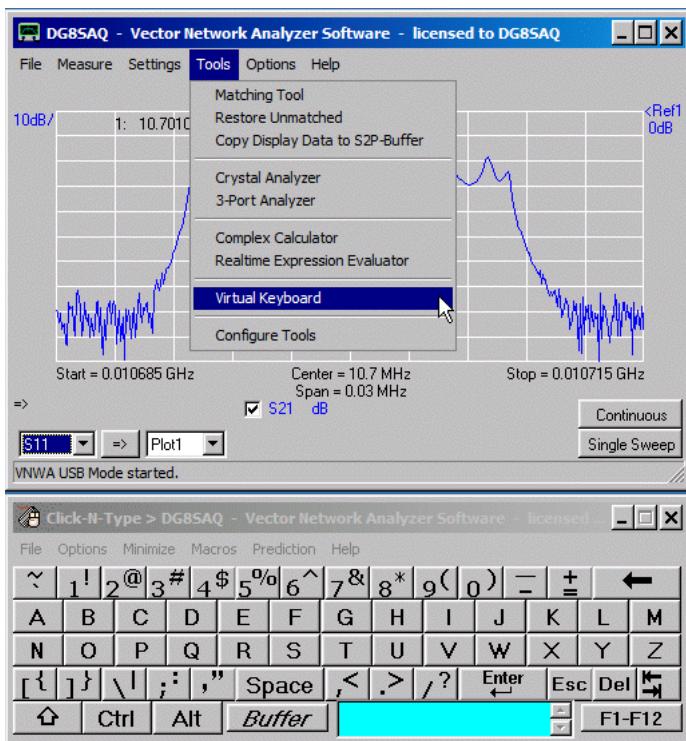
Solution 2: Using a 32 bit virtual keyboard application and adding it to the VNWA tools menu

In the following this approach is demonstrated using the free and powerful virtual keyboard software **Click-N-Type** which can be downloaded from <http://cnt.lakefolks.com/>.

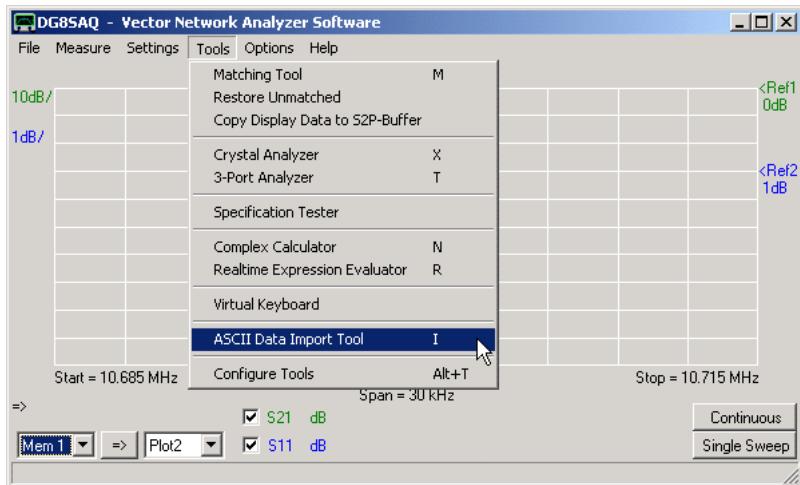
After having installed the Click-N-Type software, it can be integrated into the VNWA tools menu. To do so, open the main menu "Tools-Configure Tools" and add "Click-N-Type" to the configurable tools menu. Use the Path-button to browse for the Click-N-Type application on your disk and give it a descriptive name (here "Virtual Keyboard").



After closing the "Configure Tools" menu, a new submenu "Virtual Keyboard" will be available in the "Tools" main menu and, if selected, it will launch the Click-N-Type virtual keyboard:

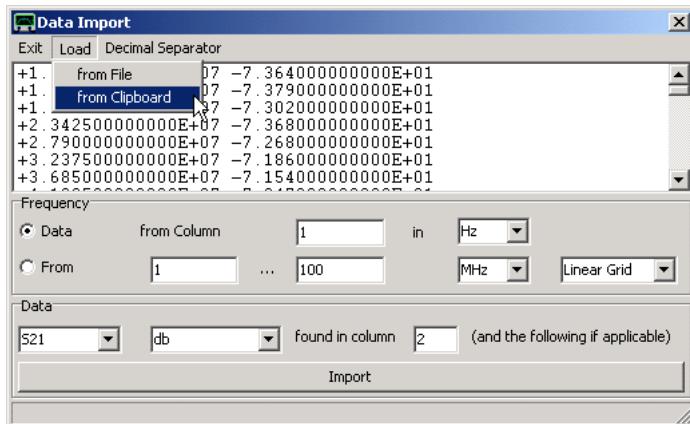


TOOLS - ASCII IMPORT TOOL



Often data measured with the VNWA is to be compared to third party data. If the latter is available in Touchstone snp format, then it can directly be read into the VNWA software.

If the third party data is only available as raw rows of ASCII numbers, then the **ASCII Import Tool** helps to read it. The import tool is either accessible via the "Tools" menu as shown above, via the keyboard short cut "i" or via the main menu "File"->"Import Data"->"Any ASCII Import".



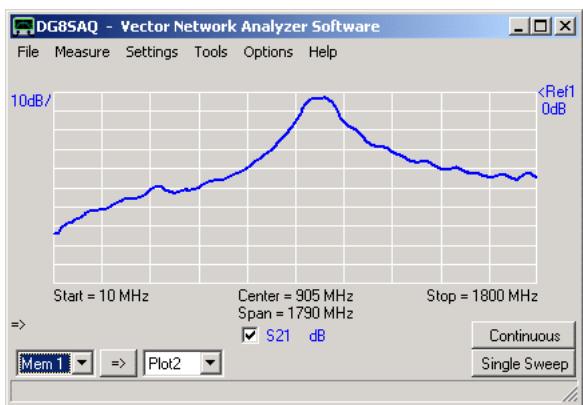
Data can be read from a file via "Load"->"from File" or by dropping the file via drag and drop onto the list field. Alternatively, data can be entered from the Windows clipboard.

Above data has been entered via clipboard/copy and paste from a HP spectrum analyzer and the HP benchlink software.

If the numbers use "," as decimal separator instead of a decimal point, the "Decimal Separator" menu allows to change the data to using a decimal point.

The list field shows the raw input data. Comment lines should be deleted by right-clicking them and selecting "delete".

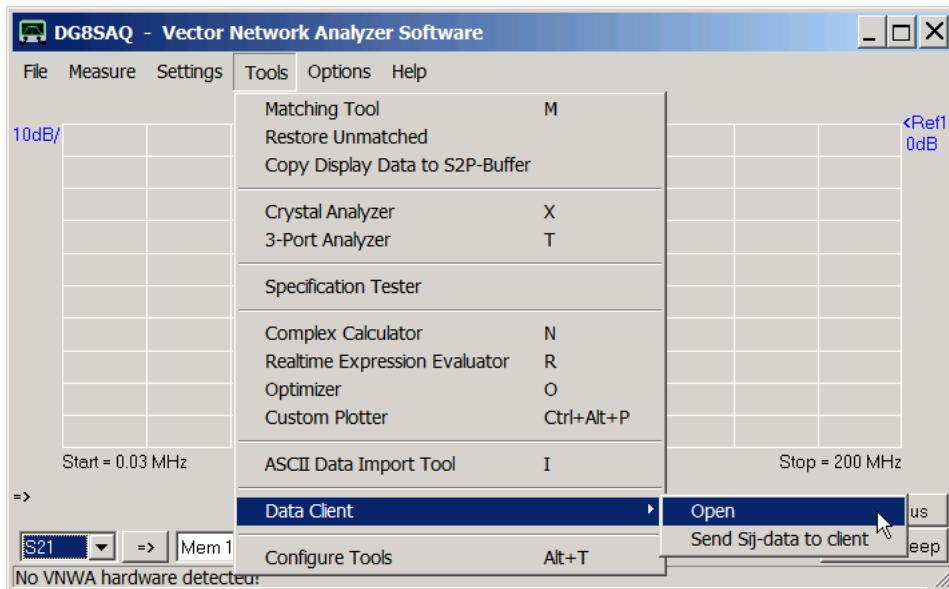
Below the raw data list, the user specifies which columns are to be read in what way and where the data is to be copied to. In above example, the first column is the frequency in Hertz and the second column is the attenuation in dB. If no column defining a phase is specified, then the phase is set to zero. By pressing the "Import" button, the data is copied to the S21 data field:



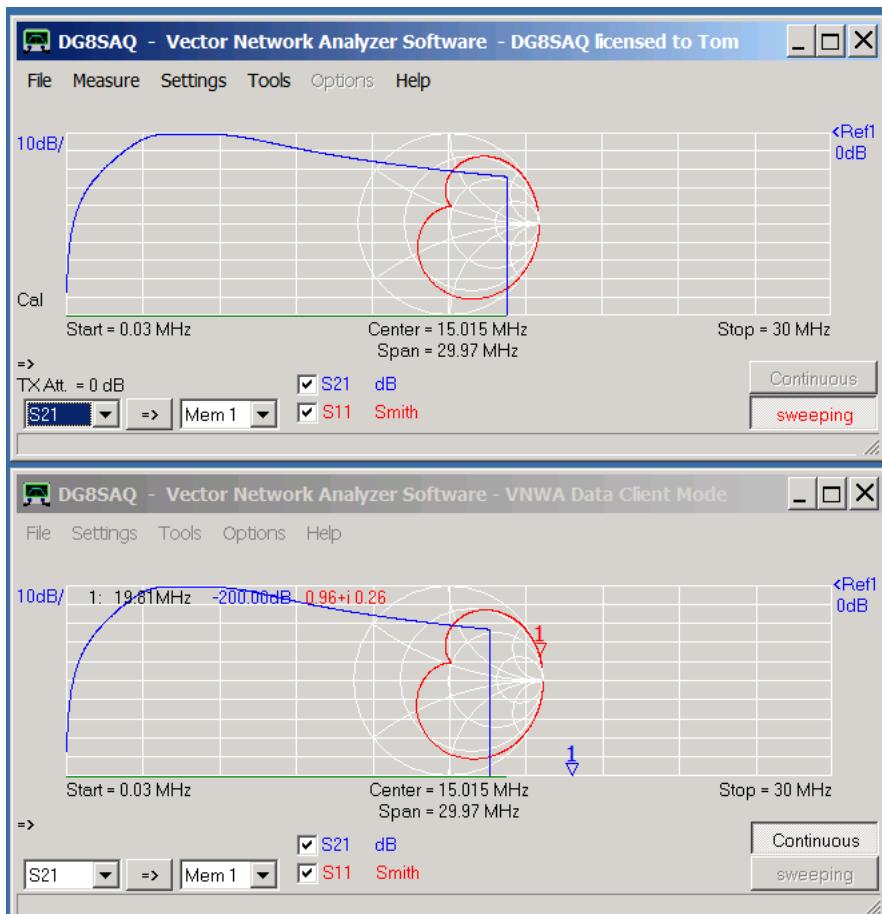
TOOLS - DATA CLIENT

A VNWA data client is a second VNWA software instance that serves as an additional display window to display measurement data.

It can be opened via The **Tools-Data Client-Open** menu:



When doing a sweep in the main instance window (=server), the measurement data will immediately be shown on the client instance window, too, as long as the client is in sweep mode:

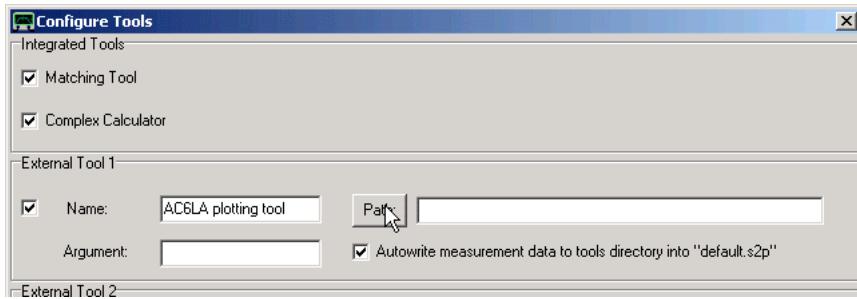


- ▶▶▶ **Hint:** Multiple data clients can be open at the same time.
- ▶▶▶ **Hint:** Clients may be in zoom mode, while the server is not.
- ▶▶▶ **Hint:** Server and client(s) may use different sets display settings and markers.
- ▶▶▶ **Hint:** The client(s) can be configured to receive data into Plot memory fields instead of default Sij data fields, thus allowing server and client(s) to display different frequency axes, e.g. the server sweeping a linear frequency axis but a client displaying the same data in a logarithmic frequency scale.
- ▶▶▶ **Hint:** When closing the server, all clients started from the server will be closed automatically. Note, that If more than one client is open, only one set of client display parameters will be memorized until the next launch of clients.

A detailed description of the data client functionality can be found on page [VNWA Data Client](#).

TOOLS - CONFIGURE TOOLS

The "configure tools" menu allows to **customize the tools menu** by showing or hiding the built-in tools and/or by **registering up to four external tools** to the "Tools" menu. An example is presented in section "Interfacing ZPlots".



External tools are characterized by four parameters which can be selected by the user:

Name: The name of the external tool that will be shown in the "Tools" menu.

Path: Path including the application file name of the external application. Valid applications are files and links that can be started in Windows by double-clicking.

Argument: An optional runtime argument that can be passed to an external tool, e.g. a file name for a text editor.

Autowrite: When the "autowrite" checkbox is activated, current measurement data will be written to the directory specified by "Path" as default.s2p prior to launching the external tool and also every time a sweep cycle is completed. This allows automatic data transfer to external tools.

►►► **Note:** You have to **activate an external tool by the checkbox** on the very left to make it visible in the tools menu.

OPTIONS

The VNWA main menu "Options" offers the following functions:

Select Instrument is only visible, if N2PK support is activated

Operation Mode

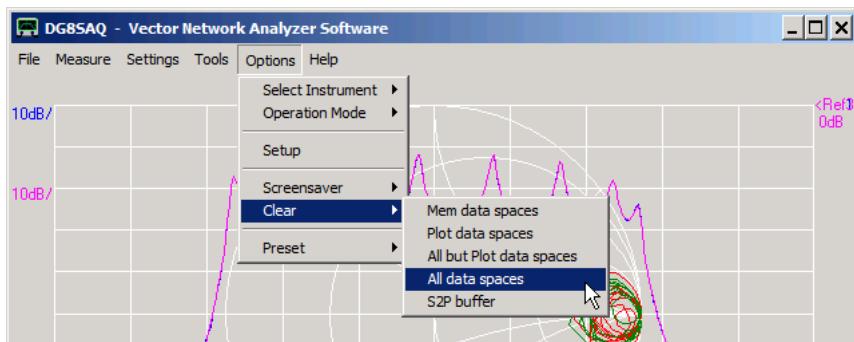
Setup (VNWA, default)

Setup (N2PK-VNA, only activated if N2PK instrument is selected)

Screensaver

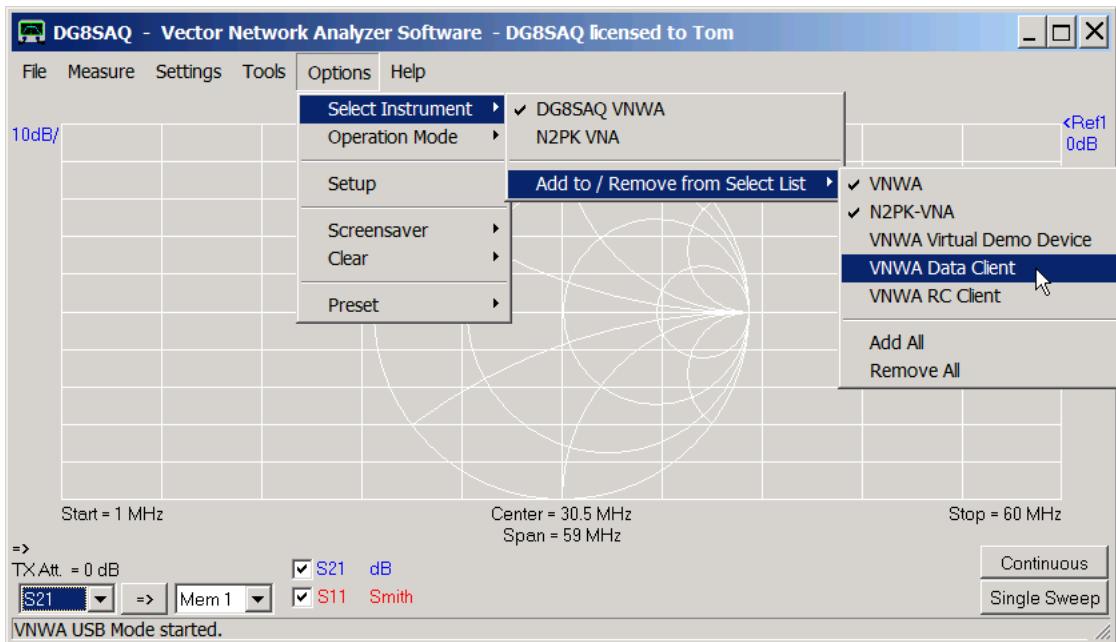
Clear allows to erase various combinations of data spaces. Erase means that traces will be nulled.

Preset



OPTIONS - SELECT INSTRUMENT

The menu item "Select Instrument" allows to select, which kind of hardware architecture the software VNWA.exe is to control:



Via the Add/Remove submenu instruments can be made visible or invisible in the select list. This is useful as VNWA36.7.4 and on offer several virtual VNWA instruments.

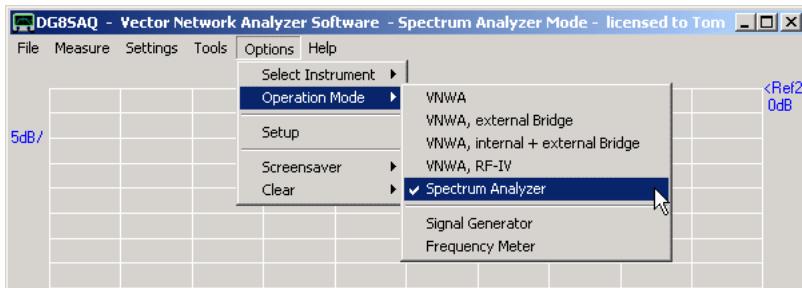
►►► **Note:** Default hardware is the DG8SAQ VNWA hardware.

►►► **Note:** Depending on the instrument selection, clicking the "Options"- "Setup" menu item will open the specific setup of the selected instrument.

OPTIONS - OPERATION MODE

►►► **Note:** This menu is only visible in VNWA instrument mode.

The menu item "Operation Mode" allows to select different operation modes for the VNWA hardware:



1. VNWA

This is the standard **vector network analyzer mode**, which allows to perform vector transmission and reflection measurements.

2. VNWA, external Bridge

This is a mode for **reflection measurements only** as the RX input is used as bridge detector for an external reflection bridge. This mode might be useful, when measuring antennas in a strong BCI environment. For details see page "External Bridge Mode".

3. VNWA, internal + external Bridge

This is a mode for **two simultaneous reflection measurements**. The TX port is used in the normal way, while the RX input is used as bridge detector for an external reflection bridge. The internal bridge is being mapped to S11 while the external bridge is mapped to S22.

Note that this mode requires tapping the TX signal internally with an external isolating amplifier in order to provide a TX signal for the external bridge without causing interference with the internal bridge.

4. VNWA, RF-IV

This is a mode utilizing an external RF-IV test head instead of an external bridge for impedance measurements. RF-IV allows for higher measurement stability at extreme impedances as is explained on page "How to measure Impedances - General Considerations". Details about RF-IV can be found on page "RF-IV Mode".

5. Spectrum Analyzer

The VNWA hardware can also be used as a rudimentary spectrum analyzer. Details can be found on page "Spectrum Analyzer and Tracking Generator Mode".

6. Signal Generator

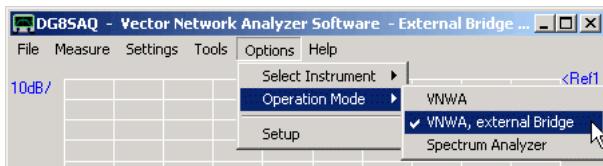
The VNWA hardware can act as a rudimentary CW signal generator. Details can be found on page "Signal Generator Mode".

7. Frequency Meter

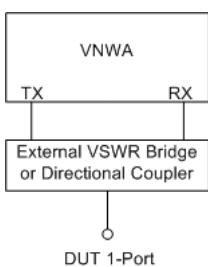
The VNWA hardware can be used as a very accurate frequency meter. See page "Frequency Meter Mode" for details.

EXTERNAL BRIDGE MODE

The VNWA hardware can also be operated with an **external reflection bridge**. To do so, select **Operation Mode - VNWA, external bridge**.



The "bridge" can be as simple as an **SMA-Tee**, or as sophisticated as a **directional coupler** or a **Wheatstone bridge**. The bridge is connected with its signal input to the VNWA TX port and the bridge voltage output is connected to the RX port as can be seen below.



Clearly, **only reflection measurements are possible** in this setup.

The setup can be SOL calibrated in the normal manner. The calibration standards are to be connected to the DUT port of the bridge.

Measurement results are **displayed in the S11 memory space**. Note, that the memory spaces **S21** and **S12** contain **meaningless results** in this case and should not be displayed.

►►► **Note:** The advantage of this setup lies in the fact, that an **attenuator pad** can be inserted between bridge and RX port. This might be necessary, when measuring **antenna impedances in strong BCI environments**. In such environments the BCI signals might well saturate the internal VNWA bridge mixer. An external bridge followed by e.g. a 20dB attenuator in front of the RX port avoids mixer saturation. Of course, the desired reflect signal is also attenuated by 20dB, but there is still enough dynamic range margin to obtain reflection data with reasonable accuracy. You could also use the internal bridge, connect your 20dB attenuator to the TX port and calibrate at the attenuator. This would also attenuate the BCI signals by 20dB, but the desired reflection signal would be attenuated by 40dB, since it travels through the attenuator twice (forth and back).

RF-IV MODE

RF-IV mode is way to greatly improve measurement stability for extreme impedance measurements (very high Z, high Q L and C, very low Z).

Principle

Instead of measuring the reflection coefficient S11 of an impedance and calculating the impedance value from it, RF-IV measures the AC voltage **V** and the AC current **I** running through the unknown impedance. These are measured by VNWA with two separate sweeps. Now, why should it be advantageous to measure an impedance with the RF-IV method instead of using the external bridge?

Example:

a) Bridge measurement

Say, you want to measure a resistor with 50 kilohms, which is 1000 times the normalization impedance $Z_0=50$ Ohms. Without measurement error, you would measure $S_{11}=(1000-1)/(1000+1)=0.998$.

If the instrument drifts such that the measured S11 value is off by 0.1%, you will measure $S_{11}=0.999$ instead.

Using this value, an impedance of $Z=50*(1+0.999)/(1-0.999)=99.95$ kiloOhms is measured.

=> Even though your test equipment offers an accuracy of 0.1%, your Z-result is off by almost 100%!

b) RF-IV measurement

Now, do the same measurement using the RF-IV technique. You will measure the voltage **V** and the current **I** with 0.1% accuracy.

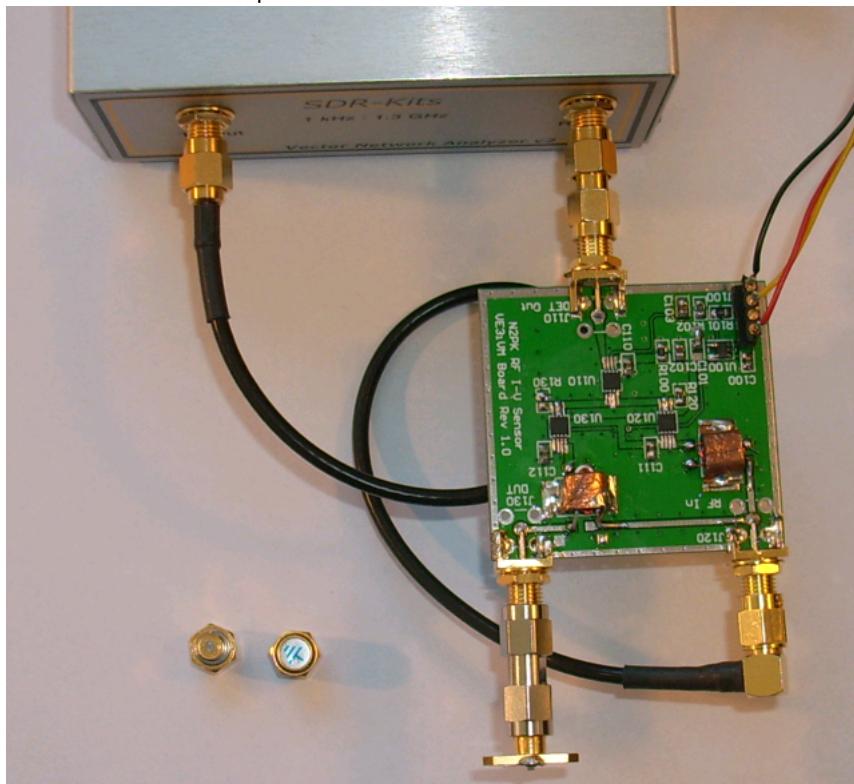
VNWA will calculate the impedance $Z=V/I$. As V and I both show an error of 0.1%, the result Z will be off by at most 0.2%, likely less, ass drift will affect I and V in the same way since both are measured with the same detector. Now, compare this to almost 100% error in bridge mode!

Summarizing, the RF-IV method is always recommendable, when impedances very close to the edge of the Smith chart are to be measured accurately.

Hardware:

I have performed my experiments with an RF-IV test head developed by Paul Kiciak N2PK and kitted by Ivan Makarov VE3IVM. For the kit see Ivan's homepage (http://www.makarov.ca/vna_payment.htm).

The test head hooked up to the VNWA looks like this:

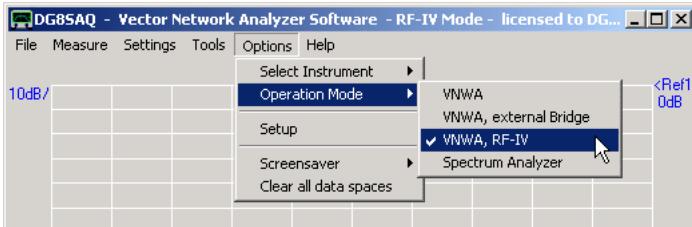


The VNWA TX signal is fed into the test head with a coax cable. The test head detector signal is fed back into the VNWA RX input. The test object (next to the two cal standards) is connected to the test head.

Note, that the test head requires a 5...12V DC power supply, which is tapped from the 5V VNWA USB power. It also requires a 0/3.3...5V switching signal to select I or V measurement mode. VNWA reuses the S-parameter test set signal for this purpose. Note, that the polarity of the control signal is of no importance. Even if I and V sweeps are being swapped, the calibration will fix this.

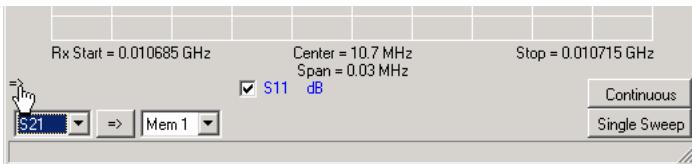
Software Settings and Measurement

In order to use the RF-IV principle, you must switch the VNWA to RF-IV mode:



Note, that the blue main window title bar also shows RF-IV Mode if selected.

Now, do a normal SOL calibration using ordinary short, open and load standards to be connected to the RF-IV head DUT port. Note, that VNWA is doing two calibration sweeps per standard now, one for I and one for V. When done, connect your DUT and press the Single Sweep or Continuous button. Note that again each measurement requires two sweeps (I and V). Switching between I and V channel can be monitored by the VNWA direction arrow changing direction from sweep to sweep:



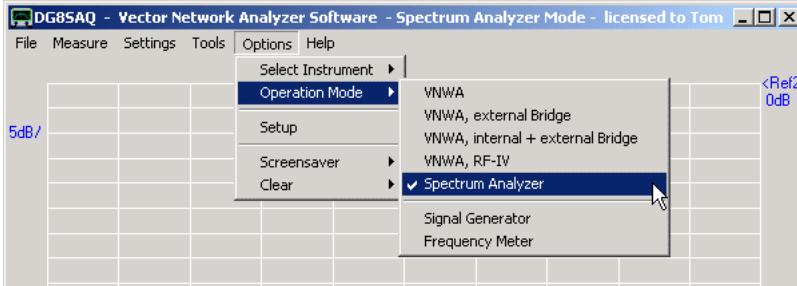
Note, that when you display S11 e.g. in a Smith chart, you will still see a reflection coefficient. But in RF-IV mode this reflection coefficient is calculated from the measured impedance.

SPECTRUM ANALYZER AND TRACKING GENERATOR MODE

Spectrum Analyzer

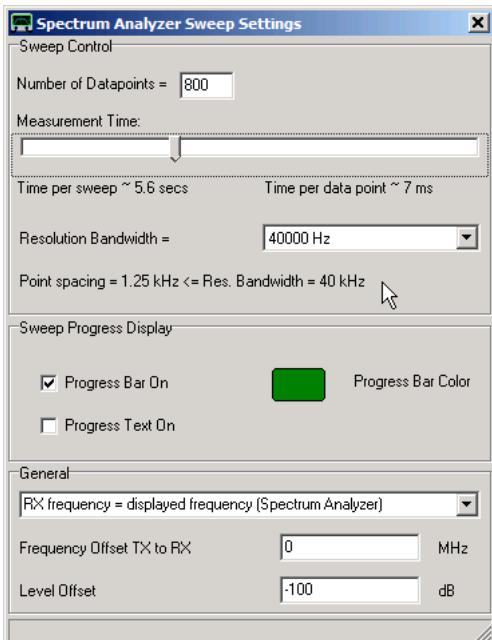
The VNWA can also be used as a rudimentary **spectrum analyzer**. Since the VNWA was never designed to be a spectrum analyzer, there are a few restrictions. Inside the VNWA, the unfiltered DDS LO signal is fed into the RX mixer, i.e. the mixer is fed with a whole spectrum of LO signals. This will lead to the fact that if one applies a pure sine signal to the RX input, one can observe not only the spectral line of the input signal, but also a lot of **birdies or ghost spectral lines**, which are actually not present. The higher the frequency, the more severe these birdies will be. Therefore it is recommended to use SA mode only at frequencies below 100MHz and with the highest possible LO clock multiplier (x20). Another set of birdies stems from the fact that the VNWA RX acts in SA mode as a direct conversion receiver, which makes the RX sensitive to **subharmonics** of the input signal. These can be distinguished by their narrower shape as compared to the real spectral lines.

In order to activate this SA-mode, select "**Options**" - "**Operation Mode**" - "**Spectrum Analyzer**".

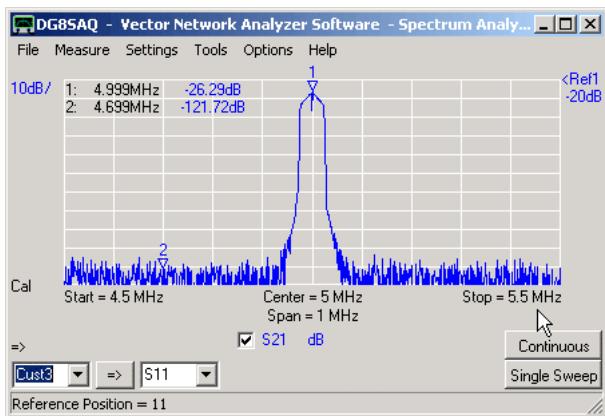


Next, you need to specify the sweep parameters. The following picture shows the settings used in the following example.

►► **Warning:** Note, that in SA mode (unless used with tracking generator) the **frequency point spacing must be smaller than the resolution bandwidth!** If this is not the case, spectrum gets lost, e.g. there might be spectral lines but you won't see them. If settings are inappropriate, the **status text** (left of the mouse arrow in above screen shot) will turn red.



A 30mVpp sine wave signal (= maximum level, that doesn't create overload yet = 30mVpp = -26.5dBm = 2.25uW) is fed into the VNWA RX port.



Note, that by choice of the level offset, **signal power** can be read off directly in **dBm**, e.g. from marker 1.

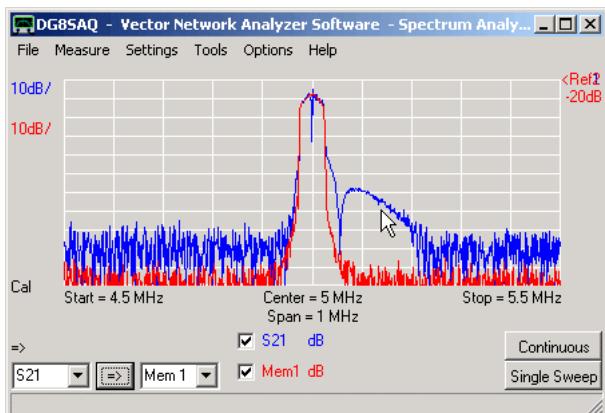
Note, that a dynamic range better than 90dB is achieved.

Note: For most sound cards, the dynamic range is limited by the DC offset of the sound card ADCs. This DC offset can be efficiently removed by performing a **crosstalk thru calibration**, which has been done in all displayed measurements.

The **shape of the spectral line** is determined by the frequency response of the sound card. As zero frequency is in the center of the spectral line and sound cards are AC coupled, there is a small notch in the center of any spectral line, which might or might not be seen dependent on choice of the frequency grid.

The picture below, compares the measurement results for two different **sweep time rates**. The red trace is identical with the above picture.

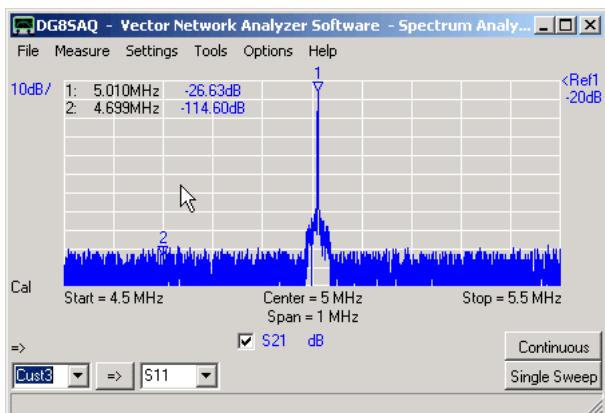
The blue trace was measured with the fastest available sweep rate (0.2ms / frequency point)



Note, that decreasing the sweep rate increases the noise floor.

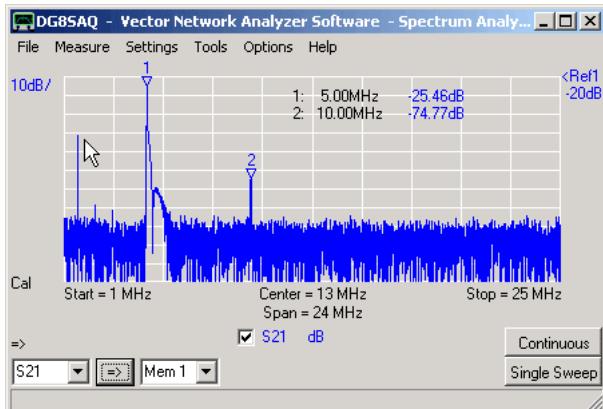
Also note, that if you sweep too fast, the IF filter (=sound card) starts "ringing" (feature above the mouse pointer) due to the the limited bandwidth.

The picture below shows again the same signal with a lower resolution bandwidth and slow sweep rate. Resolution bandwidth control is achieved by changing the sample rate and utilizing the fact, that the resampler of the Windows sound system provides a special anti aliasing filter fitting for every sampling rate.



Note, that the noise side humps are likely artifacts of the Windows anti-aliasing filter algorithm.

Below, again the same signal is swept in a wider frequency span.



Due to the fast sweep rate, ringing is visible on the right skirt of the main peak at 5 MHz. The first harmonic is clearly visible at 10 MHz. At lower frequencies, spurious signals are being seen, which come from unfiltered aliasing frequencies of the DDSes. Some of them can be identified by their width, which is half of the regular peak width.

Tracking Generator

Note, that in spectrum analyzer mode, the TX oscillator is running all the time as a tracking generator. Thus, in this mode, the VNWA can be operated as a **scalar network analyzer**.

One can still perform a SOLT calibration, but the phase information will not be available.

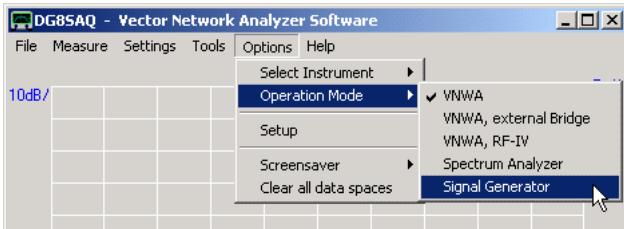
The benefit of using this scalar mode lies in the fact, that the TX frequency can be offset by a value specified in the sweep menu and the analysis bandwidth is rather large (20kHz). This allows to **measure transfer characteristics of frequency mixers and converters** by using an external local oscillator, which frequency must coincide with the specified frequency offset. Its stability must coincide with the resolution bandwidth.

SIGNAL GENERATOR MODE

Generator

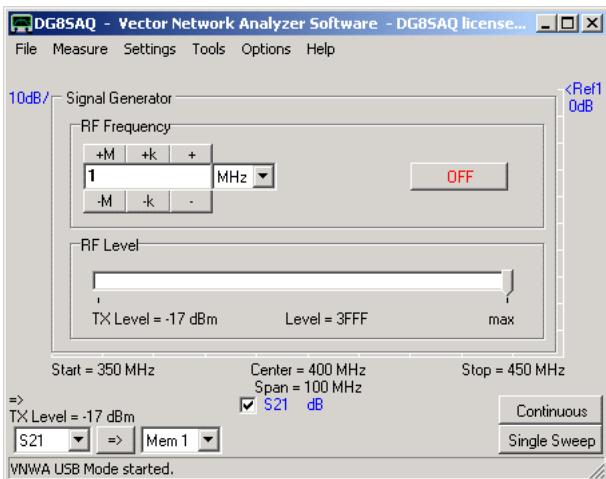
The VNWA2/3 hardware can act as a simple CW signal generator outputting a variable amplitude fixed frequency at its TX port. Moreover, VNWA3(E) allows to AM/FM modulate the output signal with a sine wave with arbitrary frequency of up to about 3 kHz (VNWA software version 36.2.z and VNWA3 firmware version 5.19 or later required).

The signal Generator mode is activated by the main menu "Options-Operation Mode-Signal Generator" or by the keyboard shortcur **ctrl-g**:

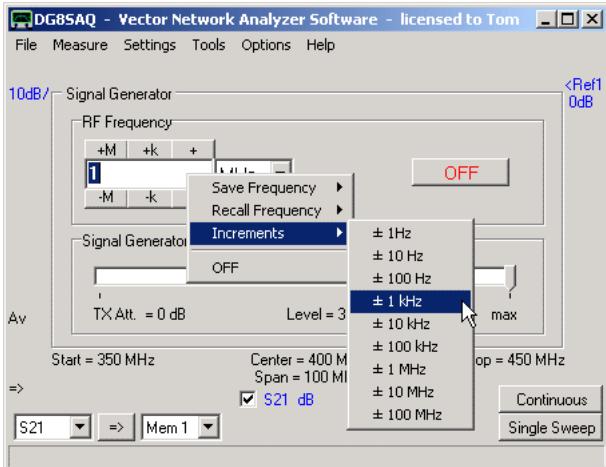


The signal generator mode is not a unique mode but it can be activated on top of other modes. Of course, it is not possible to run the VNWA as signal generator and sweep at the same time, though. An attempt to sweep will switch off the signal generator facility.

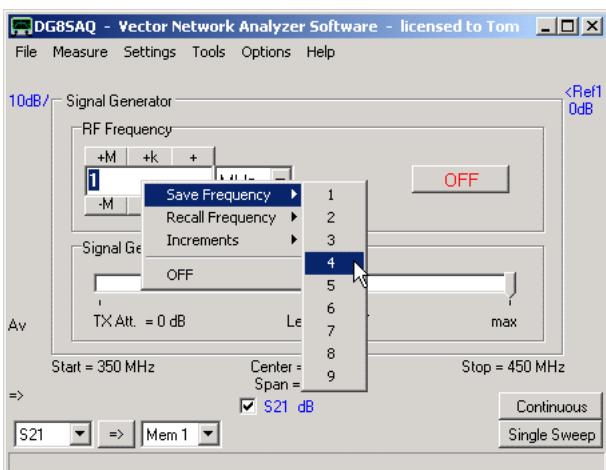
If the signal generator mode is activated, the signal generator controls become visible in the main window:



- The signal generator can be switched on or off.
 - The power level can be controlled with the RF level slider or by holding the mouse pointer over the power level label and turning the mouse wheel. The latter control uses very small steps.
- Note, that this control is only visible if your VNWA firmware supports power control. Power control is not available for VNWAs controlled via parallel port interface. Also note, that the signal generator power level is independent of the VNWA mode power level. Both levels are being memorized and applied depending on the current operation mode.
- The frequency can be changed by editing it with the keyboard, by using the pushbuttons or with the mouse wheel. The mouse wheel increment can be set by right-clicking the frequency input field:

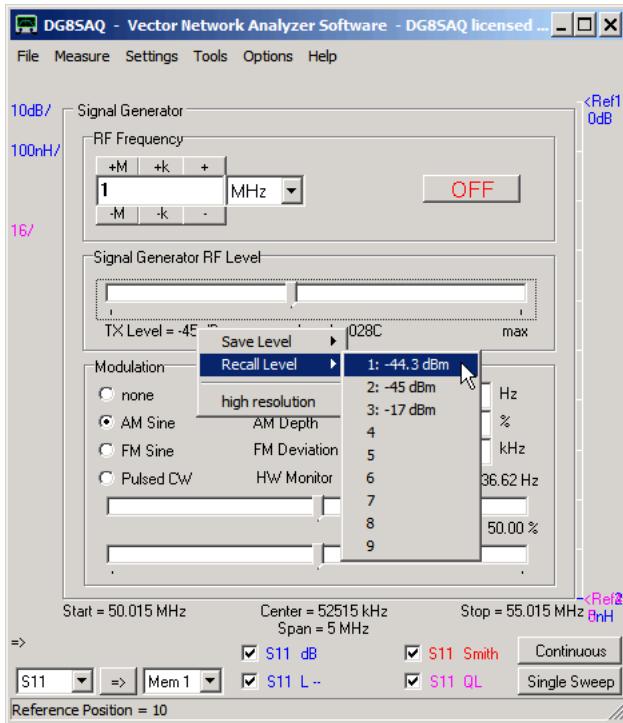


- Up to 9 frequencies can be stored or recalled via the right-click menu:

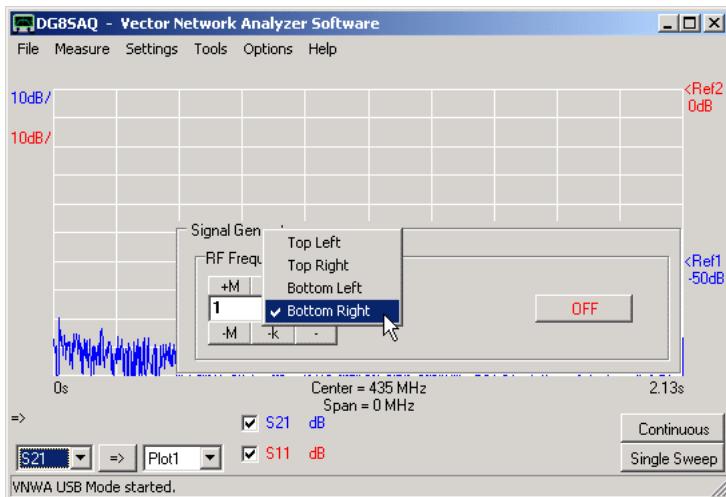


- The signal generator can also be deactivated with the right-click menu. Alternatively, it can be deactivated by selecting any other operation mode or with the keyboard shortcut ctrl-g.

- Up to 9 power levels can be saved or recalled by right-clicking the **TX-Level label**:



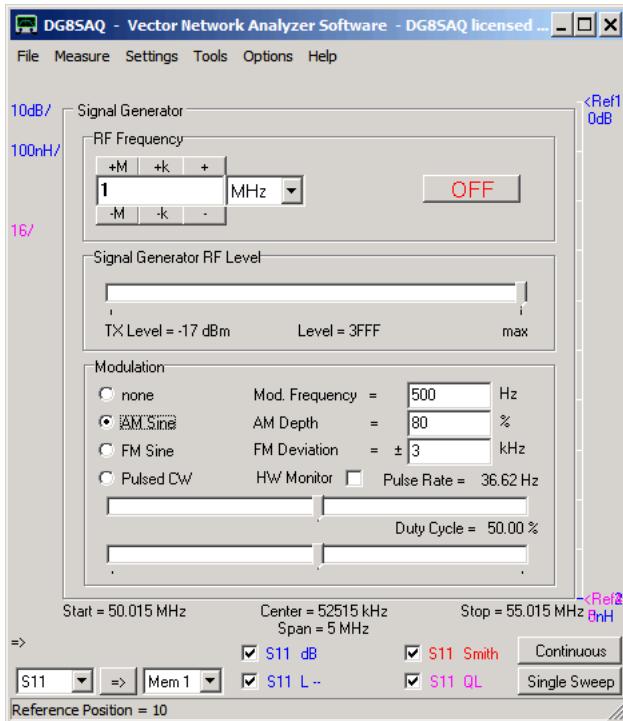
- Here, also the displayed level resolution can be increased to 0.001dB (high resolution).
- The signal generator panel can be aligned to any of the four main grid corners by right-clicking the panel title and selecting the desired alignment style.



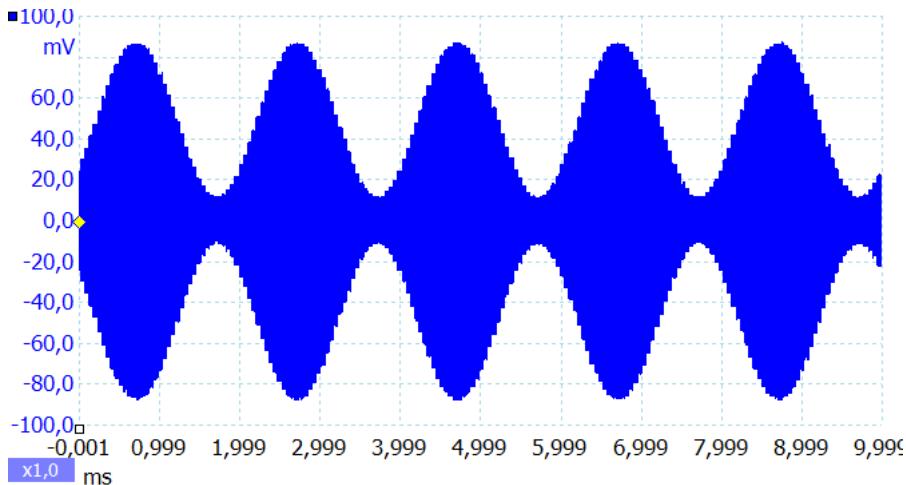
►►► **Note:** The VNWA keyboard shortcuts are deactivated while the signal generator mode is active!

Modulator

If the system supports signal modulation (VNWA3(E), software version 36.2.z or later, firmware version 5.19 or later), then the **Modulation** controls become visible:

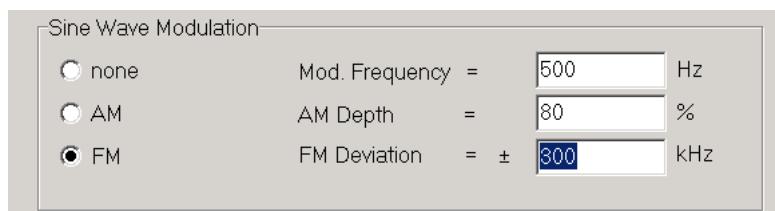


Above settings produce a 1 MHz signal amplitude modulated with a 500Hz sine wave with 80% depth:

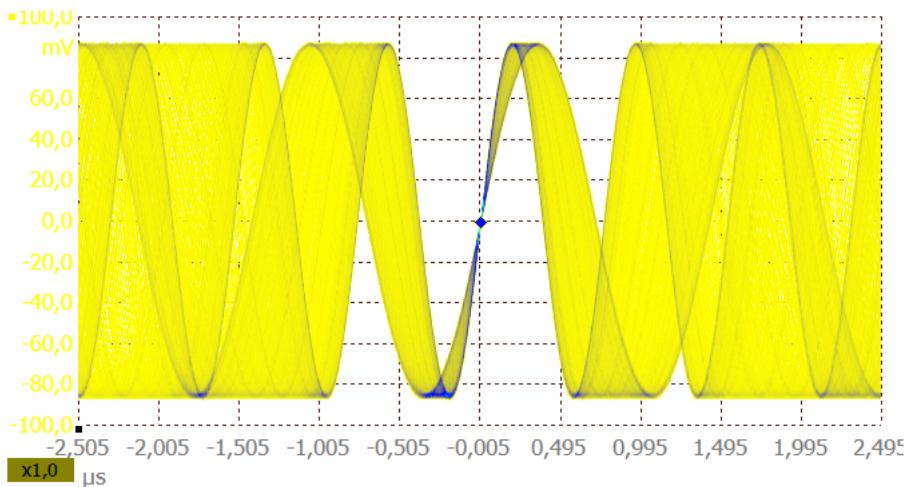


Note the little steps in the modulation, which arise from the fact that the modulation sine wave is generated by a software DDS with about 16kHz clock and 8 bit (=256 step) resolution and the modulation is performed digitally. This puts the Nyquist limit at about 8kHz. The actual modulation frequency should be chosen well below.

The same applies to frequency modulation, which is performed by digitally switching between a maximum number of 256 discrete frequencies.

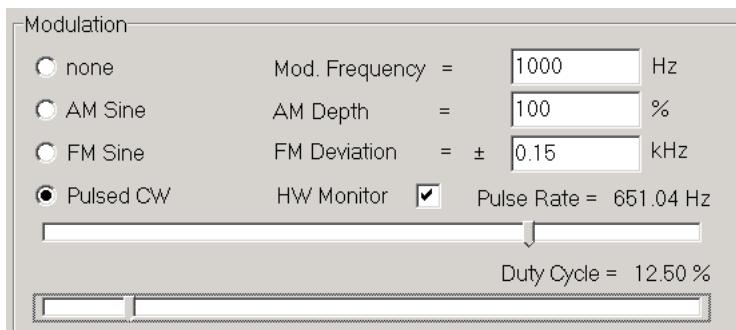


Above settings produce a 1 MHz frequency modulated with a 500Hz sine wave with 300 kHz FM deviation, i.e. the output signal varies between 0.7MHz and 1.3MHz:

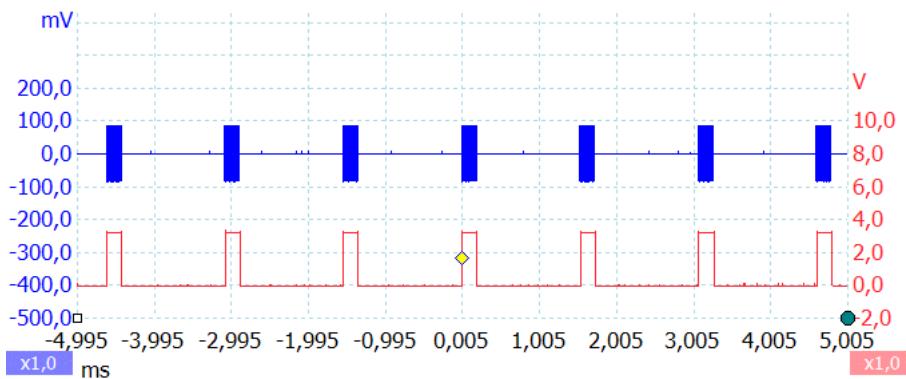


Note, that due to the digital control the FM deviation is very accurate and thus the VNWA is very well suited to calibrate FM deviation meters.

From software version VNWA36.3.8.3 and firmware v5.22 (VNWA3/3E only) on, there is also pulsed **cw modulation** available:



With above settings a train of RF pulses with repetition rate of 651Hz and 12.5% duty cycle is produced (blue trace):



If the **HW-Monitor** checkbox is checked, a synchronous digital hardware monitor pulse train is produced on the Rotor start/stop control signal. This may be used for synchronizing measurements with the RF bursts. Note that for technical reasons the hardware monitor signal shows a delay of 5us with respect to the RF bursts.

Also note, that the modulation is performed digitally by the VNWA on board MCU in real time while the same MCU has to handle USB traffic. If the USB port in use belongs to a hub branch with a lot of USB traffic, the modulation will get interrupted while the MCU handles the incoming USB signals. This will cause clicking noises when listening to the modulated signal in a receiver. Kurt Poulsen OZ7OU has investigated this phenomenon in detail and has worked out guidelines how to avoid this effect, which can be found on page Optimizing USB Traffic for Modulator Use.

FREQUENCY METER MODE

The VNWA can be used as a very precise frequency measurement instrument.

The signal to be analyzed is to be applied to the **RX port** of the VNWA. The VNWA is to be set to **zero span in VNWA mode**.

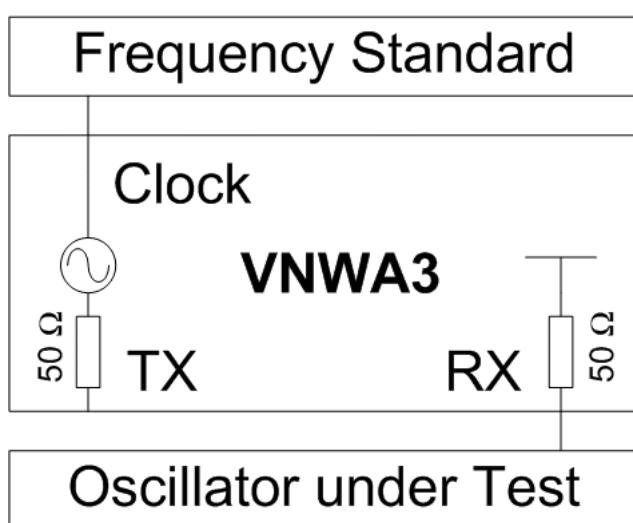
The frequency is determined by measuring the phase of the RX input signal against the VNWA reference oscillator in equidistant time steps.

As the angular frequency is the time derivative of the measured phase, the frequency can be calculated from the measured phase.

Obviously, the timing is crucial for this measurement. Therefore, it can only be done with a **VNWA in USB mode**, i.e. VNWA controled via LPT interface will not work.

In the simplest setup, the unknown signal frequency is measured against the VNWA internal clock oscillator, which -in the case of VNWA3- shows an accuracy of the order of +/-1ppm.

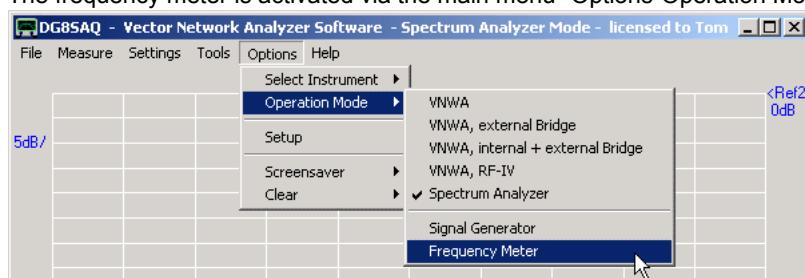
The phase measurement allows for much higher measurement precisions, even at short measurement times, therefore it is worthwhile to connect an external frequency standard. VNWA3E has an optional external clock input:



Make sure to use sufficiently large DDS clock multipliers to avoid sensitivity dips. There are several possibilities to measure the RX input frequency using the VNWA software.

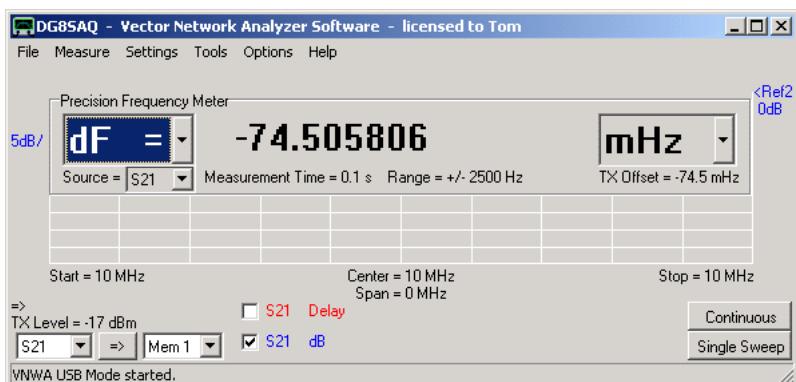
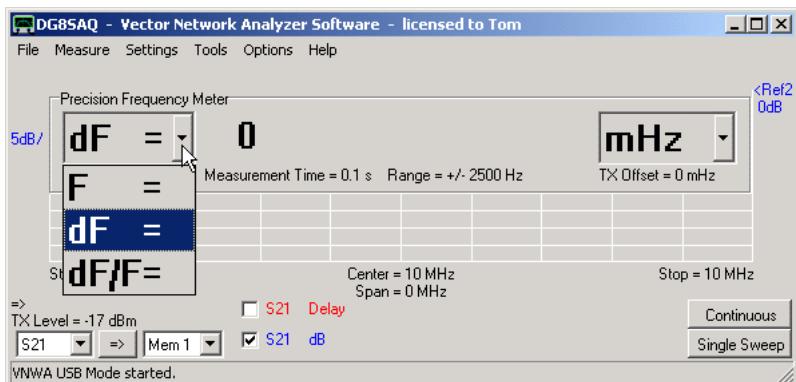
1. Frequency Meter

The frequency meter is activated via the main menu "Options-Operation Mode":



Selecting "**Frequency Meter**" will automatically select VNWA mode, set the frequency span to zero and display the "**Precision Frequency Meter**" panel.

The frequency meter can display absolute frequency **F**, frequency deviation from Center **dF** or relative frequency deviation **dF/F**:



Selectable items:

- displayed result: frequency **F**, frequency deviation **dF**, relative frequency deviation **dF/F**
- source: normally S21, i.e. RX input signal is being analyzed
- unit: in above example mHz = Millihertz is selected

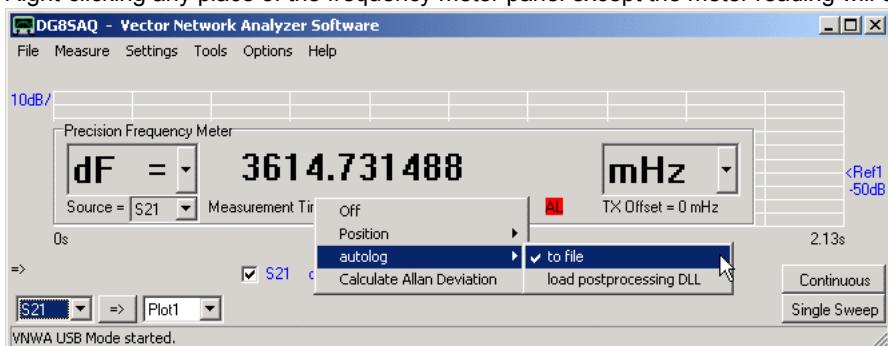
Info items:

- **Measurement time** = time per data point * number of points. This is the sweep time, which is being influenced by the sweep settings. This would be the equivalent to the gate time for a normal frequency counter.
- **Range**: This is the frequency range around Center, for which frequency measurement is possible. This is influenced by the sweep setting "time per data point".
- The longer the time per data point, the lower the measurement noise, but at the same time the allowed frequency range becomes smaller due to the underlying signal processing.
- **TX Offset**: DDSes have a very small but still finite frequency channel spacing. TX offset indicates by how much the actual TX output frequency deviates from the indicated Center frequency. This can be of the order of 100mHz. This is for info, the displayed frequency is already corrected for this deviation. But note, that when measuring the TX output frequency, you will not measure zero deviation but this finite deviation, because that's the real output frequency as in above example. The TX offset is determined by the clock multipliers and by the VNWA firmware rounding behavior. Therefore, the VNWA must be connected so the software can read out this TX offset.

Operation:

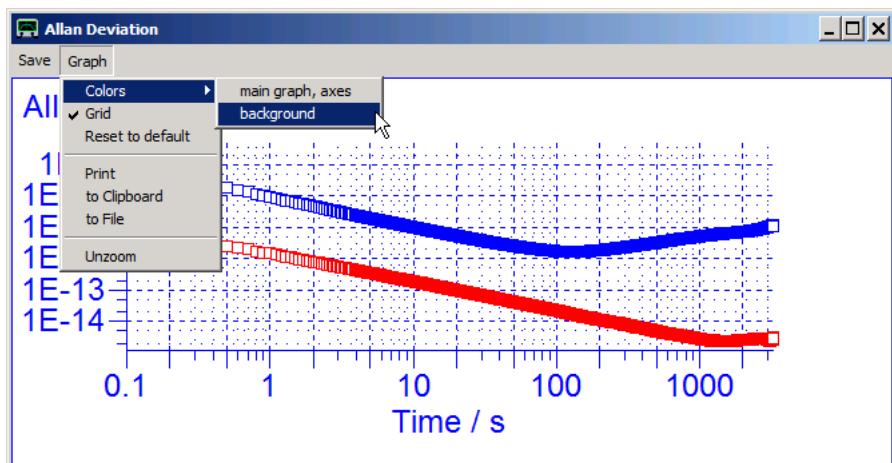
Let the VNWA sweep and the frequency meter will update the measured frequency at the end of every completed sweep. This works equally well in single sweep mode and in continuous sweep mode.

Right-clicking any place of the frequency meter panel except the meter reading will open the meter main popup menu:



There, you can

- switch the frequency meter panel **Off**
- Change the **Position** or alignment of the panel relative to the main grid, e.g. bottom left as seen above.
- **activate data logging** to a text file. The measured frequency (or whatever you choose to be displayed) is being appended to a text file of your choice together with the elapsed time since the first sweep was started. Activated data logging is being indicated by the red AL-Label, see above. This feature is useful for e.g. obtaining the long time behavior of a crystal oscillator. Note, that all data will be appended to the selected file, i.e. old data will not be deleted. So, if you want to start a new list, select a new log file or delete the old log file.
- If the frequency is to be measured together with some other parameter e.g. voltage or temperature, this can be achieved by linking a user written **dynamic link library (dll)**, which is called at the end of each sweep and which then performs the latter measurement and returns the result to the VNWA software, so the external measurement can be logged together with the measured frequency to the same data file. A dll code example returning a simple string can be found on page "Frequency Meter Postprocessing dll".
- **calculate the Allan deviation** from sweep data and Mem* data and display it:



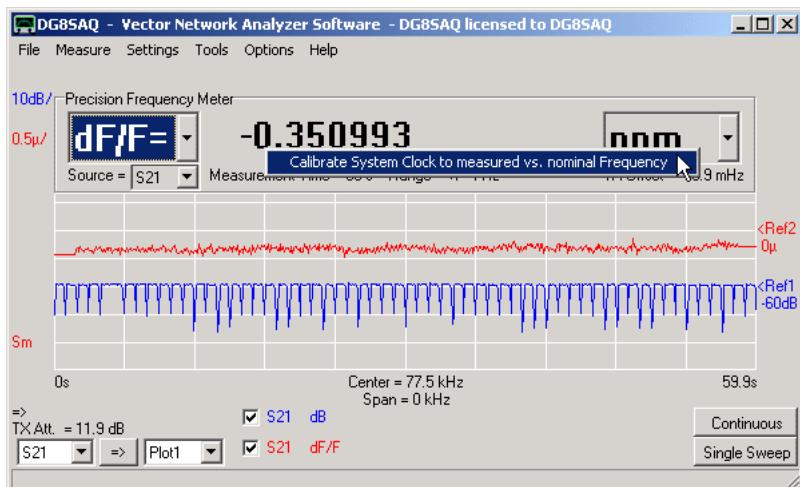
Note, that the Allan deviation is not only calculated and displayed from the data specified in the Frequency Meter's source field, but also from all Mem-fields visible in the main window. The mem data is plotted in the same color as on the main window.

The Allan deviation plot menu offers option to configure the plot, to save the data to an ASCII file, to save the graph to a file, to the clipboard or to print it.

Note the cursor cross in above plot which is activated by holding the left mouse key down. Also note the cursor coordinates on the lower left of the plot.

The graph also allows box zooming, which is accomplished by holding the ctrl-key down, then pressing the left mouse key and drawing a zoom-box with the mouse. First release the mouse key, then release the ctrl-key to invoke a zoom. Unzooming is accomplished via the Graph menu or by right-clicking the graph and selecting unzoom.

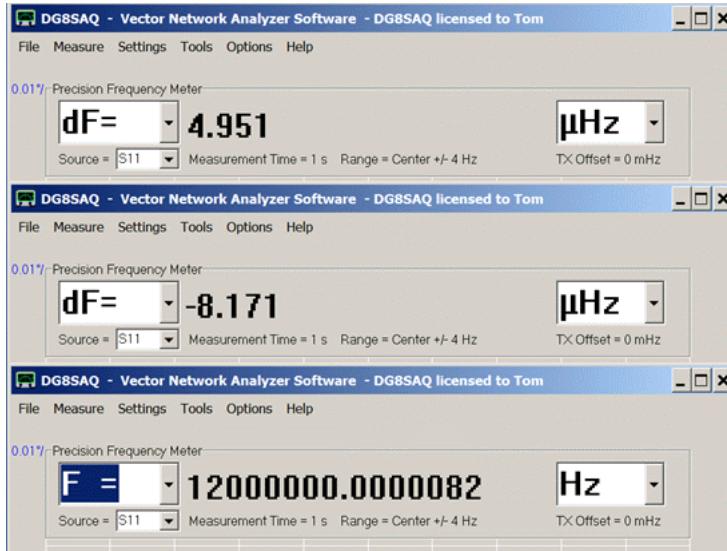
Right-clicking onto the meter reading lets you **calibrate the VNWA clock** to the measured and displayed frequency versus the nominal center frequency:



Details can be found on page Calibrating VNWA System Clock using Frequency Meter.

2. Accuracy

In order to get a feeling for the measurement accuracy to be expected, I have measured the VNWA3 internal TCXO against itself by measuring the TX output frequency by selecting source S11. Clearly, this measurement should exactly reproduce the indicated TX Offset, i.e. dF=0 in below example. The time per data point was set to 100ms which leads to a Range of +/-4Hz and 10 data points per sweep was selected, which leads to a total measuring time of 1 second. The center frequency was 12 MHz.



The above picture shows three typical sweeps, two of which display dF.

With the selected settings and 1second measurement time, it turns out that the frequency of 12 MHz can be measured with an accuracy of about +/-10 micro-Hertz!!!

Now try to beat this with a normal frequency counter.

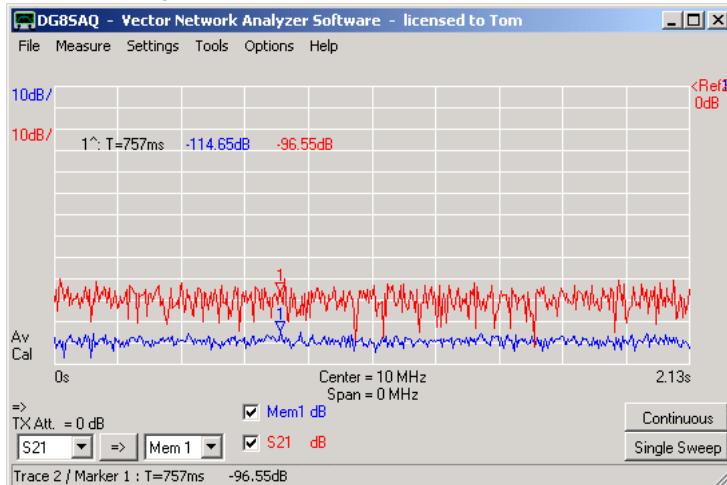
►►► **Note:** When displaying or logging absolute frequencies, you may be running out of digits as even the large number of double precision digits can be exceeded here, see above absolute frequency display.

Possible sources of measurement error:

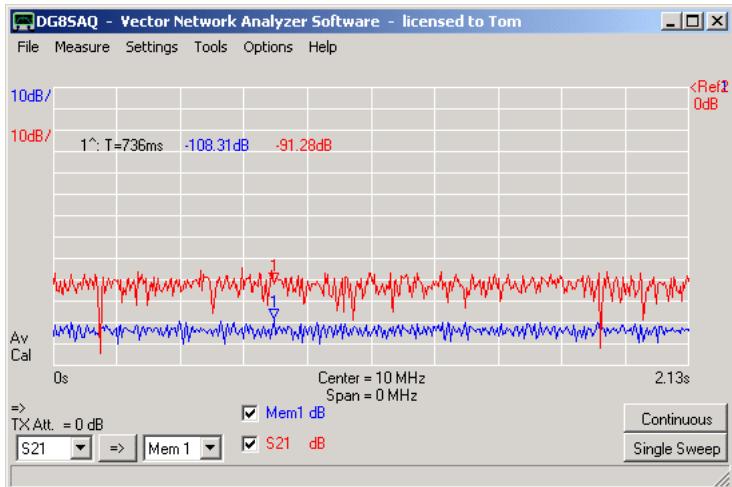
Thanks to Ulrich Bangert, DF6JB, for pointing out possible error sources!

a) Signal crosstalk:

When some of the reference signal leaks into the measurement signal channel, a phase error will result. In order to estimate the significance of this error, isolation measurements have been performed VNWA3E main audio codec ...



and auxilliary audio codec:



Long time averaging has been used (blue traces) to get below the system noise floor (red traces). For both codecs, the signal crosstalk is well below 100dB, i.e. the interferer amplitude is less than 10^{-5} times smaller than the signal amplitude.

This will lead to a phase error smaller than 10^{-5} rad. So, within an analysis time interval of 1 s the frequency error will be less than $10^{-5}/(2\pi \cdot 1\text{ s})$ or about 2 micro-Hertz.

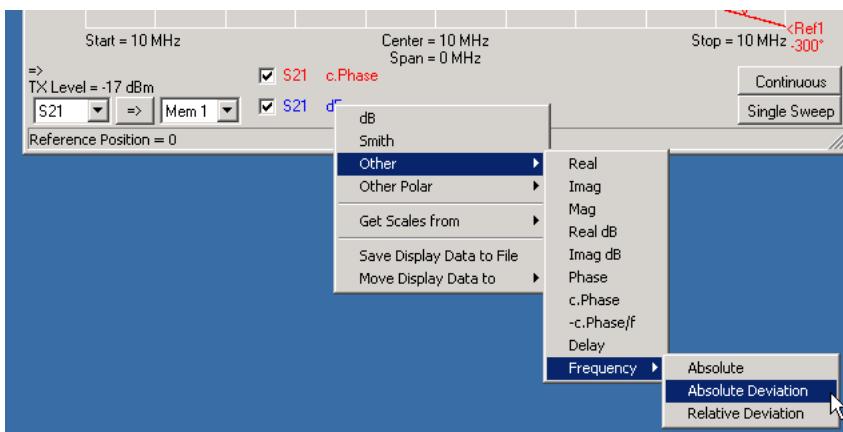
For longer measurement intervals this error will become even smaller according to a 1/time law.

b) DDS digital to analog converter (DAC) nonlinearities:

When a DDS samples a signal in fractional-n mode (i.e. the DDS core divides the input clock signal by a non-integer number), different sections of the DAC characteristic will be sampled periodically with some times rather long repetition rates. If the DAC characteristic shows deviations from the ideal linear behavior, these deviations will translate into phase errors, which may vary slowly depending on the DDS frequency register content. These phase errors may interfere with very accurate frequency measurements. They can be avoided by making sure that the DDS works as an integer frequency divider. E.g. if two 10 MHz signals are to be compared, **use clock multipliers and premultipliers which are powers of 2**.

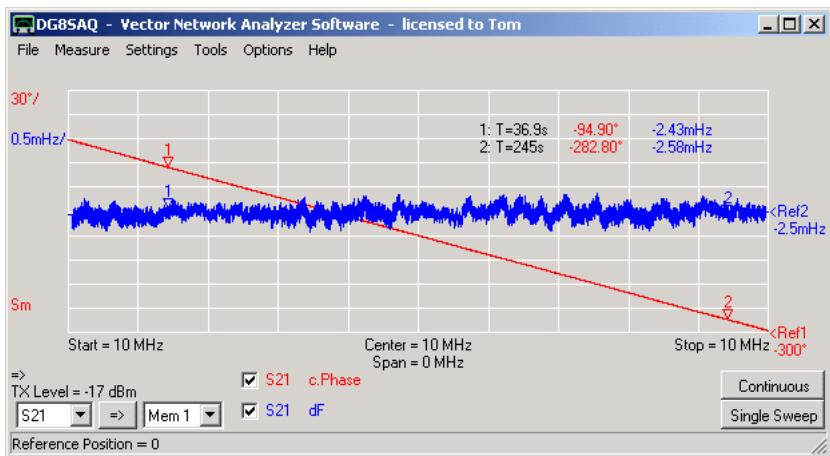
3. Frequency Trending

As explained above, data logging can be used to obtain frequency vs. time data for a stability analysis over a long period of time. Over a short period of time, the time dependent frequency can directly be displayed graphically using display types dF, F or dF/F:



Example 1

The following screenshot shows a measurement on one Rubidium 10 MHz frequency standard, while the VNWA3E is clocked by a second independent Rubidium 10 MHz frequency standard. The sweep has been done with 65000 data points and covers a total time of 260 seconds:



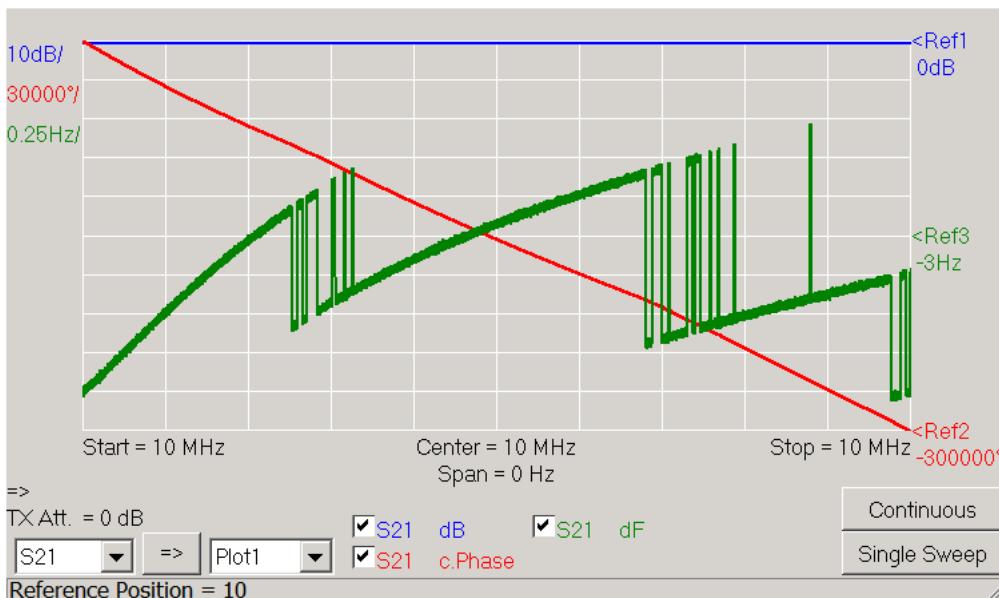
As one can see, the Rubidium standards are detuned by about 2.5 milli-Hertz against each other with some temporal fluctuation (<+0.5 milli-Hertz), which is almost invisible on the phase plot. The frequency offset shows as linear phase progression. Note, that numerical noise from calculating the time derivative of the phase (i.e. the frequency, blue trace) has been removed by trace smoothing over a time interval of about 3 seconds. Note, that frequency markers display the time at which the data point was measured after sweep start instead of frequency for zero span which is used here.

►►► **Note:** The maximum allowed number of data points has been increased with VNWA version 36.0 from 8192 to 65000 for this application.

The software may react very sluggish, when using such a vast number of points!

Example 2

Another example for frequency trending is shown in the next image. Here, a 10 MHz Rubidium frequency standard has been measured against the VNWA3E internal TCXO. The sweep time was again 260 seconds at 65000 data points:



Assuming the Rubidium standard to be accurate, the internal TCXO must have a deviation of about +3Hz +-1Hz or +0.3ppm+-0.1ppm, which is excellent. One can clearly see that the temperature compensation of the TCXO is achieved by digitally switching the frequency in small increments. The effect looks dramatic on the plot because of the very high accuracy of the test setup, but in fact it is only a small correction. Note, that the phase curve is smooth while only slightly changing tilt.

FREQUENCY METER POSTPROCESSING DLL

The Frequency Meter Mode is a highly precise method to measure the frequency of an RX input signal. It also allows logging of the measured frequency together with the time of measurement to a data file. If the frequency is to be measured together with some other parameter e.g. a voltage or temperature, this can be achieved by linking a user written **dynamic link library (dll)**, which is called at the end of each sweep and which then performs the external measurement and returns the result to the VNWA software, so the external measurement can be logged *together* with the measured frequency to the same data file. A dll code example returning a simple string can be found here:

| Postprocessing DLL Code example (Borland Delphi 6)

```
library CounterDLL;

uses
  Windows,
  SysUtils,
  Classes;

{$R *.res}

procedure _Init; export; cdecl;
begin
  FreeConsole;
  AllocConsole;
  writeln('UserDLL function "Init" called.');
end;

procedure _Close; export; cdecl; //called only once when dll is unloaded
begin
  FreeConsole;
end;

procedure _Process(Data: PChar); //called only once when dll is loaded
  export; cdecl;
begin
  writeln('UserDLL function "Process" called.');
  StrPCopy(Data,'Hello World!'); //called after every sweep, data is passed as a string to VNWA, max. size 128
  characters
end;

exports _Init;
exports _Close;
exports _Process;

begin
end.
```

CALIBRATING VNWA SYSTEM CLOCK USING FREQUENCY METER

General

The **Frequency Meter** offers a very convenient way to **calibrate the VNWA system clock** which drives the DDS chips.

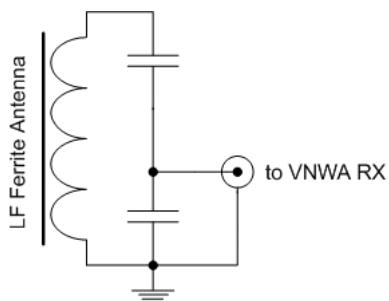
Simply apply a precise frequency signal to the RX port of the VNWA. Set the VNWA center frequency to the nominal source signal. Let the Frequency Meter measure the frequency.

Then invoke the "calibrate clock" command and the DDS clock frequencies will be corrected automatically.

Simple high precision frequency source

LF and VLF radio stations generally derive their transmission signal from a very precise atomic clock. E.g. the German time signal transmitter DCF77 transmits a time code on 77.5 kHz with sufficient power to be received all over Germany and in most neighbor countries. Its signal is derived from a very precise Caesium fountain clock located at the PTB in Braunschweig.

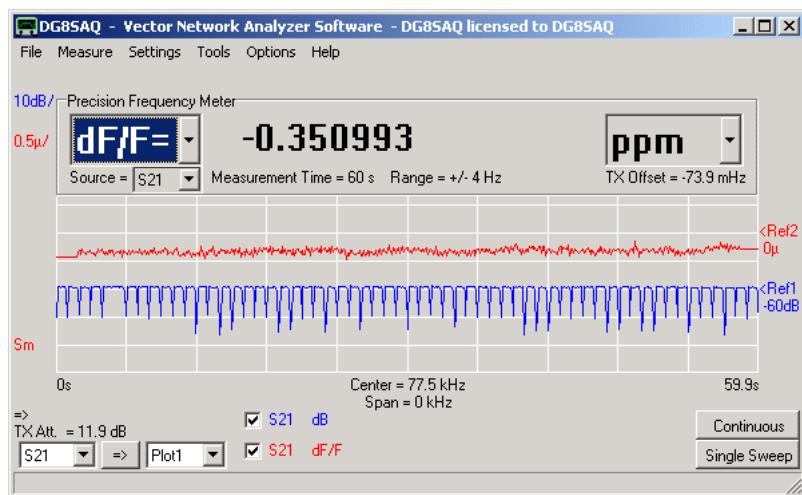
As the VNWA RX is a very sensitive receiver, a simple ferrite antenna is sufficient to receive this signal:



Capacitors are to be chosen such that a reasonable match to 50 Ohms is achieved, i.e. the capacitor to ground must be a lot bigger than the other one.

Many countries have their own LF transmitters distributing the standard time and a standard frequency.

With this simple antenna I have measured the DCF77 signal in 500 km distance from the transmitter:



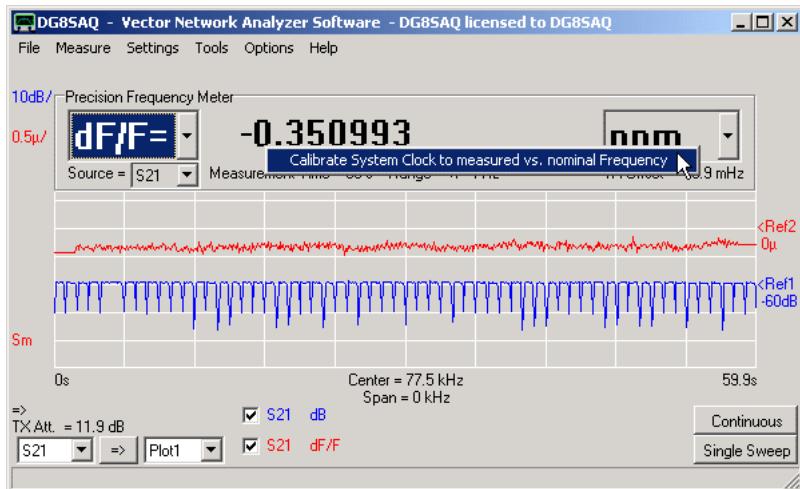
One can clearly see the amplitude modulated time code on the blue trace with an amplitude notch every second except for second 59.

The amplitude level of -60dB corresponds to a signal amplitude at the RX port of about -88 dBm, not exactly much. Over a 1 minute period the signal frequency is measured to be -0.35 ppm off. Of course it is the VNWA TCXO that is really off, but -0.35 ppm from the default value is not bad at all.

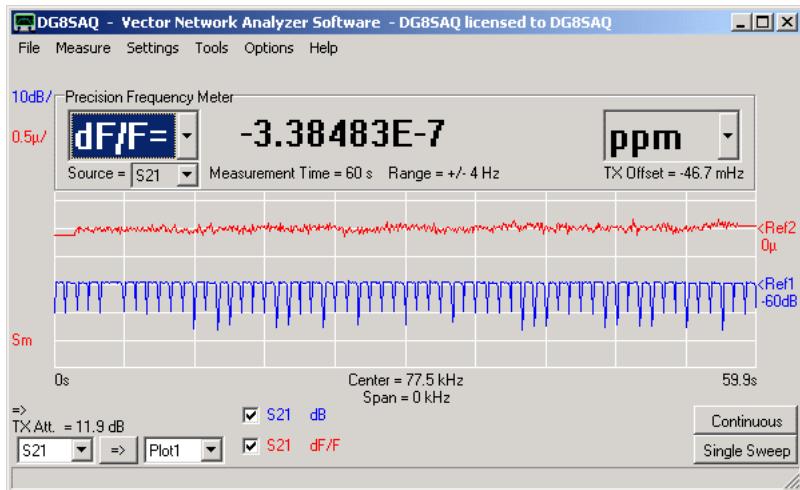
The red trace is displayed with 0.5 ppm per vertical unit. The red trace has been smoothed by a sliding average over 2 seconds.

Correcting the VNWA system clock

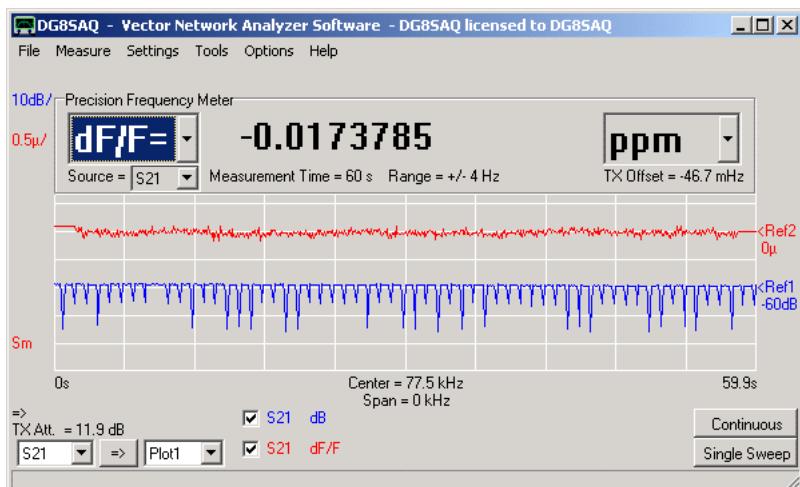
Correcting the VNWA system clock is very simple now. Right-click the frequency reading...



... and invoke the **calibrate system clock menu**. The correct DDS clock frequency will be calculated and applied:

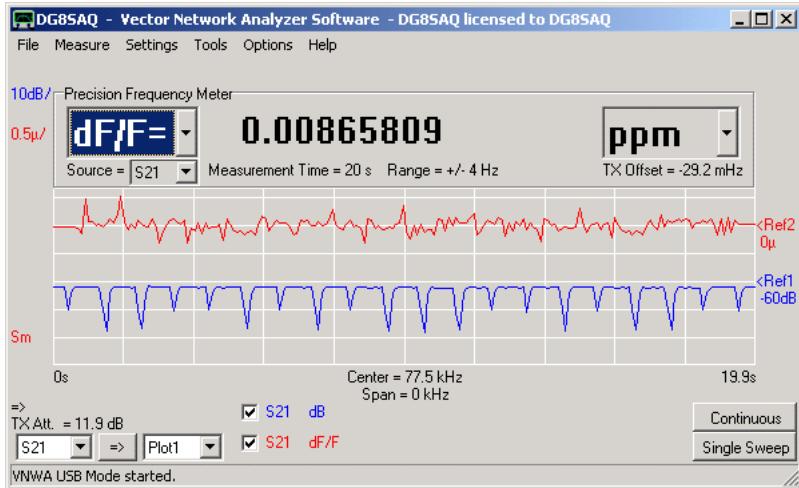


Note, that the frequency offset has been corrected to zero (apart from roundoff error). If we measure again, the measured frequency offset will be a lot smaller than before the calibration:

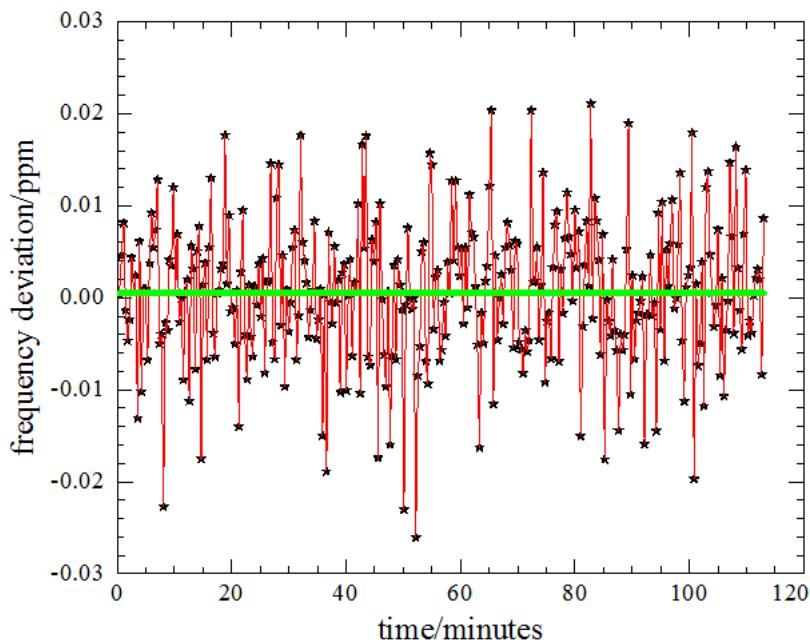


Accuracy considerations

To check reliability of such a low level reference signal (-88 dBm !), I have measured the DCF77 signal against a Rubidium frequency standard over an extended period of time.
Here is the last of 20 second sweeps over 2 hours of time:

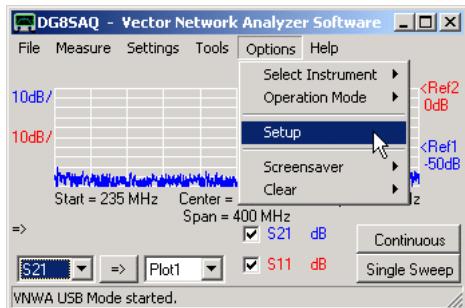


The frequency readings have been logged over the full 2 hours from 11:00 to 13:00 at daytime. Each star is one 20 second frequency measurement. For clarity, the measurement points have been joint by the red polygon.



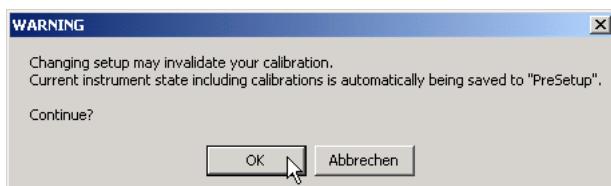
None of the 20 second sweeps was off by more than +/-0.03 ppm. The 2 hour average (green line above) yielded 0.0005 ppm or 5×10^{-10} , which is about the accuracy of my Rubidium standard.
Thus, the measured frequency is accurate and a 20 second measuring time is sufficient to calibrate to +/-0.03 ppm, which exceeds the stability of most TCXOs.

OPTIONS - SETUP (VNWA)



The VNWA setup window allows to specify all **VNWA hardware related settings**. A quick start guide to do the setup can be found here.

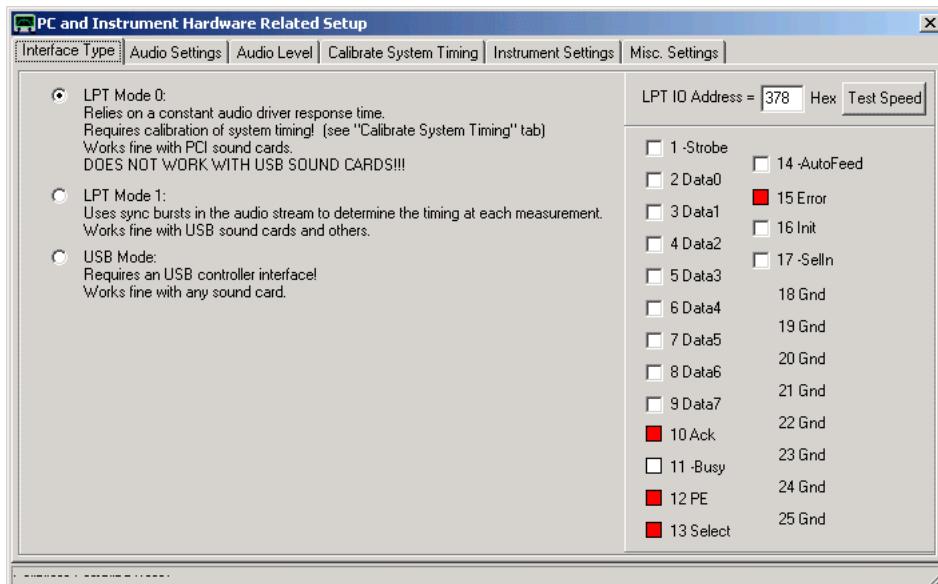
When attempting to enter setup while a calibration is active, a warning will be issued:



If you do not want to be warned every time you enter setup, you can suppress the warning by pressing *ctrl-w*. The warning can be reactivated by pressing *ctrl-w* again.

The setup menu:

Setup input is organized in several tabs, which open when they are clicked. Some of these tabs are context dependent, i.e. they are only visible if applicable to the selected interface type.



Interface Type

USB Settings

Audio Settings

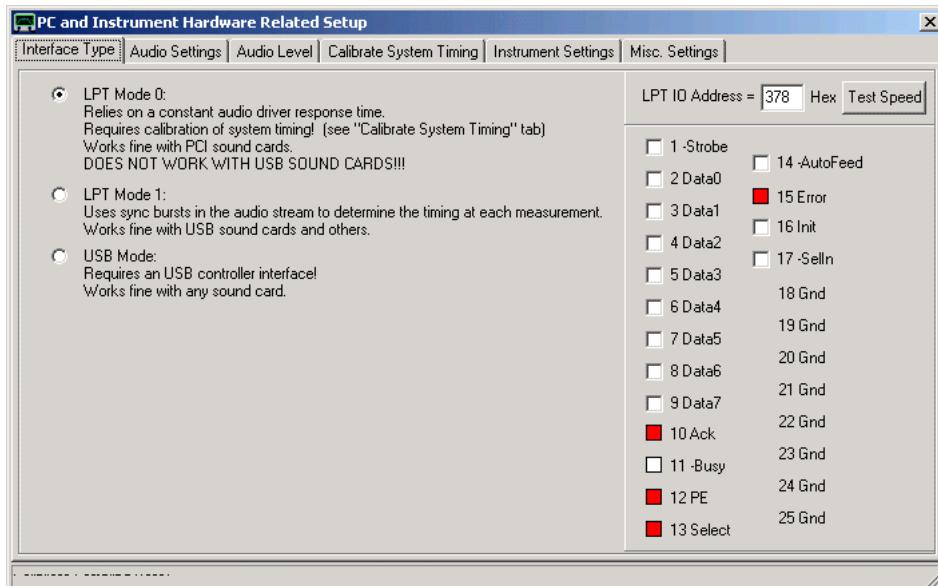
Audio Level

Calibrate System Timing

Instrument Settings

Misc. Settings

VNWA SETUP - INTERFACE TYPE TAB



The VNWA can be controlled via the PC parallel printer port (LPT) or via the DG8SAQ USB_VNWA interface.

For LPT control, there are two timing modes available:

LPT Mode 0 is very experimental and requires a special system timing calibration.

LPT Mode 1 is the safe mode. If activated, the Calibrate System Timing Tab is **invisible**.

If **USB Mode** is activated, all LPT related controls are switched invisible.

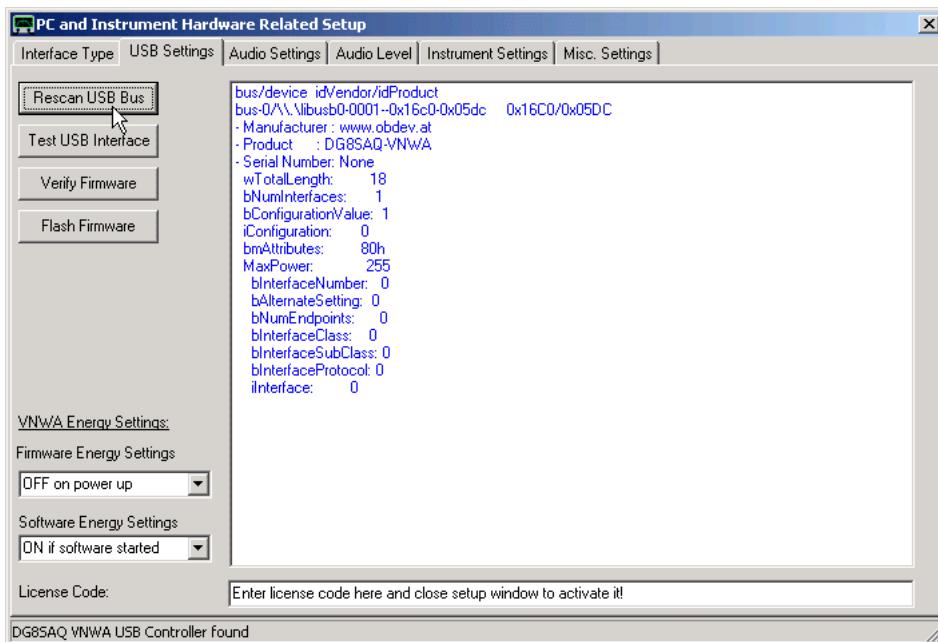
-For proper operation via LPT, the correct **LPT IO address** must be specified here. Any address between 0 hex and FFFF hex can be entered.

-Also crucial for proper operation via LPT is the **minimum transfer speed**, the parallel port must achieve. This can be tested by pressing the "**Test LPT**" button. The test result is displayed in the status line on the very bottom of the setup window. **Note, that this test does not check if your LPT lines are actually switching!** Only the port driver software speed is determined.

-If you have trouble with your LPT interface, you may use the controls on the right for diagnosing it. The checkboxes allow to statically toggle any LPT port output line. The lamps allow to detect the levels of all LPT input lines. Note that **inputs will only be read when outputs are being changed**. If you don't have a scope for checking levels, you can make a loopback cable and control the switching action here. Note, that the controls resemble the LPT port pinout configuration.

VNWA SETUP - USB SETTINGS TAB

►►► **Note:** This tab is only visible, if USB interface type is selected.



In this tab you can configure and test the **DG8SAQ USB_VNWA interface**.

Rescan USB Bus:

This button allows to scan the PC USB interfaces for the DG8SAQ USB_VNWA interface. If successful, the device properties are being listed on the right as shown above.

►►► **Note:** The proper USB device driver must already be installed for this test.

►►► **Note:** In order to use the following controls, the USB Interface must have been **activated** with a **valid license key**.

Test USB Interface:

With this control a communication test between PC and USB interface is initiated. The result will be displayed in the status line on the very bottom of the window.

Verify Firmware:

With this control you can compare the firmware inside your USB device with a hex-file in order to find out if the two are identical or not.

Flash Firmware:

With this control, a firmware update can be uploaded to the DG8SAQ USB interface.

►►► **Warning:** You may damage your interface if you upload unapproved firmware or if you unplug or power down your USB interface during firmware upload. See here for details of the upgrading procedure.

Firmware Energy Settings: These settings tell the USB firmware when to power up the VNWA independently of the VNWA software.

- **Off on power up:** The power supply for the VNWA will **not** be switched on, when plugging it into a powered USB outlet or when powering up the PC.

- **On on power up:** The power supply for the VNWA will be **switched on automatically**, when plugging it into a powered USB outlet or when powering up the PC.

►►► **Note:** The former setting will help conserve energy, e.g. when using battery power. The latter guarantees for maximum warmup time of the instrument.

Software Energy Settings: These settings tell the VNWA software when to power up the VNWA.

- **VNWA always OFF:** The power supply for the VNWA is switched off at all times. This setting may be useful to reset the DDSes or to conserve energy.

- **VNWA always ON:** The power supply for the VNWA is switched off at all times. This setting is useful to guarantee maximum warmup time for the instrument.

- *On if software started:* Upon start of the VNWA software, the VNWA is powered up and upon program termination, the VNWA is powered down automatically. This setting may be useful to conserve energy in battery powered systems.

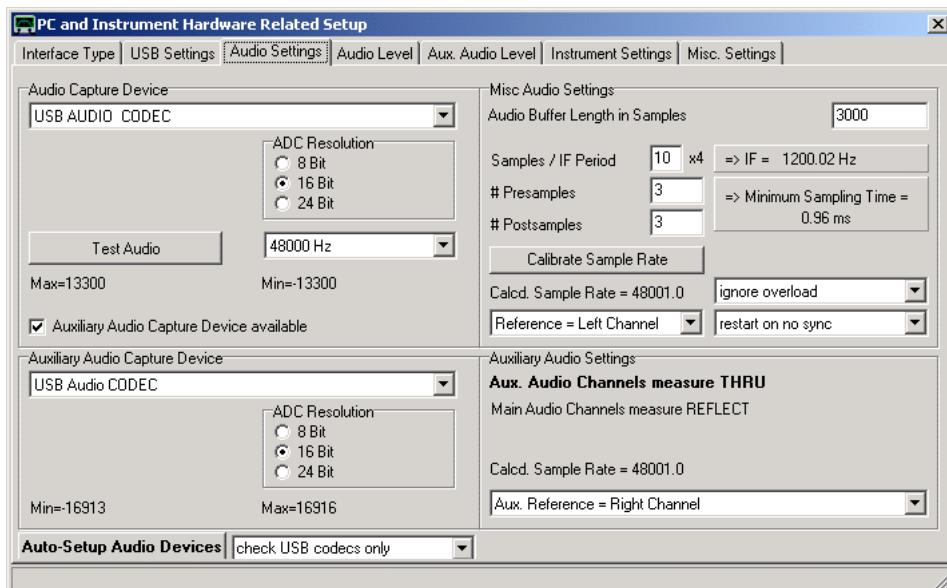
License Code:

Enter a valid license code here.

►►► **Important:** Immediately after entry of a valid license code, you will not yet be able to e.g. test the USB interface. You must **activate the license code** first! In order to activate the license code, you must **either close the setup window or rescan the USB bus.**

VNWA SETUP - AUDIO SETTINGS TAB

The audio settings tab allows to control all sound card related settings.



Audio Capture Device

In the audio capture device panel all sound card settings are being controlled. Select:

For Windows Vista, Windows 7 and Windows 8 (or if the audio mixer is deactivated) select
- the audio capture device

For Windows XP and earlier with activated audio mixer select (entry fields suppressed in above screenshot as audio mixer is deactivated by default):

- the used *sound card*
- *recording* or capture, NOT playback!
- a *stereo line-in or microphone input*, NOT Loudspeaker!

For all Windows systems:

- a standard *sampling rate*. 48 kHz is recommended. Lower sampling rates reduce the CPU speed but also degrade the signal to noise ratio. Higher sampling rates are usually achieved by resampling the 48 k stream from the USB sound codecs.

- the *ADC resolution* of the selected sound card. If a lower resolution than available is selected, only the selected number of bits is used. If a higher resolution is selected, the unavailable bits are padded with zeros, but at the expense of CPU time. The 8 bit setting is only for educational purposes to demonstrate the effects of quantization noise.

►►► **Note:** 24 bit capturing only works on Windows XP and older.

Proper operation of the sound card can be tested with the "**Test Audio**" button. The labels below the button show the smallest and largest sample within a data block. The audio data can be visualized in the "**Audio Level**" tab. Also, the recording levels can be set there if supported by the operating system and codec.

►►► **Note:** Most controls will become inactive when "Test Audio" is activated. Also, the setup window cannot be closed at once in this case.

Auxiliary Audio Capture Device

VNWA3E offers a second audio codec for simultaneous measurement of S11 and S21. The auxiliary codec is to be set up in the same way as the main codec in the Audio Capture Device panel.

Misc. Audio Settings

Available controls:

- "**Buffer length in Samples**" specifies the length of each ring buffer segment. Recommended is 3000. Shorter length means more frequent screen update at the expense of increasing CPU load. Experiments show, that windows cannot cope satisfactorily with buffers shorter than 800 samples.

- "**Samples / IF period x4**" specifies the IF frequency. Note, that the entered number is multiplied by 4. So, if 10 is entered, that means 40 samples per IF period. At a sample rate of 48 kHz, this leads to an IF of $48 \text{ kHz} / 40 = 1.2 \text{ kHz}$, which is displayed next to the edit field. The highest selectable IF is 12 kHz at 48 kHz sampling rate which is also the highest one that works with the used USB sound codec, which has a limited bandwidth of 20 kHz.

- "# Presamples" specifies the number of samples, that are skipped after a frequency change. Recommended is 3.

- "# Postsamples" specifies the number of samples, that are skipped before a frequency change. Recommended is 3. Presamples and postsamples are safety margins to cope with variations between the sampling clock and the CPU clock. It is to be avoided, that data is analyzed, that was taken, while the VNWA received a frequency update due to the resulting digital noise.

►►► **Note:** A measurement is only possible, if at least one IF period + the presamples + the post samples fit into the selected measurement time per data point. The minimum necessary measurement time is displayed. If a shorter measurement time per data point is selected, there's a timing conflict and the software will refuse to sweep.

►►► **Hint:** For measurements in the 0.1...20 kHz range, it is advantageous to manually select the lowest possible IF and use a long measurement time per data point. This avoids interference of RF and IF in the sound card.

- "**Calibrate sampling rate**" button: It is required, that the sound card sampling rate is well known in order to retrieve the data from the correct time spots in the audio stream. Therefore, the sampling rate needs to be compared once to the CPU clock by pressing this control. Note, that a calibration is only possible if a VNWA is connected. The procedure requires 30 seconds.

- Pulldown control "**Reference = Right Channel**": Use this button to control which audio channel the software is to interpret as reference channel.

- Pulldown control "**ignore overload**": The VNWA software detects potential signal clipping in the audio stream. use this control to determine the action to be taken upon an overload condition.

Possible Choices:

-*ignore overload*

-*stop after sweep on overload*

-*immediately stop on overload*

- Pulldown control "**continue on no sync**": This control only shows effect for LPT mode 1 and USB mode, which both use sync bursts. Occasionally, sync bursts can get lost due to other Windows traffic. Select if you want to continue sweeping, restart a sweep or stop sweeping if a sync is lost. Recommended is "restart sweeping".

Possible Choices:

-*stop on no sync*

-*restart on no sync*

-*continue on no sync*

-*silent restart on nosync* In silent restart mode, the "**no sync**" label on the main window is suppressed. This setting might be useful on slow PCs, where e.g. moving a marker might lead to a no-sync condition.

Auxiliary Audio Settings

Select here which of the auxiliary codec's channels is the reference channel.

Auto-Setup Audio Devices button

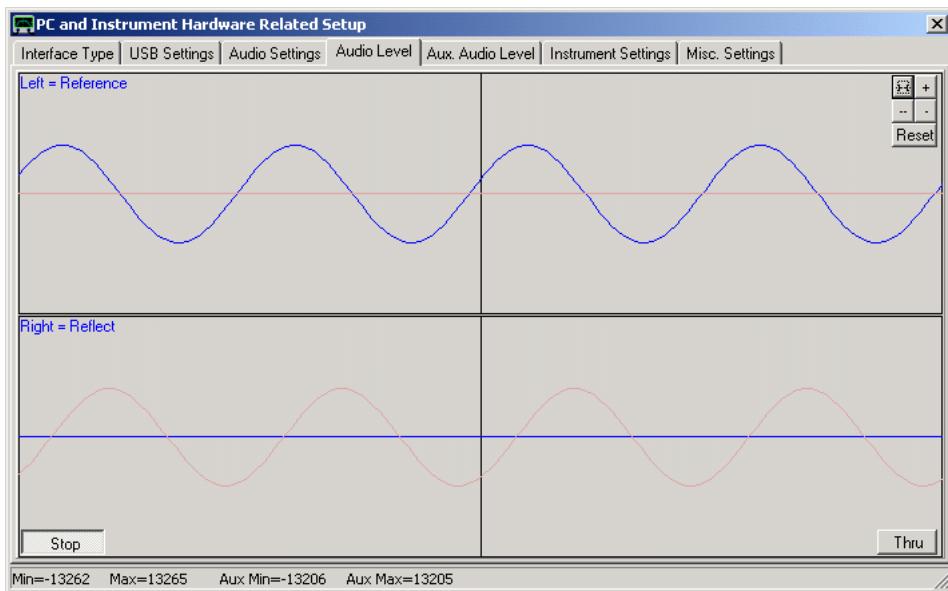
It is quite tedious to go through the audio setup manually. Therefore VNWA software version 36.2.z and newer offers the **Auto-Setup Audio Devices** button.

Pressing this button takes care of the complete instrument setup! More details on the auto-setup feature can be found on page Automatic Configuration of the Software for Usage with a DG8SAQ VNWA (Auto-Setup).

The selection box next to the Auto-Setup Audio Devices button can be used to speed up audio device selection by suppressing evaluation of non-USB sound devices which are normally not used for the VNWA (unless you have modified your hardware to do so).

VNWA SETUP - AUDIO LEVEL TAB

The audio level tab allows to visualize incoming audio data and controlling the recording levels.



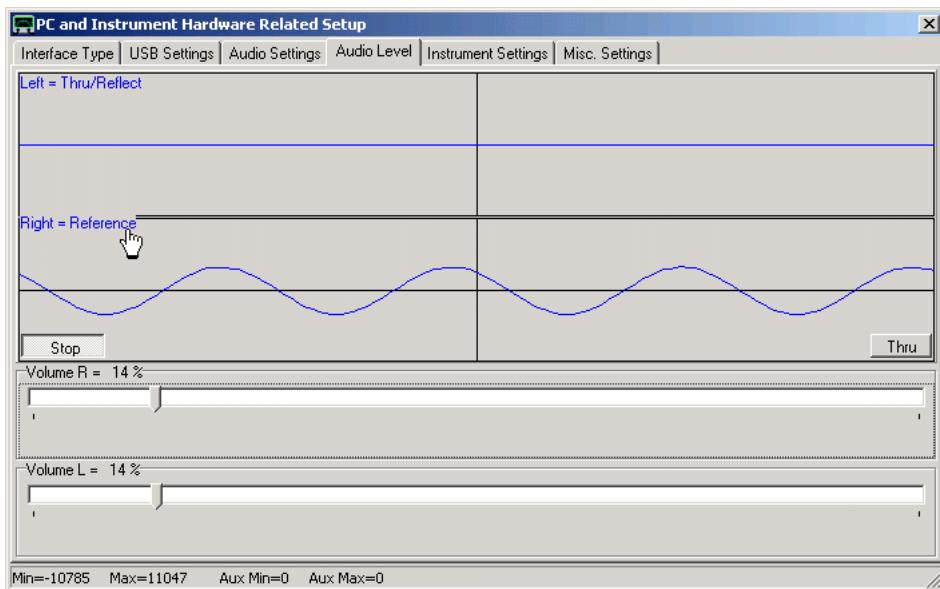
Note, that the faint red traces are only visible if a second auxilliary audio device has been activated. The auxiliary audio levels can also be studied in the "Aux. Audio Level" tab.

Controls:

- **"Start / Stop"** button: Use it to start and stop sampling. If the VNWA is connected, the reference signal should always be visible. In the above example, the reference signal comes in on the lower right channel.
- **"Thru / Reflect"** button: It toggles the instrument between thru and reflect measurement. In the above example, the instrument is in thru mode without a connection from TX to RX. Therefore no thru signal arrives.
- **"Channel labels"**: Left-click onto the blue channel labels (Left=..., Right=...) in order to change the Reference channel selection. Alternatively, this setting can be changed in the "Audio Settings" tab.

+, +++, -, --, Reset buttons: With these controls it is possible to fine tune the VNWA IF frequency. The aim is to have the sine waves move as little as possible. On most machines this adjustment can be omitted. On some machines, S-parameter measurements show very small non-static oscillations, which can be removed by properly tuning the IF.
IMPORTANT: Do calibrate the codec sample rate and the VNWA clock frequency before using these controls, otherwise tuning might make things worse.

If the selected audio capture device is NOT a VNWA2 or VNWA3 internal device, but e.g. the PC motherboard sound device which generally supports hardware capture level control, then additional volume slider controls become visible (not in Windows Vista or later):



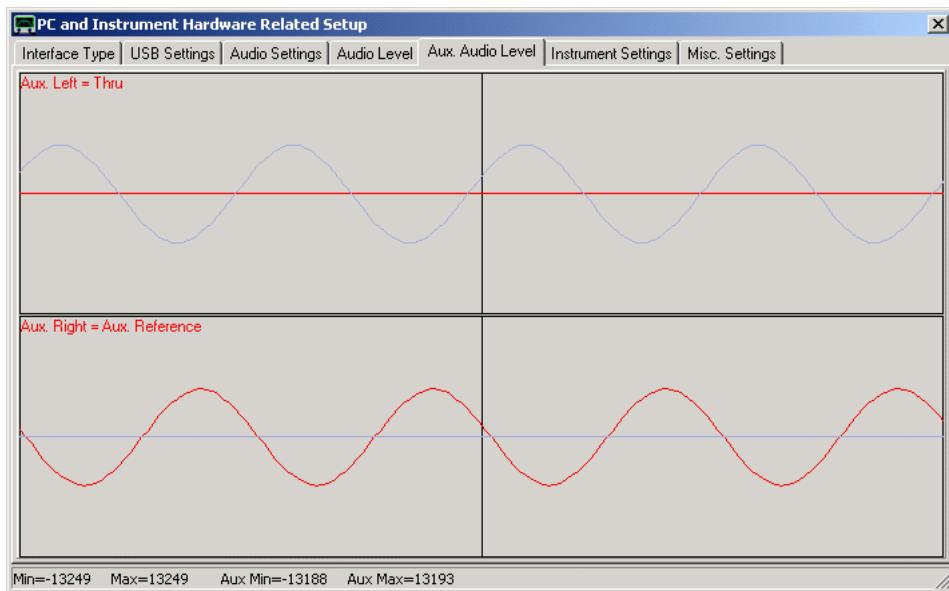
- **Volume sliders:** Use these to control the recording volumes. Make sure that the audio levels are at most about 50% of the displayed vertical span. At 100% level clipping occurs. Note that the difference between 50% span and 100% span is only 6dB. So it is not worthwhile to go to the limit and risk clipping. VNWA will issue an overload warning at 90% level during a sweep already.

►►► **Note:** Volume sliders are not available on Vista and Windows7 / Windows 8 machines. The volumes must be set via the Windows mixer in this case.

►►► **Note:** It is not possible to close the setup window during active sampling. An attempt to do so will stop sampling, but won't close the window. Try to close the window a second time in this case.

VNWA SETUP - AUX. AUDIO LEVEL TAB

The Aux. Audio Level tab allows to visualize incoming audio data on the second auxilliary sound device.



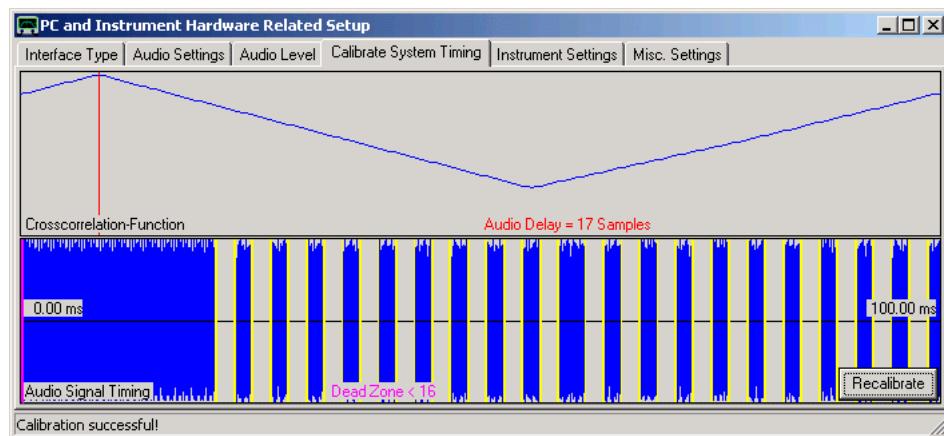
Note, that the faint blue traces are the main sound device signals.

Controls:

- **"Channel labels"**: Left-click onto the red channel labels (Left=..., Right=...) in order to change the Reference channel selection. Alternatively, this setting can be changed in the "Audio Settings" tab.

VNWA SETUP - CALIBRATE SYSTEM TIMING TAB

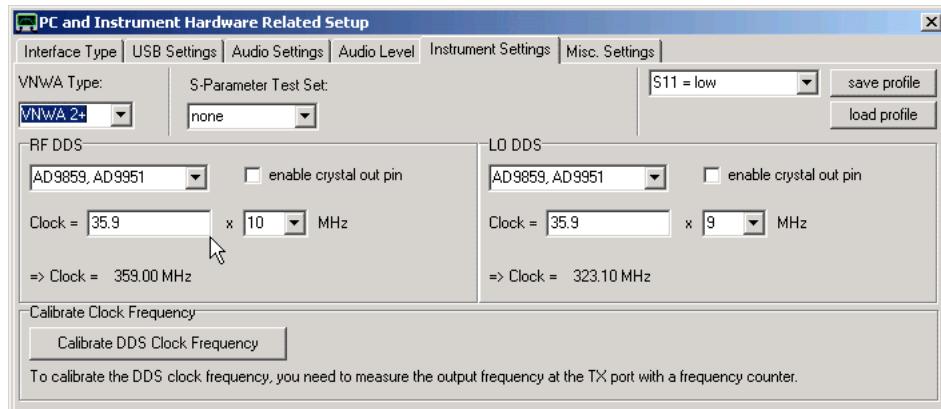
The "calibrate system timing" tab is only visible if LPT mode 0 is selected. It is used to determine the sound driver latency (=audio delay).



Press the "Recalibrate" button and observe the displayed audio delay (17 samples in the above example). Repeat this several times. **If the audio delay varies more than +1 sample over several measurements, LPT mode 0 is not suitable and you must select LPT mode 1.**

VNWA SETUP - INSTRUMENT SETTINGS TAB

The "instrument settings" tab allows to specify the details of the connected **VNWA hardware**.



General controls

- **"VNWA Type"**: For usage with VNWA2.1 upwards select "VNWA2+", for VNWA3.* select VNWA3.
- **S-Parameter Test Set**: Select none if you don't have one. If available, manual or automatic measurement direction control can be selected. Also, the switch response time (=switch delay) and the polarity of the control line can be chosen.
- **"Control line: S11 = ..."** pulldown control: For usage with VNWA2.1 upwards select "S11 = Low" as seen above.
- **"safe profile"** button allows to save the VNWA hardware settings in a *.prf profile file. This is only useful, when the same software is to control several VNWAs with different settings, as the settings are stored in ini-files anyway.
- **"load profile"** button allows to reload a VNWA hardware profile from a *.prf file.

DDS controls

- **DDS Types:**

Select **AD9859, AD9951** on both DDSes for VNWA2.1 upwards and for VNWA3.

- **"enable crystal out pin"**:

Uncheck for VNWA2.1 upwards.

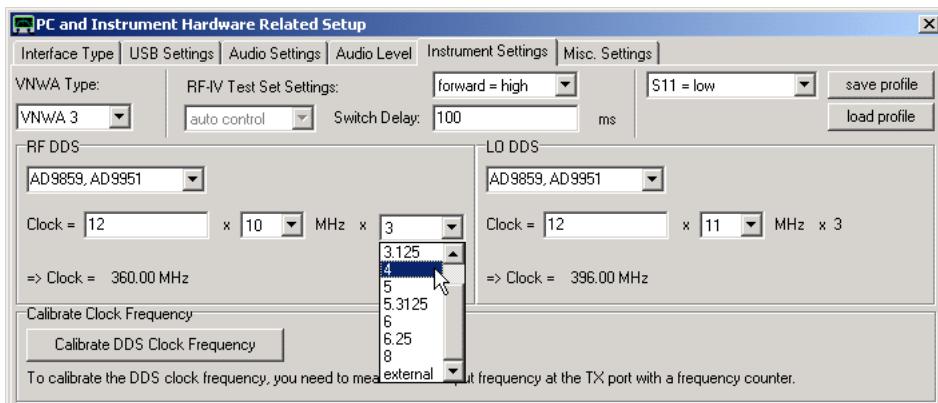
- **"Clock"**:

Enter the DDS input clock, which is **identical on both DDSes** for VNWA2.1 upwards and amounts to about 3x the used crystal frequency.

- **clock multipliers**

must be as high as possible, but different such that the smallest common multiple is as high as possible. Obviously, for VNWA2.1 and upwards 20 and 19 is the best choice in this respect. On the other hand, VNWA electrical power consumption is proportional to the clock multipliers. So, if the VNWA is used in the short wave bands only and power consumption is an issue, use of smaller multipliers can be advantageous.

VNWA3 contains an additional common **premultiplier** for both DDSes which is programmable. It is used to multiply the 12MHz TCXO clock up to 36MHz by default, but other premultipliers can be selected as well:



Also, an external clock source can be selected here. In this case, the common premultiplier is bypassed and the external clock is directly applied to the DDSes. Use a 36 MHz external clock to obtain VNWA2 functionality with VNWA3. Lower external clock rates will lower the usable instrument bandwidth.

Make sure not to exceed DDS core clocks of about 750 MHz. There is no danger of hardware destruction, but the DDS internal clock multipliers will simply not produce any clocks above about 750 MHz and the VNWA won't work.

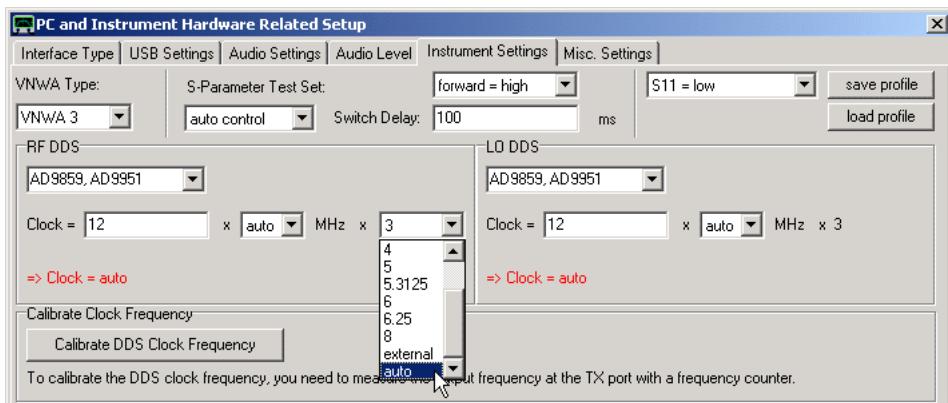
In order to cover the frequency range 600 MHz...900 MHz, dynamical switching of clock multipliers is mandatory. Select clock multipliers "auto" to use this feature. On doing so, the "**Auto Clock Multiplier Settings**" window will open.

Auto Clock Multiplier Settings: Base-Clock = 35.9 MHz				
File	Options			
Seg.#	start f	f/Base-Clock	n LO	n RF
1	0 MHz	0	10	9
2	107.717 MHz	3	20	19
3	574.491 MHz	16	10	11
4	610.397 MHz	17	11	13
5	664.256 MHz	18.5	14	13
6	718.114 MHz	20	15	14
7	771.973 MHz	21.5	17	15
8	825.831 MHz	23	17	15
9	887.643 MHz	25	18	17
10	969.454 MHz	27	19	18
11	1041.265 MHz	29	20	19
12	1292.605 MHz	36	14	15
13				

The "**Auto Clock Multiplier Settings**" window is already prefilled with proven values. Here, the clock multipliers nLO and nRF are specified for a list of frequency segments, where each spans from the specified start frequency to the start frequency of the next segment. Here, you could also specify a low frequency segment from e.g. 0...30 MHz with lower clock multipliers to conserve energy. Note, that you can insert or delete lines by right clicking on a line. Note, that there must not be empty lines. Also note, that you can save and reload this data to or from a *.mul file by using the "File" dialog. The "**Options-Restore default multiplier table**" dialog allows to restore default multiplier settings.

Dynamical switching of VNWA3 premultiplier:

From software version VNWA35.8.p and firmware version 5.13 on, the VNWA3 premultiplier can be dynamically changed during the sweep in addition to the DDS clock multipliers. This greatly increases the number of multiplier combinations for improving signal strength and interference. The **autopremultiplier**-feature is activated by choosing the premultiplier (by default set to 3) to be "auto":



After doing so, an extended clock multiplier table pops up, which allows to select different premultiplier values for different frequency segments:

File	Options	Seg.#	start f	f/Base-Clock	n LO	n RF	premult.	max. clock
1	0 MHz	0	5	7	8	672 MHz		
2	47.1 MHz	3.925	9	7	6.25	675 MHz		
3	72.2 MHz	6.016667	11	10	5.3125	701.25 MHz		
4	94.9 MHz	7.908333	7	9	6.25	675 MHz		
5	111.3 MHz	9.275	10	9	6	720 MHz		
6	140.2 MHz	11.683333	10	11	5.3125	701.25 MHz		
7	158 MHz	13.166667	9	10	6.25	750 MHz		
8	220.6 MHz	18.383333	10	9	6	720 MHz		
9	223.9 MHz	18.658333	10	11	5.3125	701.25 MHz		
10	252.7 MHz	21.058333	20	19	3	720 MHz		
11	281.9 MHz	23.491667	18	19	3.125	712.5 MHz		
12	288.2 MHz	24.016667	11	10	5.3125	701.25 MHz		
13	302.6 MHz	25.216667	11	10	5	660 MHz		
14	326.1 MHz	27.175	10	9	6	720 MHz		
15	352.5 MHz	29.375	10	11	5.3125	701.25 MHz		
16	378.8 MHz	31.566667	20	19	3	720 MHz		
17	394.3 MHz	32.858333	18	19	3.125	712.5 MHz		
18	400 MHz	34.000000	20	19	3.125	720 MHz		

Note, that allowed premultiplier values can be selected from a dropdown box by clicking onto the appropriate cell as shown above.

VNWA contains a default table which is being used if no custom table is available. This table could **remove all major interferers in the frequency range up to 1.3 GHz**. Above, the signal strength could be improved such that some signal can be obtained up to 2 GHz now (with lots of interferers, though).

The multiplier table can be modified and optimized. The modified table will be stored in the files VNWA*.mul in the program directory.

Delete these files if you want to restore the default table values after the next program start or use the dialog "Options-Restore default multiplier table" to immediately restore default multiplier settings.

►►► **Note:** When optimizing the multiplier table, you should avoid the following DDS multipliers: 4, 6, 8, 12, 16. These have shown to cause amplitude instabilities of the DDS chips. There are no restrictions on the premultipliers!

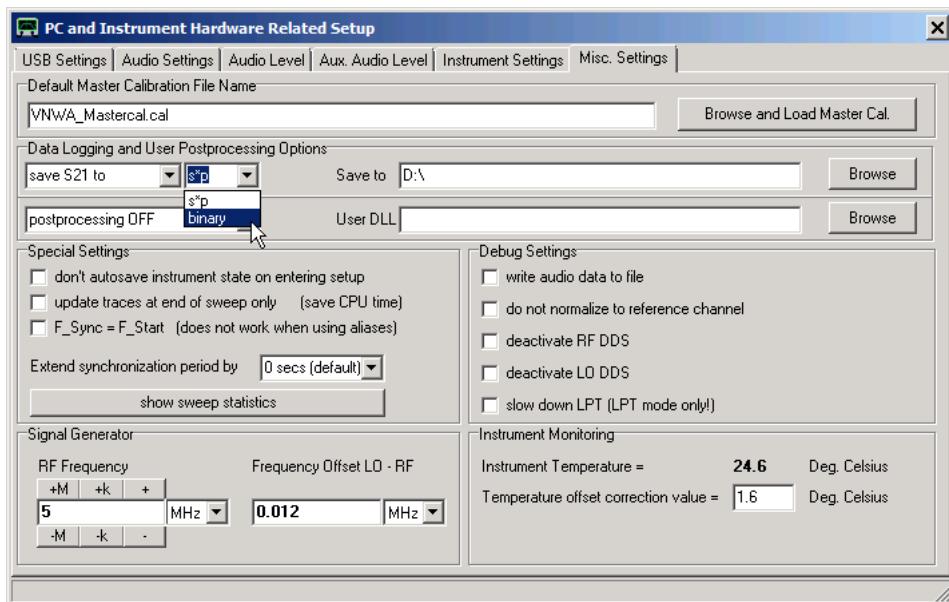
Calibrate Clock Frequency

It is crucial for the precise function of the VNWA, that the DDS clock frequency is accurately specified. If not, the frequency axis unprecise, but worse, the IF is unprecise and even dependent on the measurement frequency. This might cause signal degradation together with the narrowband digital filters, which might show up in oscillatory noise or interference.

The "**Calibrate DDS Clock Frequency**" button allows to calibrate the clock with the aid of an external frequency counter. Pressing it, a 10 MHz signal will be generated at the VNWA's TX port. Measure it with a frequency counter as accurately as possible and enter it into the appearing frequency field. Having done so, press the appearing "Done" button and the software will calculate and update the DDS clocks.

An alternative approach to calibrate the DDS clock to an arbitrary precision frequency source is described on page [Calibrating VNWA System Clock using Frequency Meter](#).

VNWA SETUP - MISC. SETTINGS TAB



"Default Master Calibration Filename":

Here, you can load an existing master calibration file or specify a master calibration filename to be loaded at the next program restart.

"Data Logging Options"

This feature is useful if you want to **automatically save a lot of sweep data for later analysis**.

Here, you can activate automatic data logging, i.e. after every completed sweep cycle the specified data will be automatically saved to a file. Available file formats are Touchstone (*.s1p or *.s2p) or a more storage efficient binary file format (*.v1b or *.v2b).

The uniquely and automatically generated file name may look like "LogData_11_06_2011_22_04_02_123.s1p" where the first three double-digits denote the date and the remaining digits denote the time in hours, minutes, seconds and milliseconds when the saved sweep was completed. Note, that the ordering of the date figures may depend on Windows country and language settings.

You can specify the file path to which data is being saved by directly entering it or by pressing the "Browse" button, but you cannot influence the file name.

Be aware, that every completed sweep cycle will add a file to your hard disk! In order to avoid accidental activation of this feature, a red label in the top right corner of the VNWA main window will warn you when data logging is activated:



The "Misc. Settings Tab" can also be accessed directly by double-clicking above red data logging label.

"Special Settings": The purpose of these special settings is to reduce CPU load and HDD usage for slow machines.

- "*don't autosave instrument state on entering setup*": Normally, the instrument state is automatically saved to the HDD when the setup window is being opened. This might lead to considerable delays when running VNWA on a slow medium like a memory stick or a solid state disk. This delay can be avoided by deactivating autosaving using this feature.

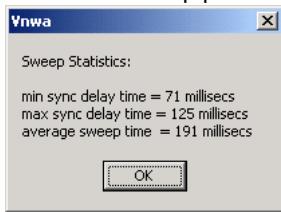
- "*update traces at the end of sweeps only*": Traces on the main window are being updated after every audio block that is coming in from the sound codec. For very large numbers of data points this is very CPU intensive, particularly when a small audio block size is being chosen. This might interfere with proper VNWA operation on slow machines, particularly in combination with data logging to slow media.

Activating this feature will dramatically reduce CPU loading at the expense of the main graphics window being updated only once per sweep at its end. Also, the progress bar will be suppressed.

- "*F_Sync = F_Start*": This enforces the sync pulses to be generated on the start frequency and not on an optimized frequency. This is useful when measuring extremely high Q low frequency components like kHz crystals. If the sync pulse frequency is not equal to the start frequency, the sync pulse will cause ringing in the device under test which will lead to interference with the measured data if the DUT ringing time is comparable to the sweep time. Note, that this setting will not work when measuring with high frequency aliases, i.e. at measurement frequencies >500MHz.

- "extended synchronization period by...": On slow or heavily loaded machines the operating system might occasionally delay dispatching the start sweep command to the VNWA hardware. This might lead to a loss of sync if the VNWA software isn't aware of it. Use this feature to make the VNWA software aware that such a situation might arise. You can extend the synchronization period up to 125 seconds. Note, that if sync really gets lost, VNWA will not be responding for this extended period.

- "show sweep statistics": With this feature you can check the actual delay of the sync signal experienced over the last continuous sweep period as well as the average time per sweep:



If the maximum sync delay time reaches 500 milliseconds or one experiences occasional losses of sync, then one might consider to extend the synchronization period by a value larger than the maximum experienced sync delay time. Above statistics were collected with VNWA3 and default settings on a virtual XP system in VMWare player run on a Windows7 machine, which works very well.

"Debug Settings":

These are used for VNWA software development and for educational purposes only!

►►► **WARNING:** Don't touch these debug settings unless you exactly know what you are doing! **For proper VNWA operation, all checkboxes must be unchecked!**

Options:

- "write audio data to file": If selected, the raw stereo audio stream data is stored after each sweep to the file out.dat. => requires considerable processing time. A good free software tool to view the raw audio data is the plotting software GENPLOT.
- "do not normalize to reference channel": If selected, the reference channel data is ignored. This is useful to evaluate absolute signal amplitudes of the DDSes and noise levels.
- "Deactivate RF DDS": If selected, the RF DDS is shut down. This is useful to check the noise floor level of the instrument undisturbed by RF crosstalk. Requires USB firmware V4.2 or higher.
- "Deactivate LO DDS": If selected, the LO DDS is shut down. This is useful to check the noise floor level of the IF signal path. Requires USB firmware V4.2 or higher.
- "slow down LPT": If checked, communication via LPT is slowed down by a factor of 2.

►►► **Note:** The last two options show no effect on USB controlled VNWAs.

"Signal Generator"

This feature is intended for testing proper DDS operation. There is a more convenient signal generator feature available on the VNWA main window.

With the *RF Frequency* control, you can directly control the frequency of the RF DDS. The *Frequency Offset LO - RF* control sets the frequency of the LO DDS accordingly.

Example 1:

If RF is set to 10 MHz and an offset of 1 MHz is selected, then the RF DDS is set to 10 MHz and the LO DDS is set to 11 MHz.

Example 2:

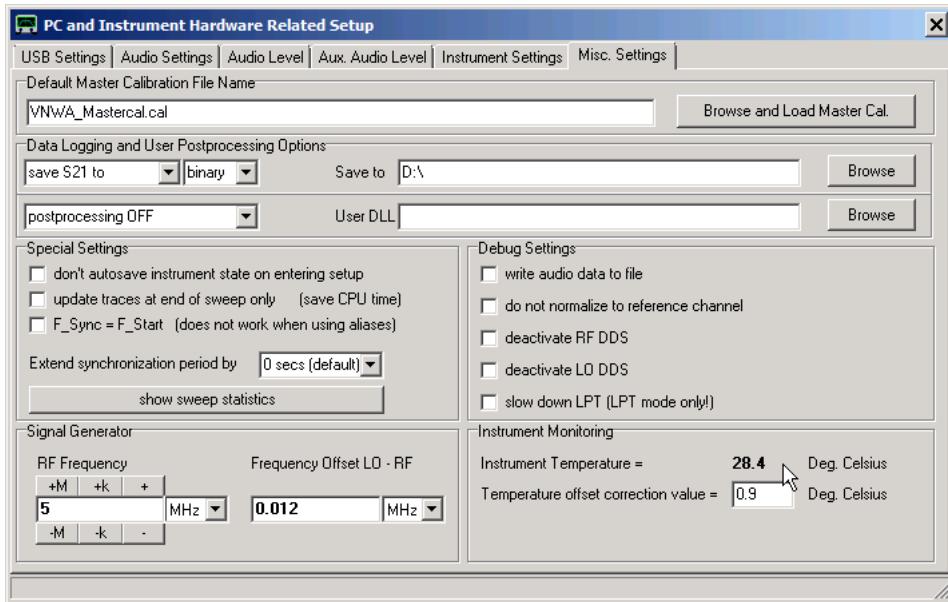
If RF is set to 5 MHz and an offset of -5 MHz is selected, then the RF DDS is set to 5 MHz and the LO DDS is set to 0 MHz, which means it is switched off.

Example 3:

If RF is set to 0 MHz and an offset of 5 MHz is selected, then the RF DDS is set to 0 MHz (=off) and the LO DDS is set to 5 MHz.

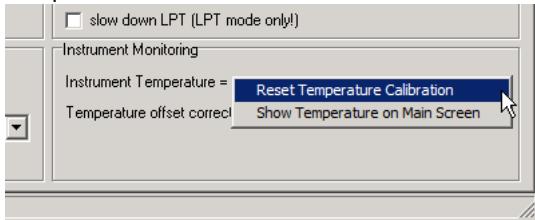
"Instrument Monitoring" (only VNWA3 with VNWA35.8.z and firmware v5.16 or later)

VNWA3 firmware version v5.16 or later allow to measure the internal VNWA temperature. If availability of this feature is detected, the temerature will continuously be updated in the "Instrument Monitoring" section (mouse pointer below):

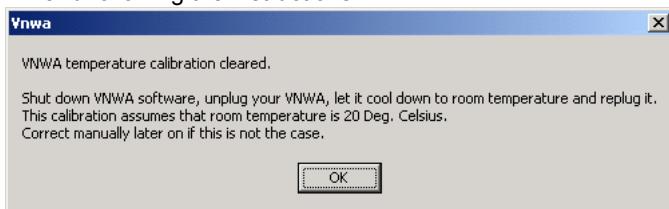


As the temperature sensor is not calibrated and not very accurate, an individual offset can be applied to improve the readings. Initially, the sensor is calibrated assuming the VNWA is exposed to 20 deg. Celsius at first start. This calibration can be redone by

1. resetting the temperature calibration by right-clicking onto the "Instrument Monitoring" panel and selecting "Reset Temperature Calibration"...

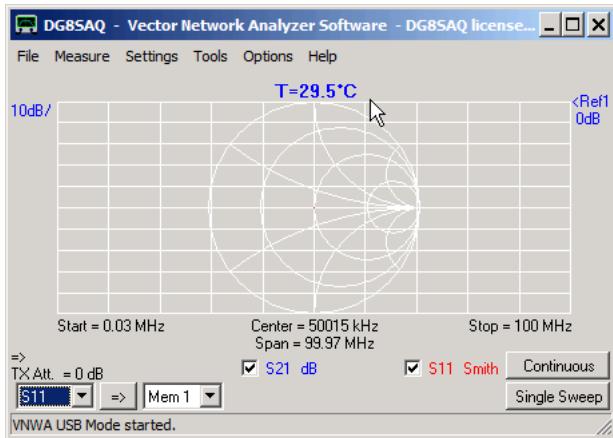


2. ...and following the instructions:



►►► **Note:** When data logging is activated using binary (*.v*b) file format, the current VNWA temperature will be logged into the data files together with the measured S-parameters.

►►► **Note:** When "Show Temperature on Main Screen" is activated, the VNWA temperature can be monitored continuously on the VNWA main window:

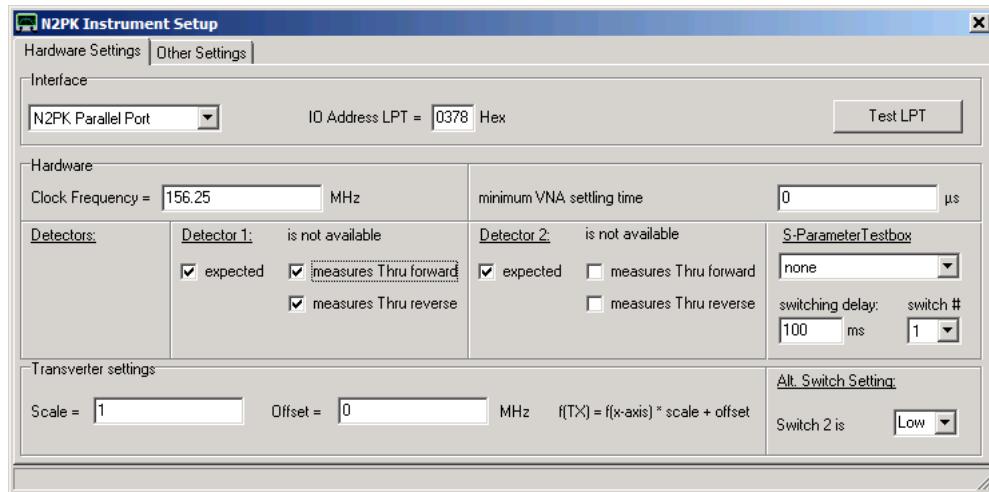


This feature can be switched off again by deactivating it in the misc settings setup or by right-clicking the blue temperature label on the main screen.

OPTIONS - SETUP (N2PK-VNA)

The "N2PK Instrument Setup" window allows to configure VNWA.exe to control an N2PK-VNA.

Hardware Settings Tab



In the **Interface** section, select the interface of your choice (either **N2PK Parallel Port** or **G8KBB USB Interface**) and in the former case the port address. The "Test LPT" button allows to test the LPT speed. The result will be displayed in the very bottom status line and should be <2 us.

The **hardware** section specifies the **VNA clock frequency** and the available **detectors** and how they are assigned. The above setup shows Detector 1 being used for Forward through (S21) and Reverse through (S12) while Detector 2 in turn is being used for Forward reflection (S11) and Reverse reflection (S22). This can be changed according to individual external wiring of your N2PK VNA. Note, that this might not be the best assignment of detectors. You can also specify if an **S-parameter test set** (automatic switch to invert the DUT) is available and if it should be used manually (user select measurement direction) or automatically (direction is switched automatically). Also, the **polarity** of the test set control can be selected. As most switches are built with slow mechanical relays, the software needs to wait until the switch is thrown. Specify the **switching delay** accordingly.

The **minimum VNA settling** time is not exactly a hardware setting. It specifies, how long the software waits after a DDS phase change for the VNWA to settle. 200us are sufficient. Entering 0 will invoke the minimum settling time influenced by the LPT transfer speed and can range from 100us to 200us.

The "**Transverter settings**" section specifies, how the main VNWA window frequency axis is to be scaled, if a frequency transverter is used together with the VNA. E.g. if scale=0.5 and offset=-400MHz, a specified measurement frequency range of 430...440 MHz will invoke a DDS output frequency of 15...20 MHz.

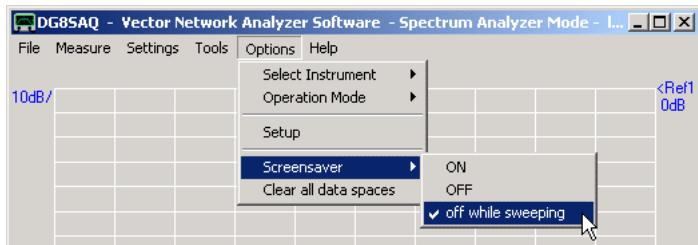
The G8KBB USB Interface allows to control two switching lines, one of which is used for controlling the S-parameter testset (usually switch #1, as is selected above). The other switch can be controlled with the "**alt switch settings**" to e.g. control additional test heads. Note, that this feature is only functional in USB mode.

Other Settings Tab

Here, you can load an existent master calibration file or specify a master calibration filename to be automatically loaded on the next program start.

OPTIONS - SCREENSAVER

The menu item "Screensaver" allows to configure the screen saver if a Windows screensaver is activated.



Options:

ON: The screensaver is permanently active and switch the screen off after a while of inactivity.

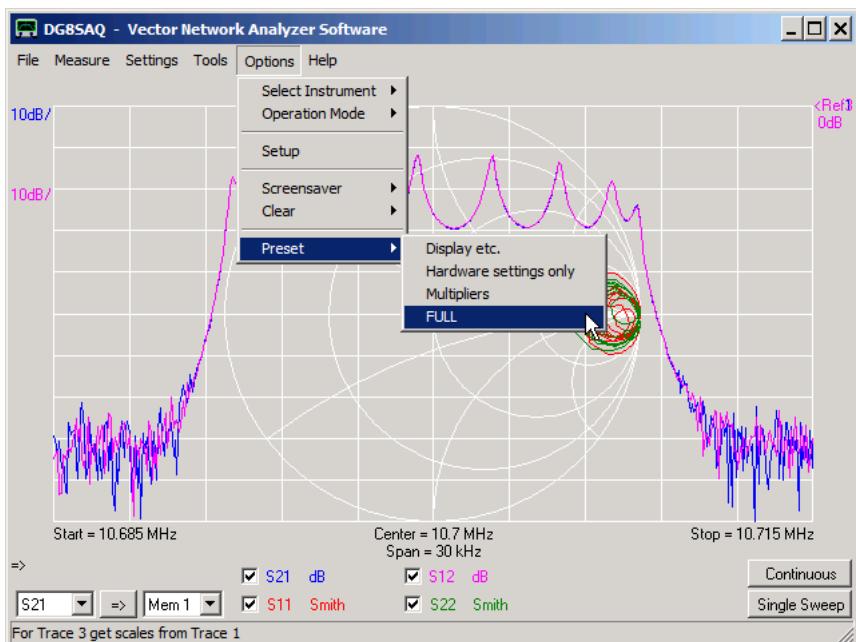
OFF: The screensaver is permanently disabled, the screen will never be switched off.

off while sweeping: The screensaver is disabled when VNWA is sweeping, such that the screen won't get switched off while measuring, but it will be switched off after a while of inactivity if VNWA is in idle mode.

Note, that these settings are local settings inside VNWA and affect the Windows screensaver only while the VNWA software is running. These settings have no effect at all if the Windows screensaver is disabled via the Windows control panel.

OPTIONS - PRESET

The **Preset** menu allows to bring the VNWA software back into a default state.



Options:

- **Display etc.:** resets all non-hardware related settings like display setup, custom traces ...
- **Hardware settings only.:** resets the hardware setup.
- **Multipliers:** resets the clock multiplier tables to default values.
- **FULL:** returns the software to a fresh installation state.

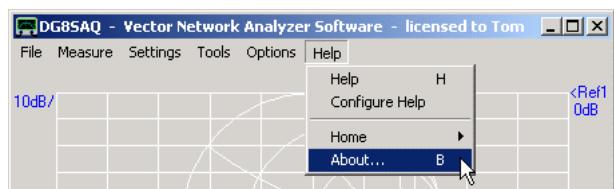
►►► **Note:** Any preset will cause the VNWA software to terminate. The preset will take effect with the next software launch.

►►► **Note:** Any preset will preserve the VNWA license key.

HELP

The VNWA main menu "**Help**" offers the following functions:

Help
Configure Help
Home
About



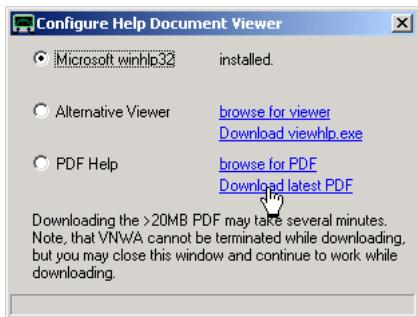
HELP - HELP

Select "help" to invoke **this help file** from within VNWA.
VNWA.HLP and VNWA.cnt must reside in the VNWA software directory in order to use this feature.

If "PDF Help" is selected in Configure Help, then an appropriate PDF Help file is required instead of the *.hlp file package.

HELP - CONFIGURE HELP

As Microsoft does not or only with delay provide viewers to view help files in *.hlp format for new Windows versions, you may choose here to use a different viewer or use the pdf help file instead. This menu also informs if the Microsoft winhlp32 viewer is actually installed.



- A possible alternative viewer is viewhlp by Alan M0PUB, which is similar to winhlp32, but lacks some of its features. It can be downloaded from the SDR-Kits website by clicking on the blue "Download viewhlp.exe" label.

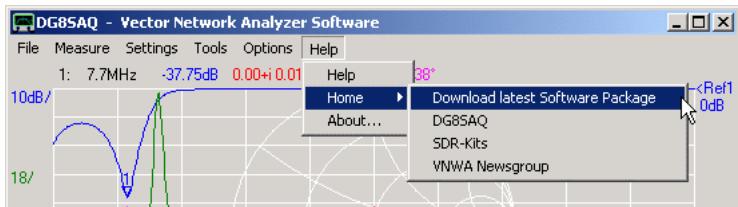
- If viewhlp is actually installed, the blue "browse for viewer" label will indicate "viewhlp.exe" instead. Clicking on this label will launch a *.exe file open dialog. Here, you can select the executable of any help file viewer you like to use. If you have an old Windows XP license, you may decide to use the XP winhlp32.exe instead on your new Windows7 or Windows8 machine, which works very well. Copy the executable to the VNWA installation folder (or any other folder) and select it here.

- Clicking on the blue "browse for PDF" label will launch a *.pdf file open dialog. Select the pdf file, that is to be opened later when you click the Help menu. After a pdf file has been selected, the label will show its name.

- Clicking the "Download latest PDF" will launch the download of the latest PDF help file from the SDR-Kits website as a background process. Note, that the download cannot be interrupted.

HELP - HOME

Select "Home" to access the **VNWA knowledge base**:



Default Links

Download latest Software Package will let you access the VNWA self installer binary for software, driver, firmware and help file updates.

DG8SAQ will let you access DG8SAQ's webpage to e.g. to look for literature and links.

SDR-Kits will let you access Jan G0BBL's ordering and support webpage.

VNWA Newsgroup will guide you to the VNWA Groups.io forum, where you can post your questions and find software and driver updates.

Customizing Links

The internet links in this menu can be customized by creating an ASCII file with name "**links.url**" in the VNWA program directory. Note, that the file extension is .url, and NOT .txt.



Every line in above file denotes a menu item with corresponding internet link. Note, that any text preceding "http" is interpreted as link name to be shown in the "Help-Home"-menu. The above file will modify the menu like this:



HELP - ABOUT

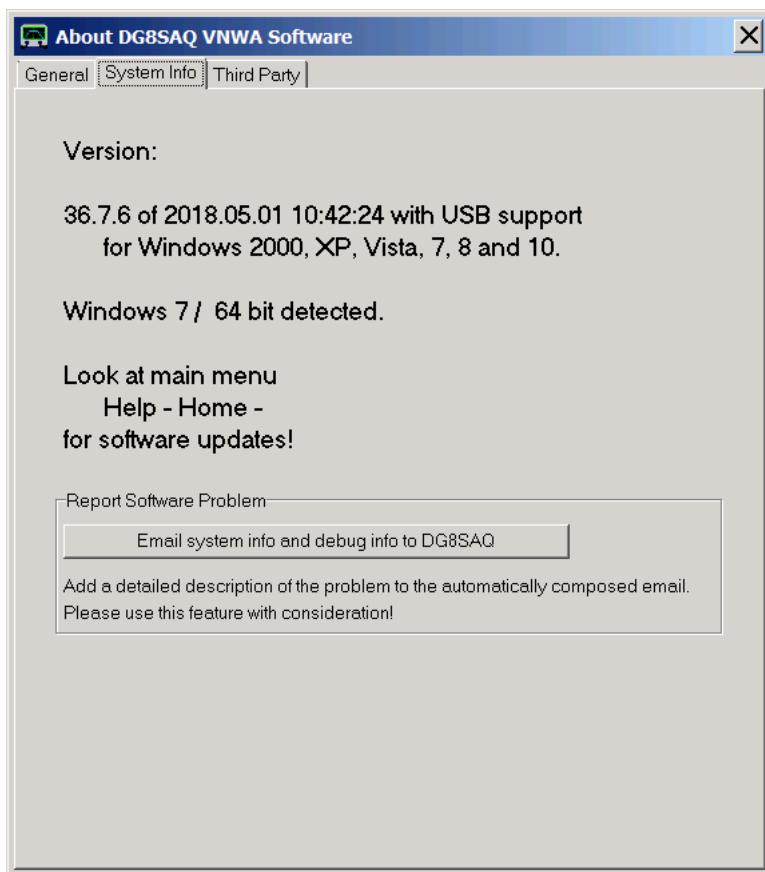
General Tab



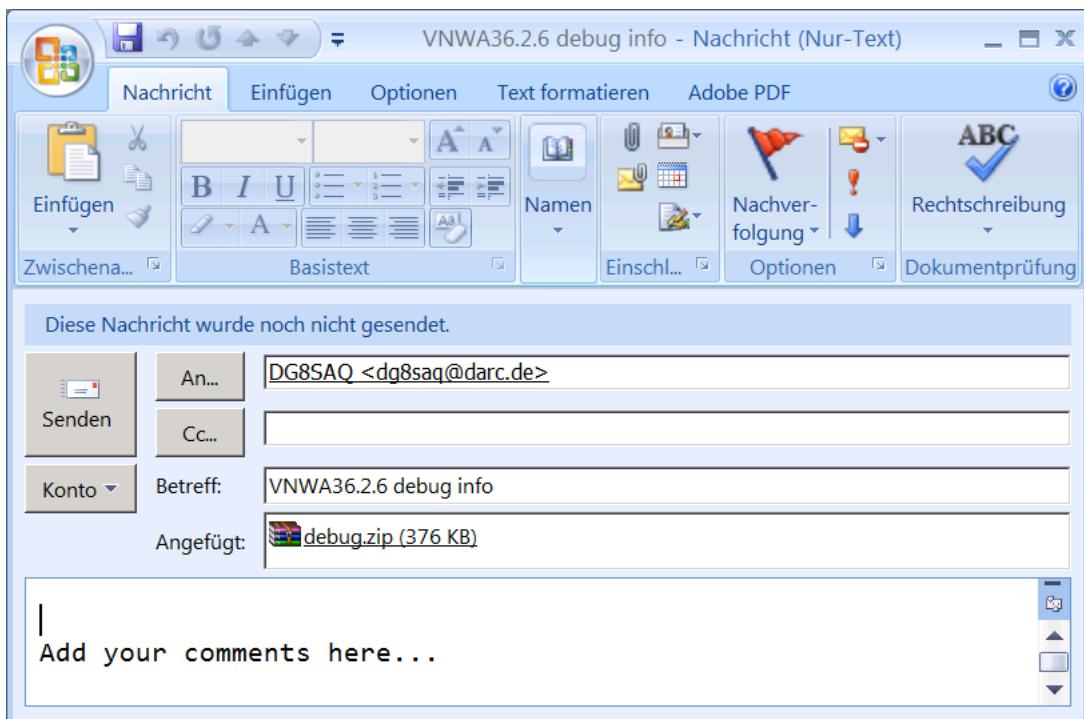
Note, that the "featured by" logo on the bottom of above screen can be customized / changed by replacing the image file "logo.bmp" in the VNWA program directory. If logo.bmp is removed, the "featured by" text and logo will not be shown.

System Info Tab

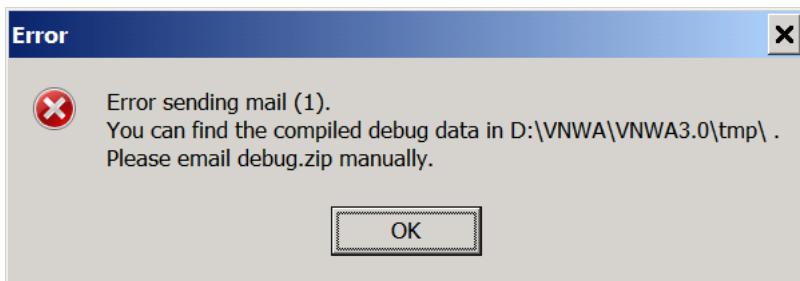
Look here to determine your software version. Here you can also automatically **generate a debug report by pressing the "Email system info..." button:**



Note, that the email is only generated if you use Microsoft Outlook or Outlook Express. Add your own comments to the automatically generated email and send it off. But please use this feature with consideration, so I don't drown in emails.

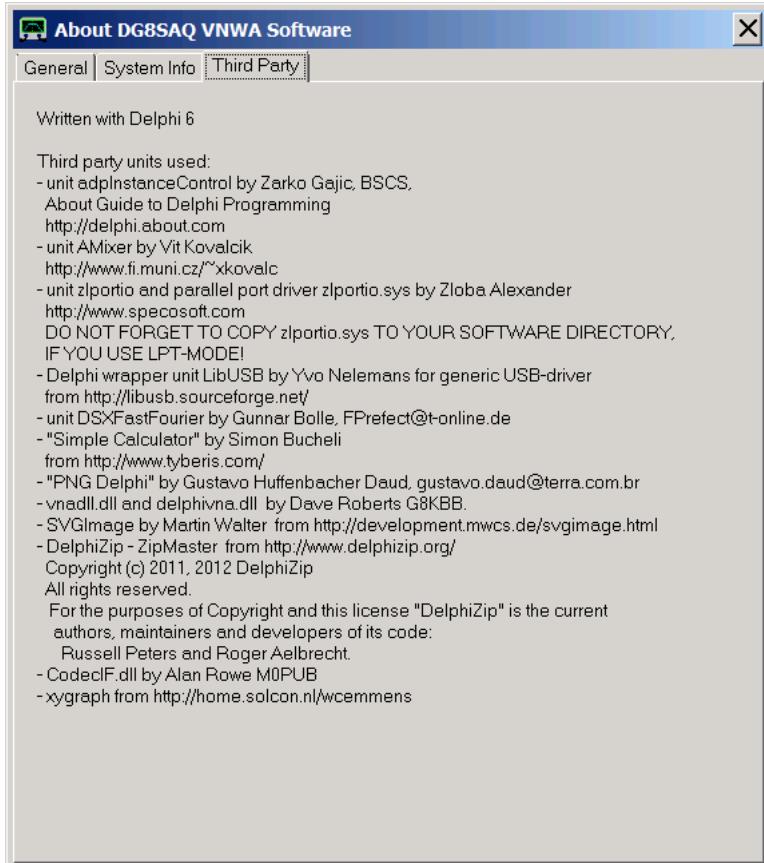


If you use a non-Microsoft mailer or if you decide to cancel the email, you will see an error window with the information on where the debug data has been saved to your disk, so you can mail it manually later.



Third Party Tab

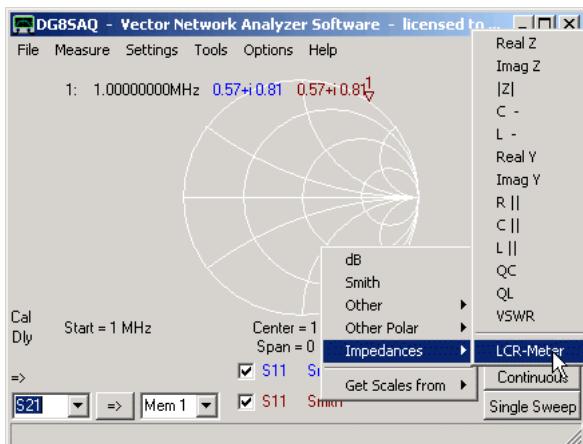
Thanks to all who contributed to this software!



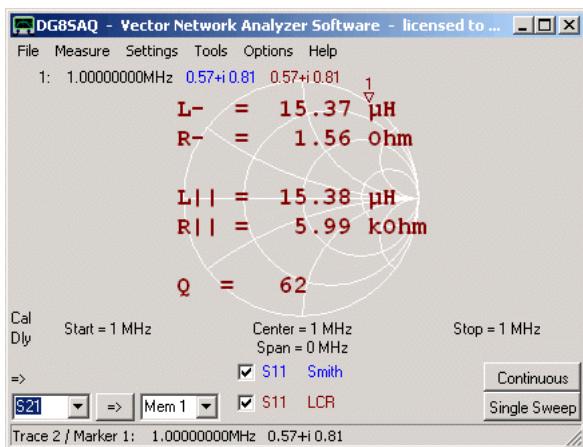
LCR METER

The VNWA + Software can be configured such, that it behaves like a standard LCR-meter being able to measure a complex impedance at a fixed frequency, decompose the result to parallel and series equivalent circuit and display the result numerically.

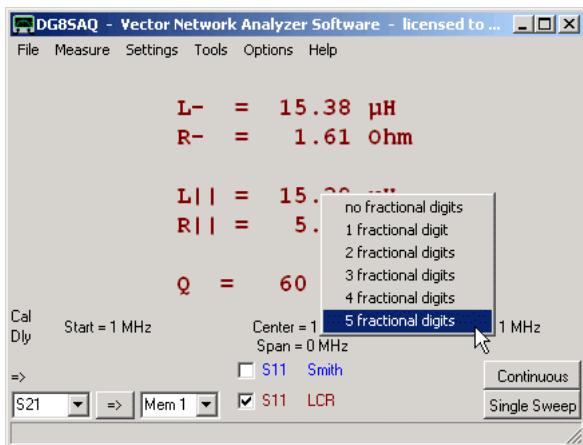
- To do so, first set the VNWA center **frequency** to the desired value and set the span to zero. Any other span will work, too, but as the evaluated numbers will be averages over the full sweep, these will also be frequency averages if a non-zero span is used.
- Set the number of data points and time per data point such, that a **sweep rate** of 1...10 sweeps per second will be performed. This will be the rate, at which the LCR and Q-values will be updated.
- Next perform a **SOL-calibration** in order to accurately measure S11, and if desired, move the calibration plane by means of a **port extension** to where you would like it to be, e.g. to the tip of a test fixture.
- Next, select **display mode LCR-Meter** for an S11 trace as seen below.



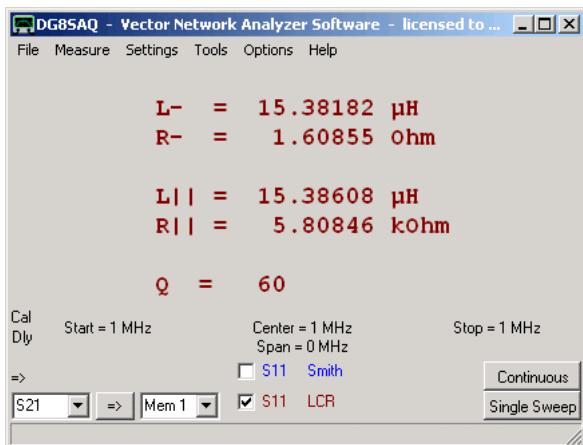
As can be seen below, numerical values appear on the main window. They display **L,C,R and Q-values** for parallel and serial equivalent circuit of the impedance to be connected to the TX port and to be measured into S11. Sweep continuously and change the test object on the TX port. You will see how the displayed values change after every sweep. The software will automatically detect if the test object is capacitive or inductive and chose to display Henries or Farads accordingly.



The number of displayed **fractional digits** can be controlled by **right-clicking onto the displayed numbers** as can be seen below. Note, that all other diagrams must be disabled in order to do so.



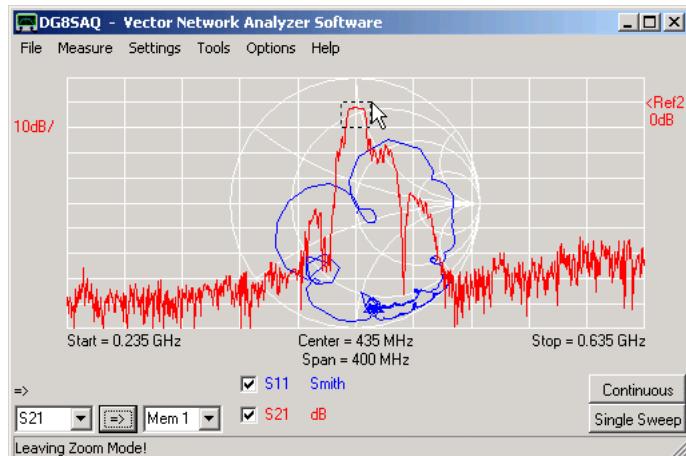
The above action will cause 5 fractional digits to be shown (below) instead of 2 fractional digits (above).



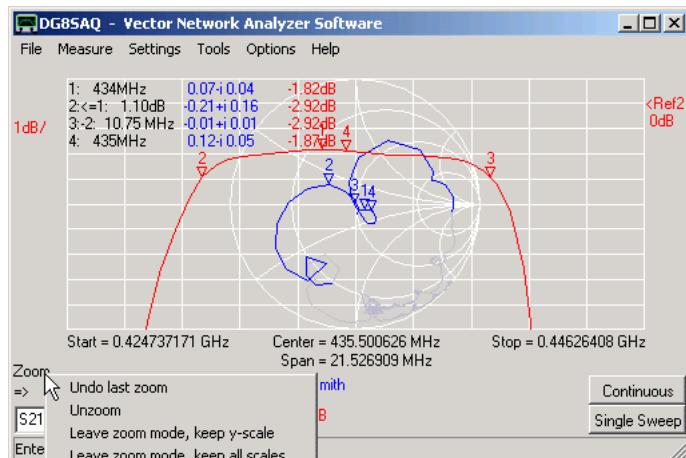
►►► **Note:** As inductivity and capacity are frequency dependent effects, that tend to zero as the frequency approaches zero, L and C measurements will become more and more inaccurate the lower the measurement frequency.

ZOOM FUNCTIONALITY

You can zoom into a displayed trace in order to view an enlarged section of it. To do so, point the mouse cursor to one corner of an imagined box on the main window grid, which you want to enlarge, press the left mouse button and draw the so called zoom-box with the mouse keeping the left mouse button pressed continuously. While you keep the left mouse button pressed and you move the mouse, you will see the zoom-box taking shape. In the following example, we attempt to zoom into a measured filter's passband:



As soon as you release the mouse button, the zoom-box will be enlarged to completely fill the main grid.



►►► **Note:** You can do multiple **consecutive zooms**.

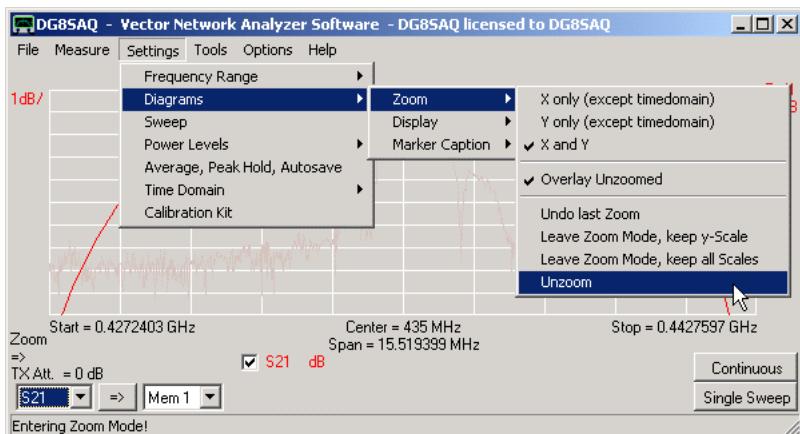
►►► **Note:** When zooming, the measurement span and measured number of data points remains untouched. Only part of the data is displayed. When sweeping in zoomed state, you still sweep the whole unzoomed frequency range, but only part of it is displayed.

►►► **Note:** You cannot zoom inside the Smith chart. But as you can see in above screenshot, the Smith chart data outside the zoomed frequency range is greyed.

►►► **Note:** You can **unzoom** by right-clicking the **zoom label** near mouse pointer. If you right-click it, the above shown **unzoom menu** pops up.

►►► **Hint:** You can also unzoom by right-clicking the main window grid.

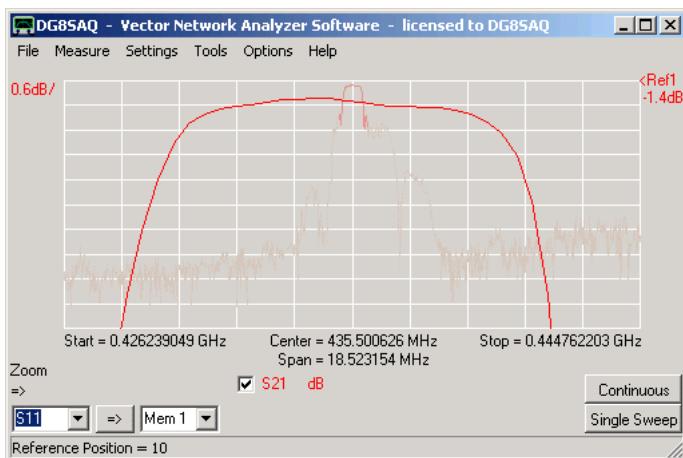
►►► **Hint:** Zoom functions can also be accessed by the main menu "**Settings**"-"**Diagrams**"-"**Zoom**".



Choices:

- **Y only ...** = only zoom vertically, leave the frequency axis unchanged (not applicable in time domain mode).
- **X and Y** = zoom frequency axis and y-axis

- **Overlay Unzoomed** = If selected, a greyed version of the unzoomed data is displayed with the zoom range highlighted together with the zoomed trace:



The following choices are only visible in zoom mode:

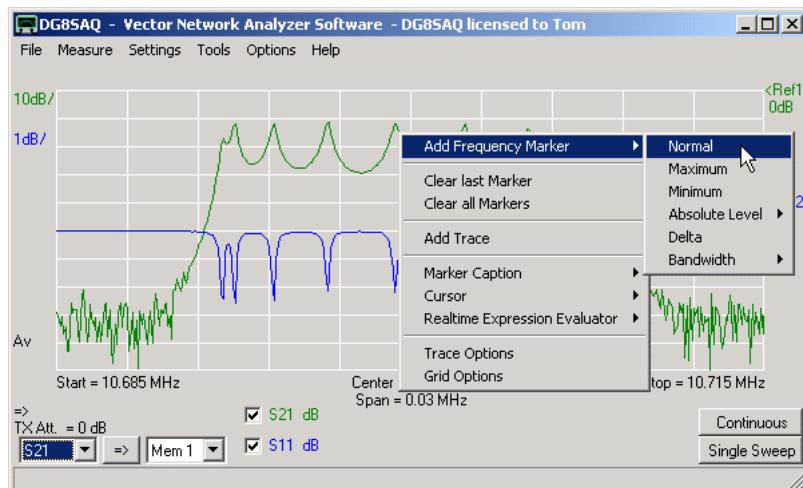
- **undo last zoom** = restore the x- and y-scales of before the last zoom
- **leave zoom-mode keep y-scale** = restore the full frequency span but keep the zoomed vertical scale
- **leave zoom-mode keep all scales** = keep the zoomed frequency span and the zoomed vertical scale. The data outside the last zoom box is lost, the visible data is interpolated to the full number of data points grid. A sweep after this will only sweep the visible frequency span.
- **unzoom** = restore the original x- and y-scales of before all consecutive zooms

MARKER SUMMARY

In order to read numerical information from displayed traces, up to **9 markers can be placed** onto the traces.

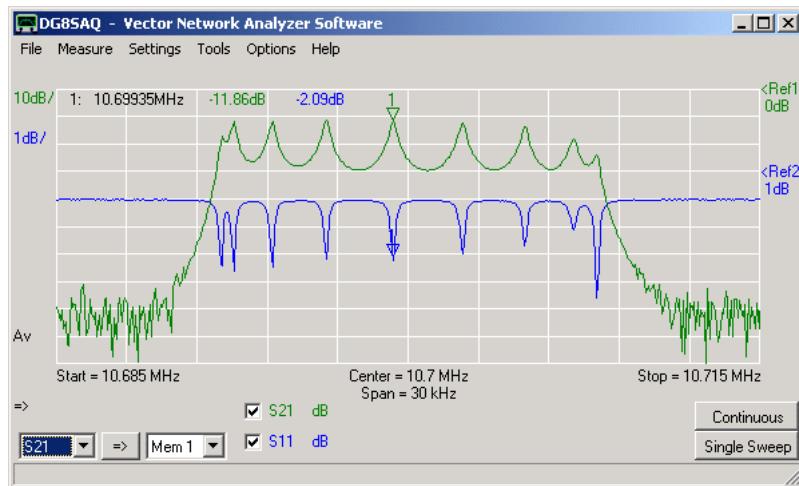
1. Adding a marker

This is done by pointing the mouse cursor to the position (=frequency) in the plotting grid where you want the marker to be and right-click the mouse. A popup menu allows to select a marker type to be placed:



Note, that with the aid of above menu you can also delete the most recently placed marker ("Clear last marker") or delete all markers ("Clear all markers").

In the following a "Normal" "Frequency Marker" is placed:

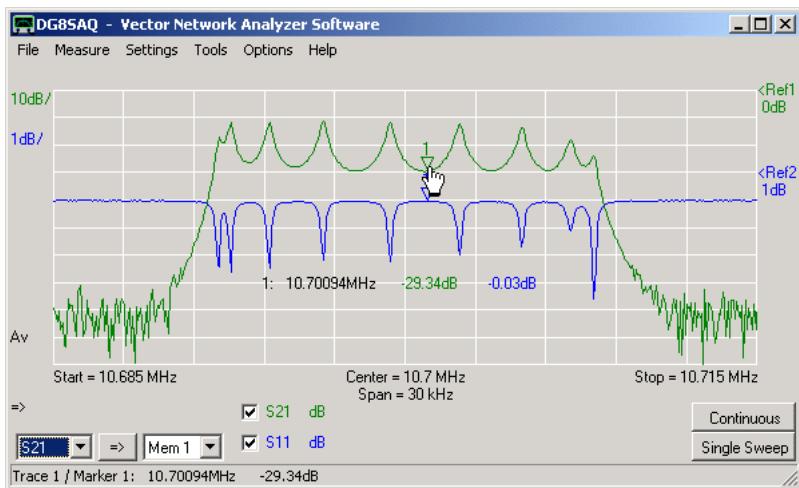


Note, that in addition to the actual markers a **marker caption** with marker number, marker frequency and marker values is displayed in the top left area of the plotting grid.

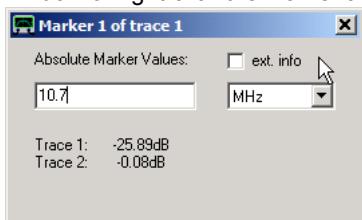
The user settable frequency unit of the center frequency is used as marker frequency unit (MHz in above example).

2. Moving a marker

Markers can be moved with the mouse. To do so, point the mouse to the marker which you want to move. If the mouse pointer changes to a hand shape, press the left mouse button and drag the marker to where you want it to be:



If the marker is to be **set to an exact frequency**, then double-click the marker to open the extended marker info window or right-click the marker and select the extended marker info menu:

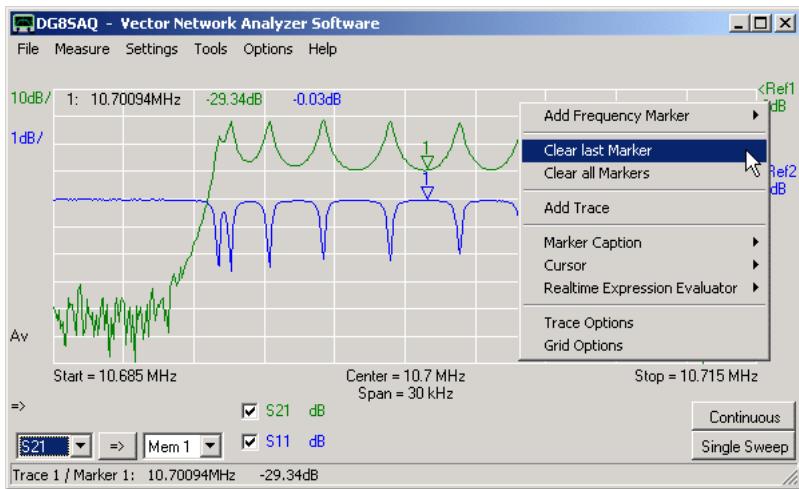


The marker frequency can be adjusted here and thus the marker will be moved accordingly.

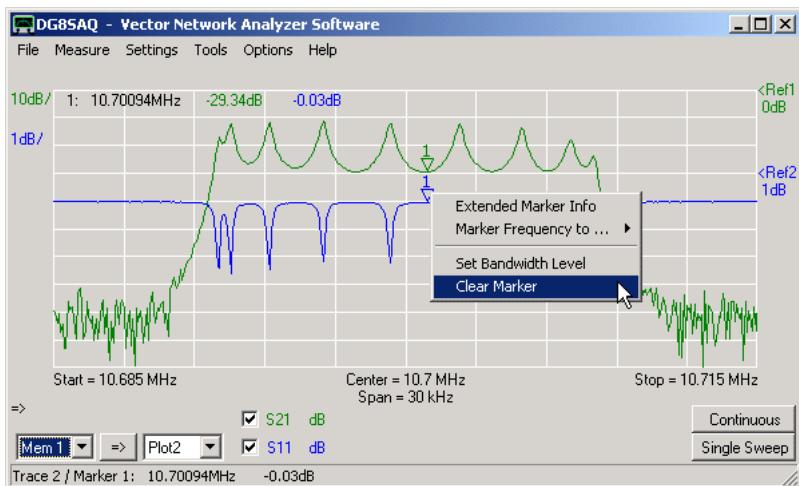
3. Removing markers

There are several ways to remove markers.

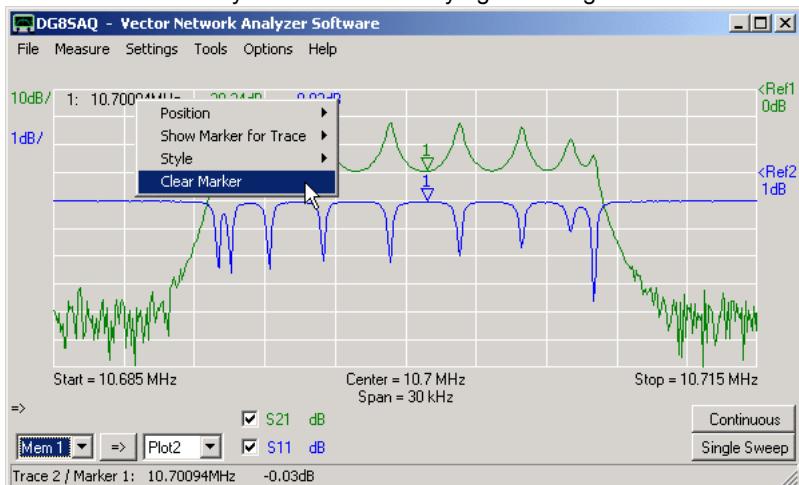
The last marker or all markers at once can be removed by right-clicking the main window grid and selecting "**Clear last Marker**" or "**Clear all Markers**":



A specific marker can be removed by right-clicking that marker and selecting "**Clear Marker**":



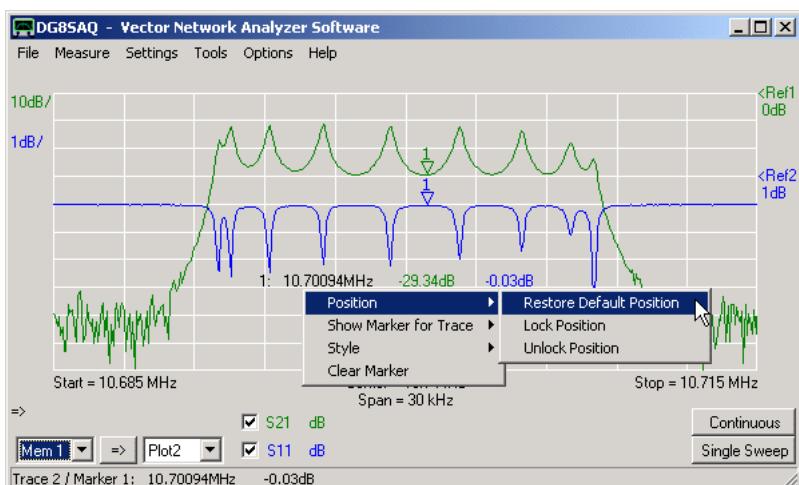
The same functionality can be achieved by right-clicking a text line of the marker caption and selecting "**Clear Marker**":



The marker belonging to that specific text line will be removed.

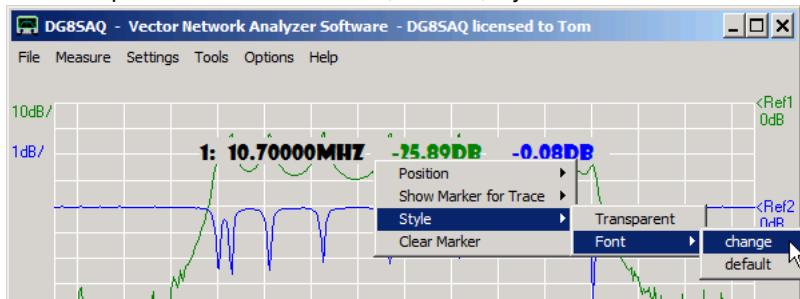
4. The marker caption

The marker caption is the text informing about marker frequencies and marker levels. It can be moved with the mouse in a similar fashion as the markers. To do so, point the mouse to the frequency part of the marker caption. When the marker pointer changes to a hand shape, press the left mouse button and drag the caption to where you want it. The default position can be restored by right-clicking the marker caption. Also, the current position can be locked or unlocked:



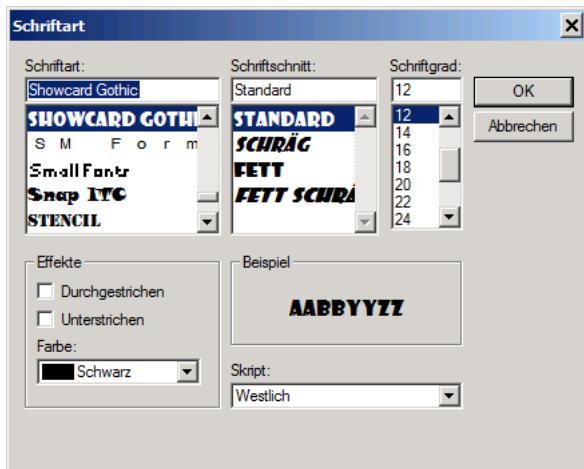
The same functionality is accessible via the main menu "Settings"- "Diagrams"- "Marker Caption".

►►► **Note:** For better readability the marker caption can be switched to non-transparent, i.e. graphic elements behind the marker caption will be invisible. Also, the font, style and the size of the marker caption can be changed:



This functionality is accessed by right-clicking the marker caption as seen above for the case of deactivated transparency.

Upon clicking the **Font - change** menu, the font manager will open:



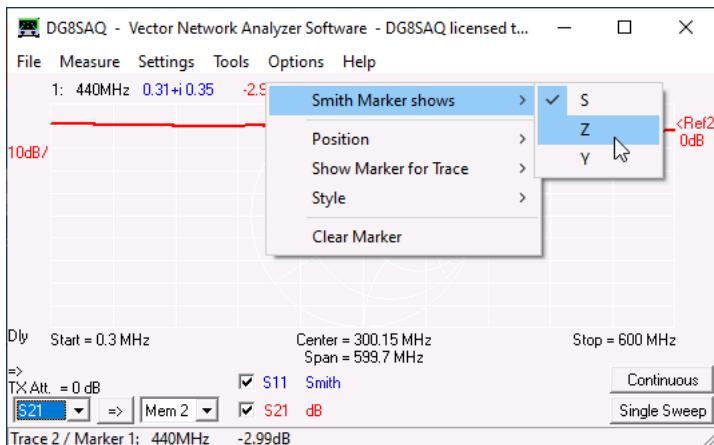
Here, the font, style and size can be selected. Note, that the color setting will have no effect here.

►►► **Note:** Left-clicking any part of the marker caption will copy the complete marker caption to the Windows clip-board, from where it can be pasted into any document, e.g. like this:

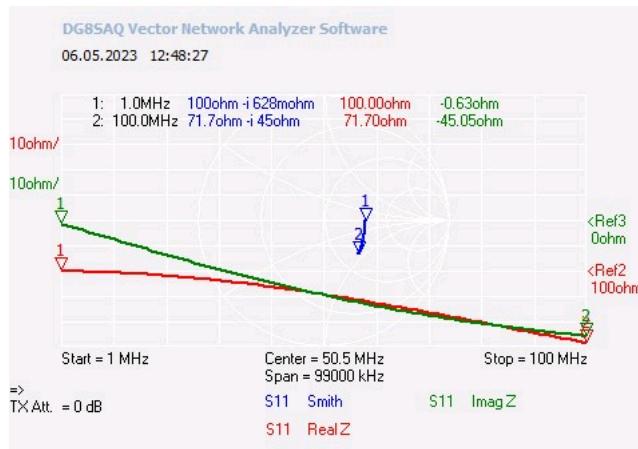
1: 0.01856MHz 0.99-i 0.01 0.01+i 0.00 0.00 -90.00°
 2: 22.99MHz 0.46-i 0.72 0.01+i 0.00 0.00 -90.00°
 3: 99.93MHz -0.34-i 0.58 0.01-i 0.00 0.00 -90.00°

Smith Marker Captions:

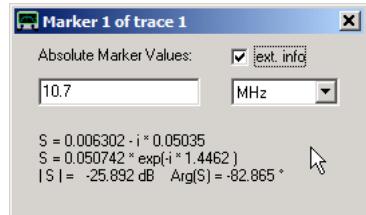
Smith markers can be evaluated as reflection coefficients S, impedances Z or admittances Y. This is set by right-clicking the marker caption or a marker symbol:



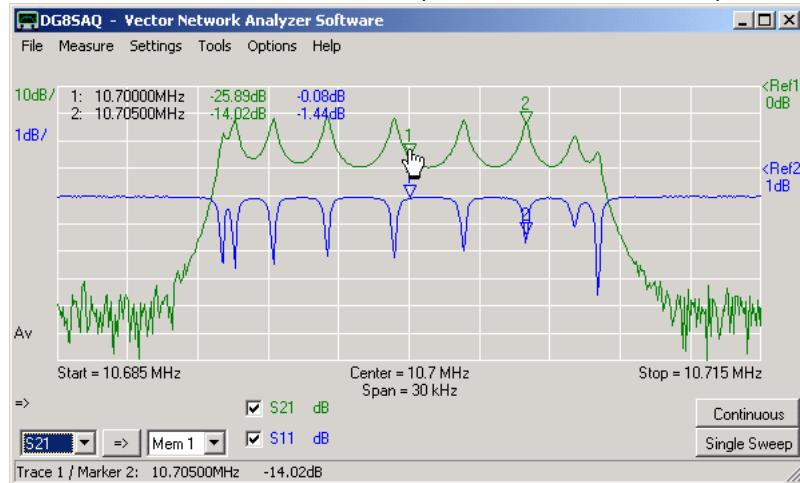
On the following screen Smith markers are evaluated as impedances:



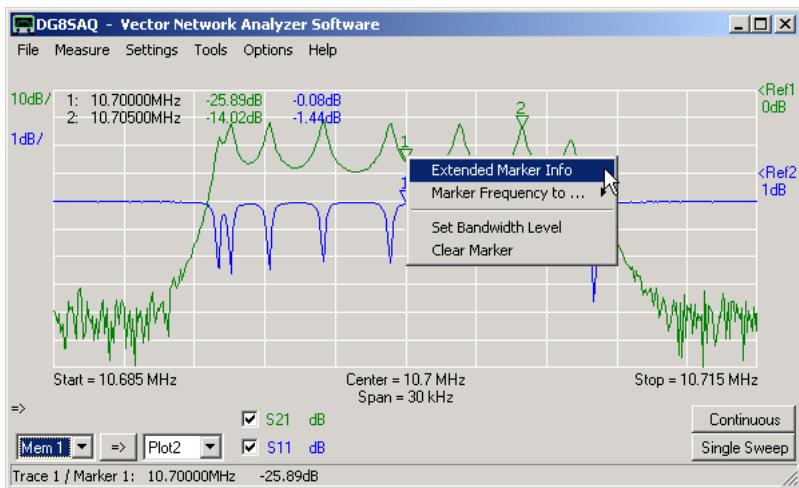
5. Extended marker information



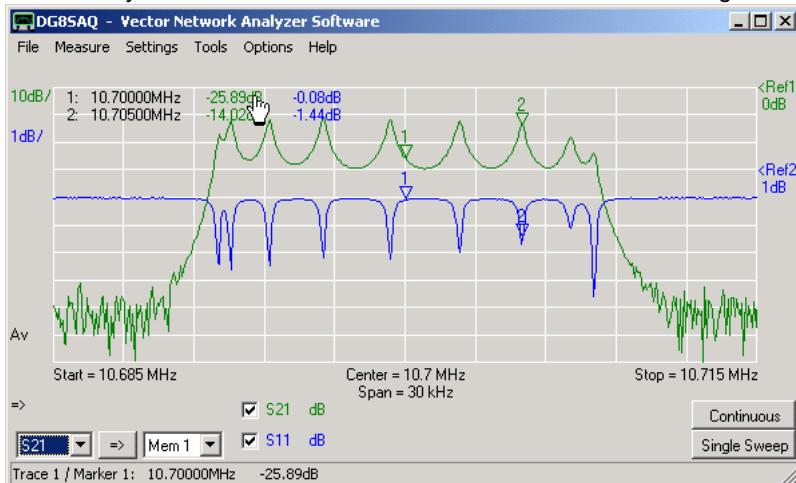
Extended marker information on a specific marker is available upon double-clicking that specific marker:



Since double-clicking the marker might slightly change the marker position, the same functionality is available without this risk by right-clicking the marker and selecting "Extended Marker Info":

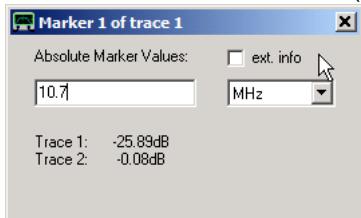


Another way to access this function is to double-click the value string belonging to the specific marker:



Note, that double-clicking the frequency part will have no effect as the frequency belongs to a whole row of markers on all traces.

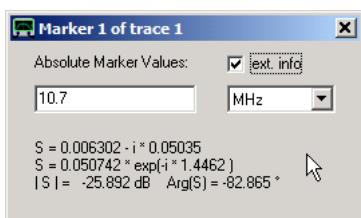
When marker 1 on the first trace (S21 trace) is selected then the following extended marker info window will show:



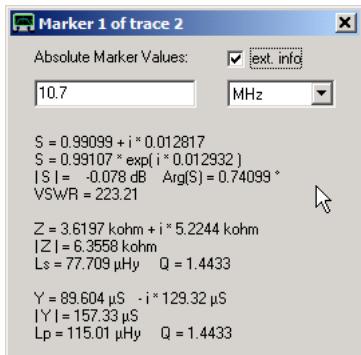
The marker frequency can be set to an exact value (not for maximum, minimum or bandwidth markers), e.g. 10.7 MHz in this example.

Note, that as long as the **ext. info checkbox** is unchecked, the values of all traces are being displayed at the same time.

Checking the **ext. info** checkbox (next to mouse pointer above) will yield extended marker info on one specific trace (the S21 trace 1 in this example):

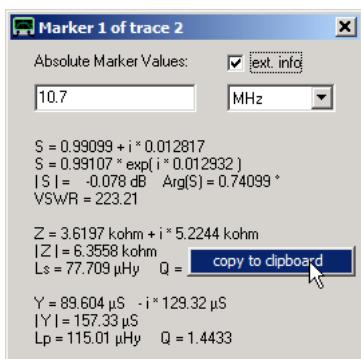


Even more extended information is available for reflection trace markers (S11, S22), for which impedances and VSWR aree being be calculated:



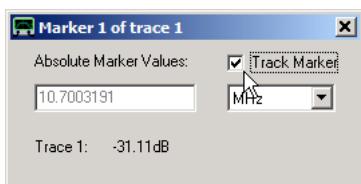
Note, that you can place the extended marker info window next to the main window and leave it open while sweeping. Whenever a trace is being updated or the corresponding marker is moved with the mouse, the extended marker info in the separate marker window is updated as well.

Note, that you can copy extended marker info to the Windows clipboard by right-clicking the extended marker window in order to paste it into any text document:



Tracking markers in spectrum analyzer mode

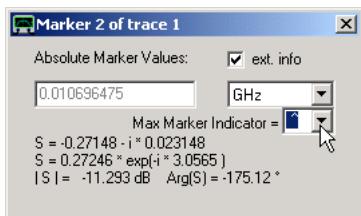
Note, that extended marker information doesn't make sense in spectrum analyzer mode, as no phase information is available there. Some times it is desirable in SA-mode to lock the instrument center frequency to a spectral line that might slowly drift. This can be achieved by setting a maximum marker that detects the spectral line. Double-clicking it will open the following marker window:



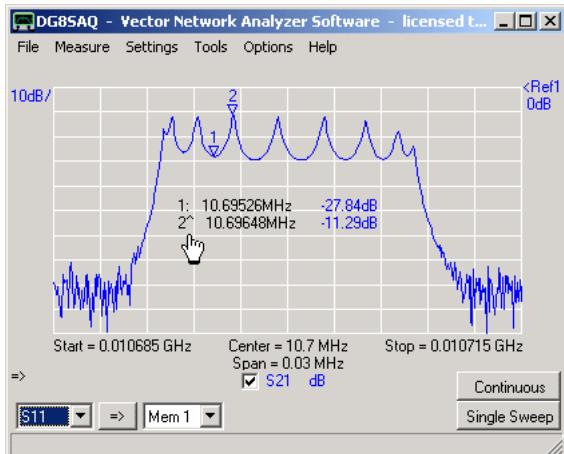
Note, that you cannot modify the marker frequency as this is determined by the maximum of the marker trace. Checking the "Track Marker" checkbox will cause the software to set the measurement center frequency to the marker frequency just before starting a new sweep. This way, it is attempted to keep the maximum in the center of the measurement span. This tracking also works for minimum and all bandwidth markers. Sometimes it is more stable to track the 10dB center frequency than the maximum of a peak.

Marking Minimum/Maximum Markers

It might be useful to make minimum/maximum markers visually distinguishable from normal markers as the former ones obviously cannnot be moved with the mouse. This can be achieved by double clicking a minimum or maximum marker (frequency markers only). The marker menu window then opens which looks like this for e.g. a maximum marker:



Note, that you can select a maximum marker indicator character (mouse pointer), which then replaces the colon after the frequency information in the marker text:



Note, that you can select two different indicator characters for minimum markers and for maximum markers.

6. Available Markers

Depending on the displayed data (frequency data, time data, radar data) the following marker types are available in the markers menu:

- Add Frequency Marker** (for frequency data and polar data, e.g. Smith charts)
- Add Time Marker** (for time domain data)
- Add RADAR Marker** (for angular data e.g. antenna diagrams)

►►► **Note:** Only frequency markers will yield an extended marker info window upon double-clicking.

Available Frequency marker types:

- **normal marker**
- **maximum marker** (automatically jumps to the maximum of the first applicable displayed trace)
- **minimum marker** (automatically jumps to the minimum of the first applicable displayed trace)
- **absolute level markers:**
 - **zero marker** (automatically jumps to the first zero crossing of the first applicable displayed trace)
 - **left marker** (automatically jumps to the first crossing at a specifiable absolute level from the left of the first applicable displayed trace, default level is minus one unit)
 - **right marker** (automatically jumps to the first crossing at a specifiable absolute level from the right of the first applicable displayed trace, default level is minus one unit)
- **bandwidth markers:**
 - **Bandpass** (places a maximum marker, two markers down one vertical division below the maximum marker and a center marker between the bandwidth markers)
 - **Bandstop** (places a minimum marker, two markers up one vertical division above the minimum marker and a center marker between the bandwidth markers)
 - **Max Left Down** (places a maximum marker, and one marker down one vertical division left of the maximum marker)
 - **Max Right Down** (places a maximum marker, and one marker down one vertical division right of the maximum marker)
 - **Min Left Up** (places a minimum marker, and one marker up one vertical division left of the minimum marker)
 - **Min Right Up** (places a minimum marker, and one marker up one vertical division right of the minimum marker)
- **delta marker** (displays the frequency distance and the vertical distance to the last normal marker)

►►► **Note:** If the frequency span is zero, the sweep time instead of the frequency is used to identify the marker position

in x-direction.

Available Time marker types:

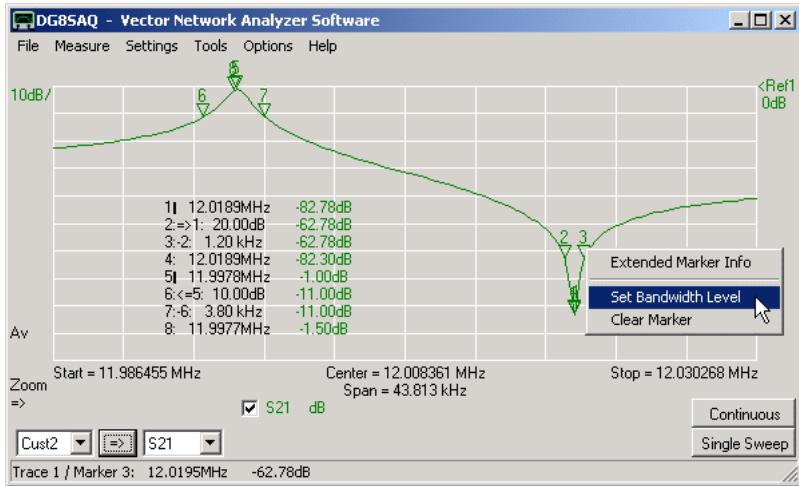
- **normal time marker**
- **maximum time marker** (automatically jumps to the maximum of the first applicable displayed time trace)
- **delta time marker** (displays the time distance and the vertical distance to the last normal time marker)

Available RADAR marker types:

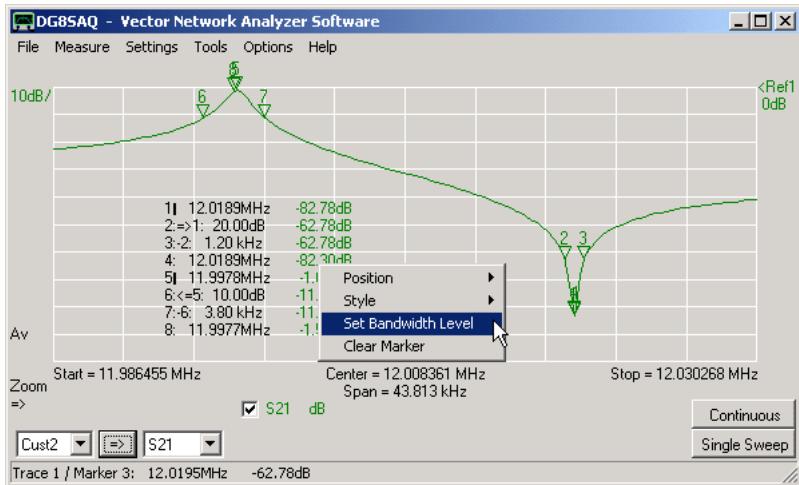
- **normal**
- **maximum** (automatically jumps to the maximum of the first displayed RADAR trace)
- **minimum** (automatically jumps to the minimum of the first displayed RADAR trace)
- **opening angle** (places a maximum RADAR marker, two RADAR markers down one vertical division below the maximum marker and a center marker between the opening angle markers)
- **delta marker** (displays the angular distance and the vertical distance to the last normal RADAR marker)

7. Bandwidth marker topics

The following screenshot shows an example of both a bandpass and a bandstop marker:



The bandwidth levels can be changed individually by either dragging the bandwidth markers (2,3,6,7) up or down with the mouse or by right-clicking one of the bandwidth markers (as seen above) or the associated marker captions (as seen below)...

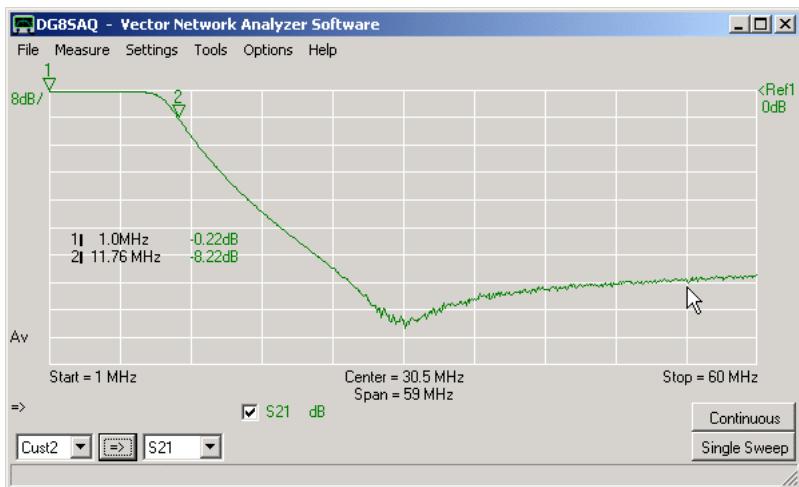


... and selecting "Set Bandwidth Level". A level input mask will pop up:



Enter a new value here and the corresponding bandwidth markers will move to the new relative level.

The same holds true for the other bandwidth markers like the "Max Right Down" marker shown in below example:



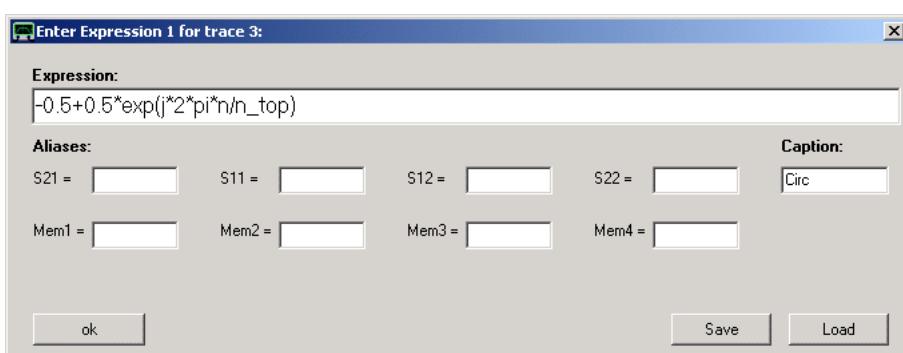
►►► **Note:** The bandwidth functions will be applied to the first frequency trace that is visible in the xy-grid. Make sure, that the trace you want to see the bandwidth of is the first one to show.

This also means, you cannot see a bandpass bandwidth on one trace and a bandstop width on another trace at the same time.

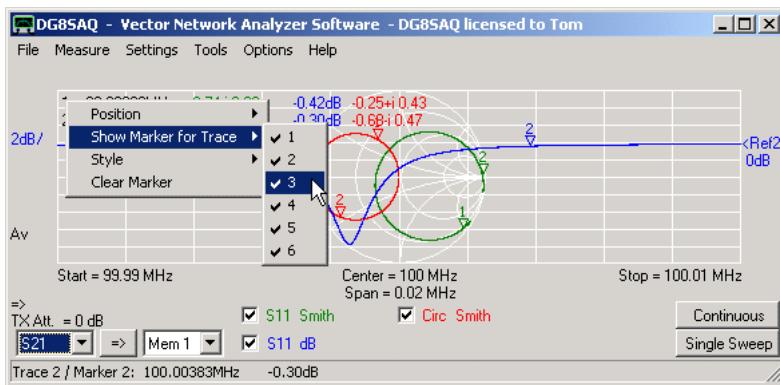
8. Switching markers OFF for specific traces

Sometimes, one wants to display traces that do not show markers. An example could be if a limit circle is displayed by using a custom trace.

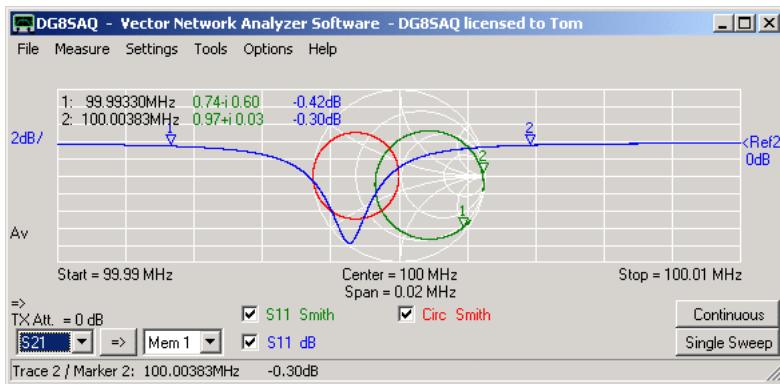
The following example custom expression displays a circle inside the Smith chart with center at -0.5 and radius 0.5:



As we do not want markers on this circle, we switch the markers for this trace off by right-clicking onto the marker caption and unchecking trace 3, which is the custom trace denoted with Circ.



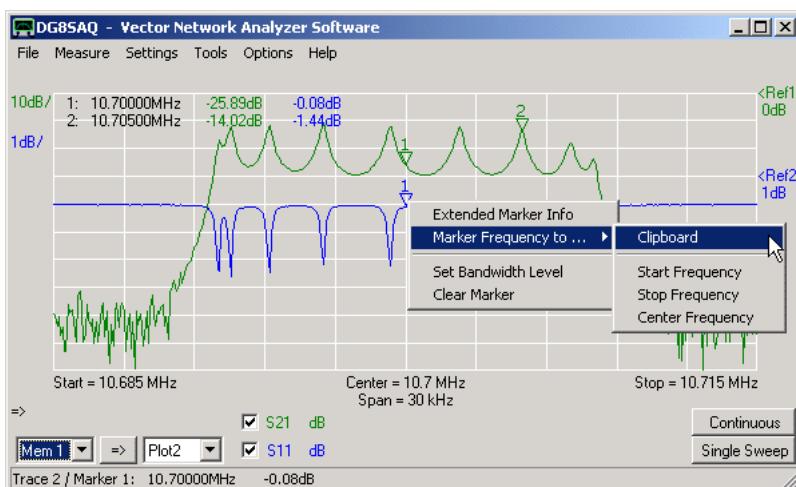
Note, that both markers and marker captions are disabled now for the red custom trace:



Note, that you can enable/disable the markers for specific traces also in the main menu Settings-Diagrams-Display-Traces.

9. Copying marker frequency information

Sometimes marker frequencies are needed for changing the VNWA frequency range or for putting it into a document. This can simply be achieved by right-clicking the marker symbol in question and selecting the menu "Marker Frequency to...":



Available destinations for copying:

- | | |
|-------------------------|--|
| Clipboard | The marker frequency is copied as string to the Windows clipboard so it can later be pasted into any document. |
| Start Frequency | The VNWA start frequency is changed to the marker frequency. |
| Stop Frequency | The VNWA stop frequency is changed to the marker frequency. |
| Center Frequency | The VNWA center frequency is changed to the marker frequency. |

Note, that when the VNWA frequency grid is changed, effects on the displayed traces will only be seen after remeasuring them.

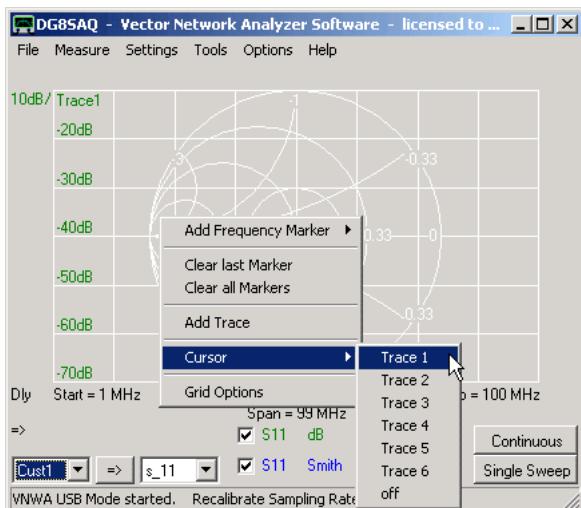
| **10. Changing marker symbol size and color**

The marker simbol size and color(s) can be changed via the "Grid Options-Misc. Options" settings. See page Settings - Diagrams - Display - Grid Options for examples.

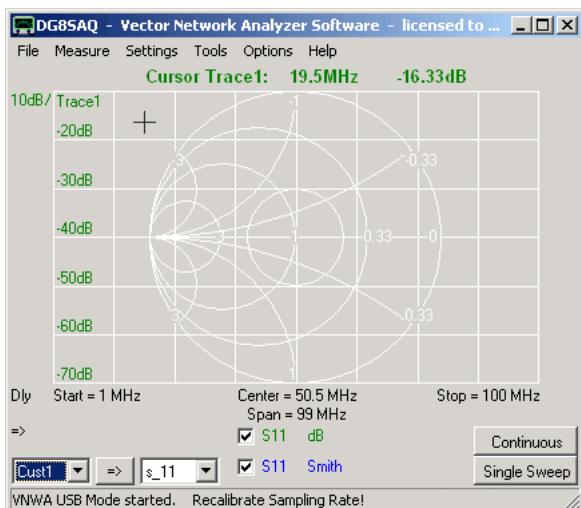
CURSOR

A **cursor** can be used to determine grid coordinates in a cartesian or Smith grid.

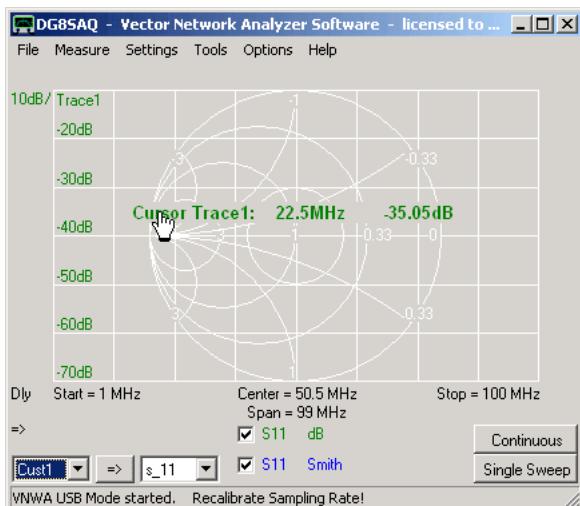
The cursor can be switched on by right-clicking onto the grid, selecting "Cursor" and selecting the trace for which the cursor is to be used for:



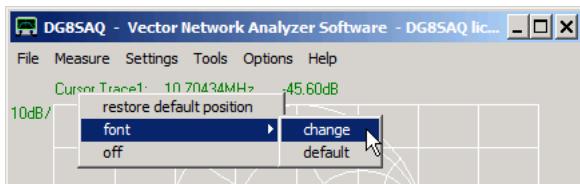
The mouse pointer will change to a cross and a **cursor label** colored with the trace color showing the cursor coordinates will appear right below the main menu:



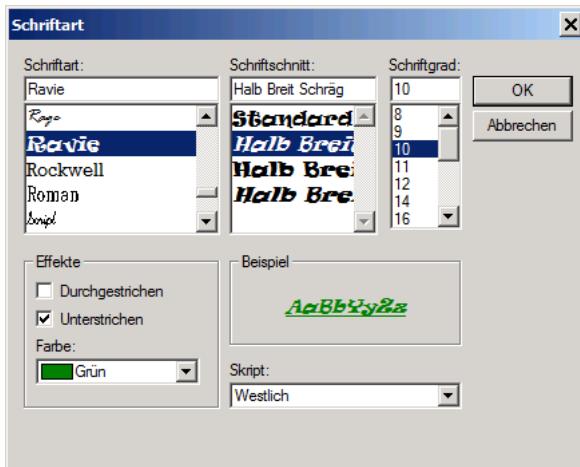
The **cursor label** can be moved with the mouse to any convenient position by holding the left mouse button pressed while pointing onto the label:



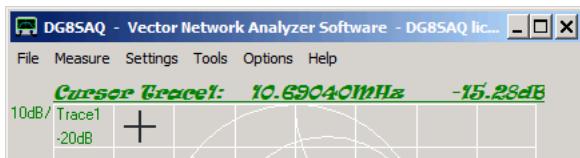
Right-clicking the label will open the label menu. There, the label may be moved back to the default top left position, the font, style and size may be changed or the cursor function may be switched off altogether:



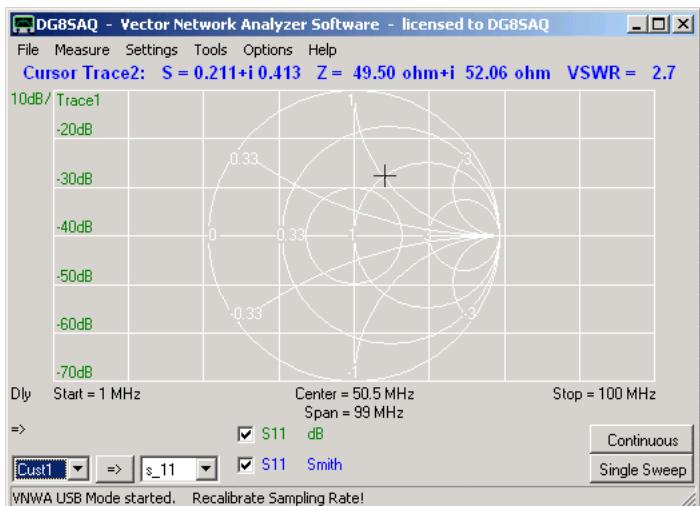
Upon selecting the **font - change** menu, the font manager opens:



There, any font style and size may be selected. After closing the manager, the cursor label will acquire the new font, style and size:



Extended information is available in a Smith grid:

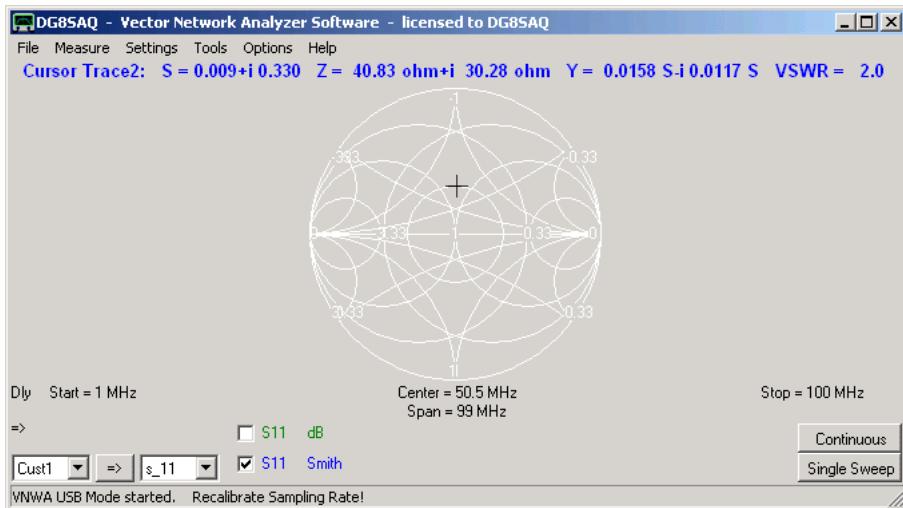


►►► **Note:**

Displayed coordinate data depends on the selected Smith grid options:

- **Impedance** information is only shown if the impedance grid is activated.
- **Admittance** information is only shown if the admittance grid is activated.
- **VSWR** information is only shown if VSWR circles are activated.

Here, all Smith grid options are activated:



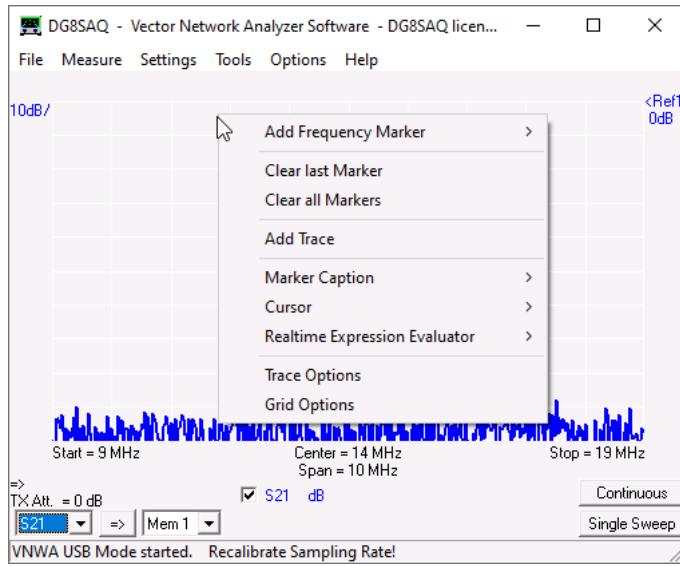
RIGHT-CLICK MENUS

Almost any text element on the VNWA main screen has an individual right-click menu that allows to access related functions quicker than via the main menus.

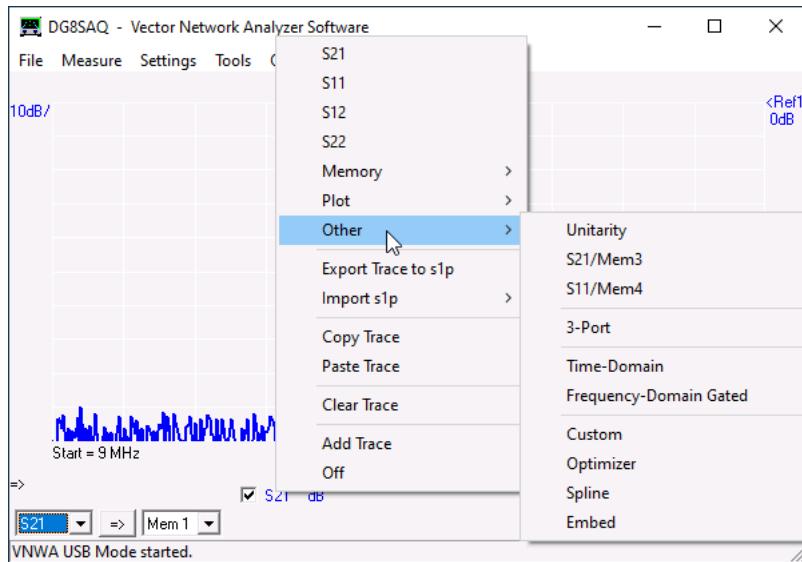
Note, that almost any text element also has a double-click functionality opening a corresponding window. Moreover, some settings can be manipulated by pointing the mouse to the corresponding label and turning the mouse wheel.

The following lists the most important right-click menus.

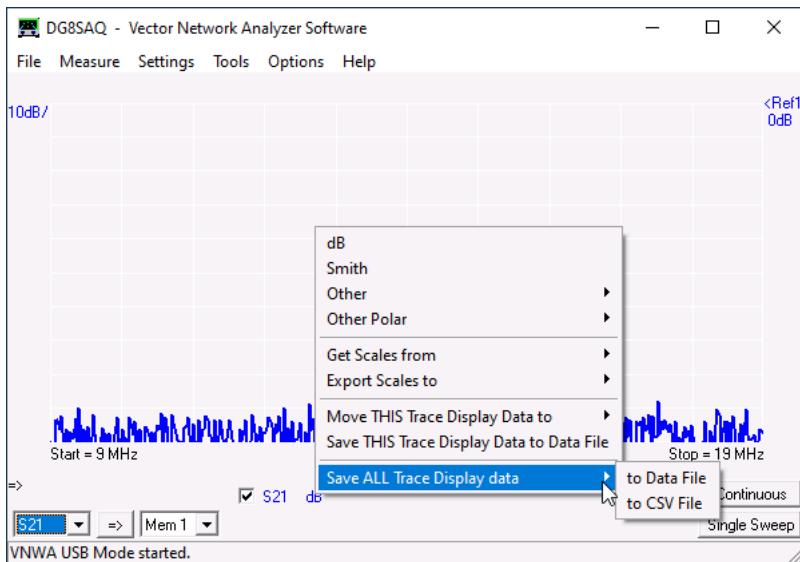
Right-clicking any position of the grid surface yields marker, trace and grid related menus:



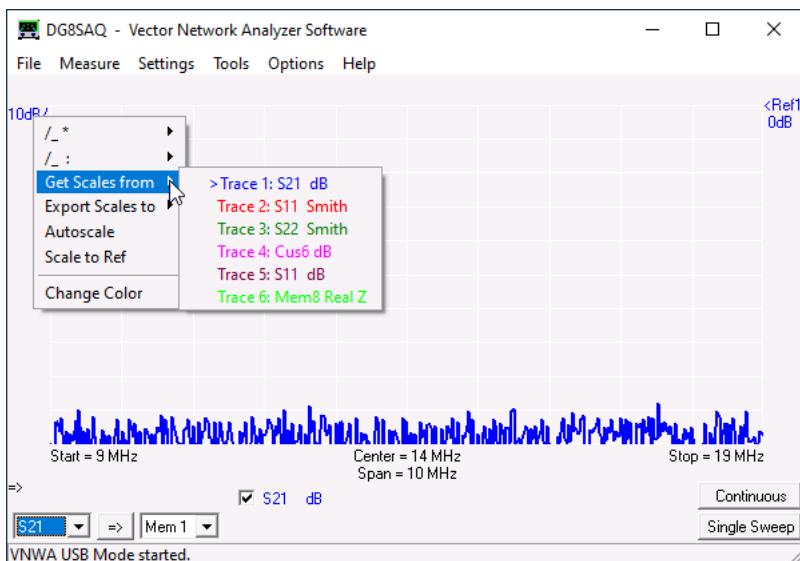
Right-clicking a trace data field label on the bottom (here S21) allows to select a different data field for the corresponding trace, import or export data, manipulate traces:



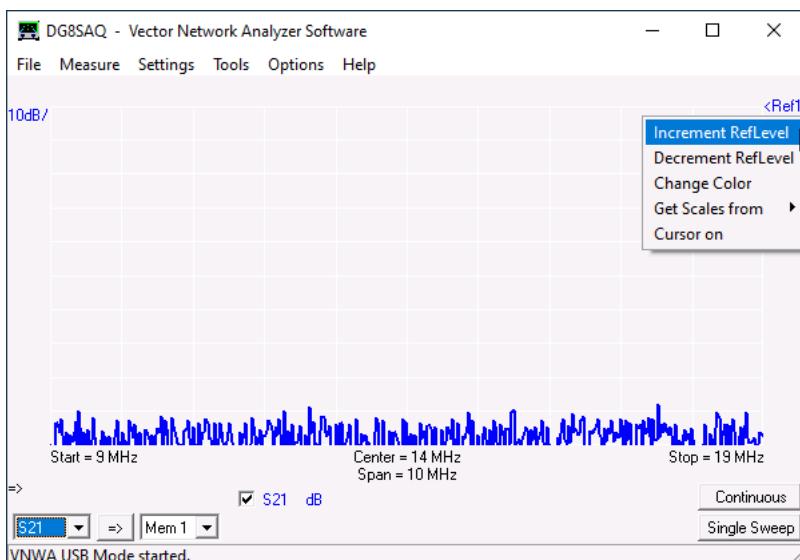
Right-clicking a trace unit field on the bottom (here dB) allows to change display mode of the corresponding trace, move and save display data:



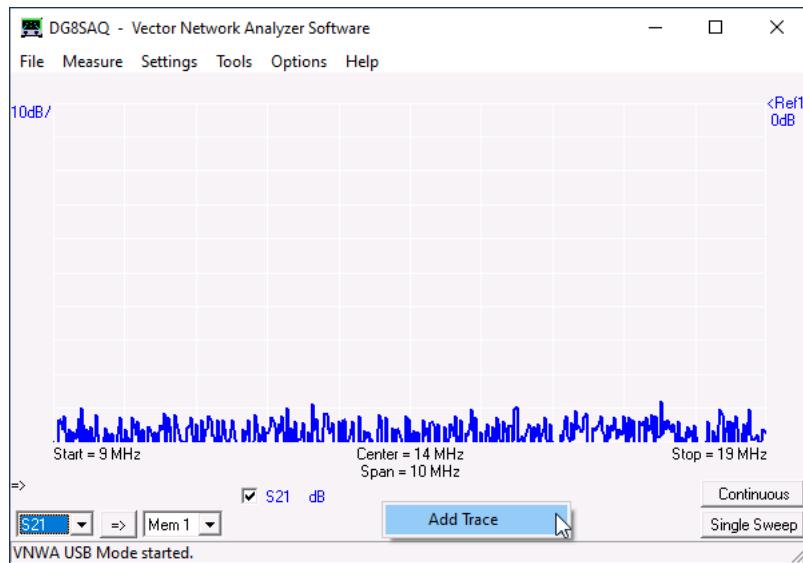
Right-clicking a */ label on the left allows to change scales and the color of the corresponding trace:



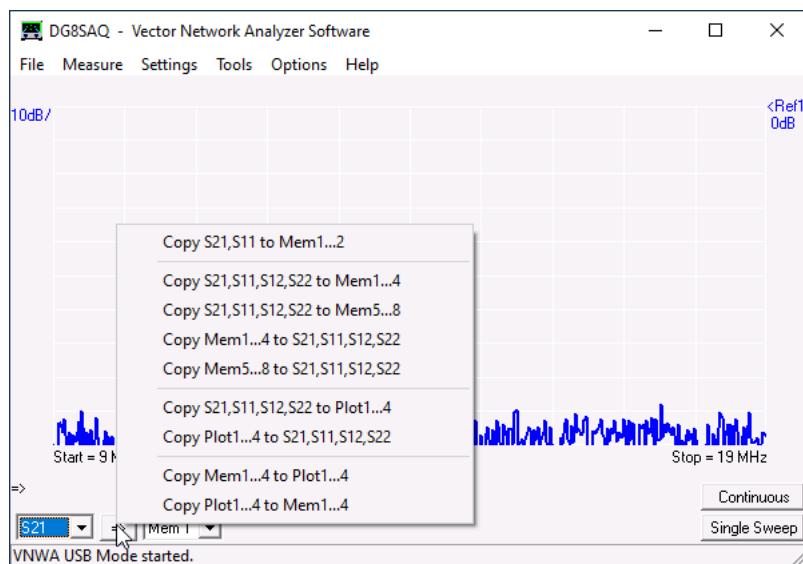
Similar controls are available through right-clicking the reference label on the right:



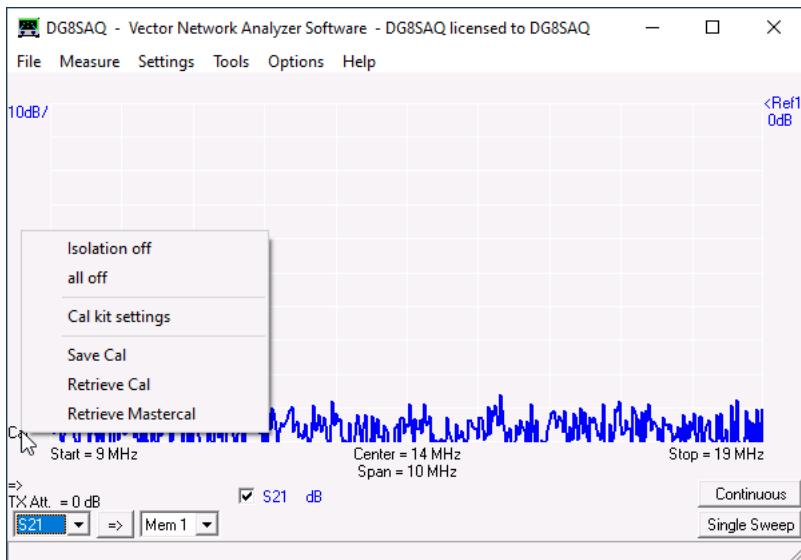
Traces can be added by right-clicking the free space below the grid:



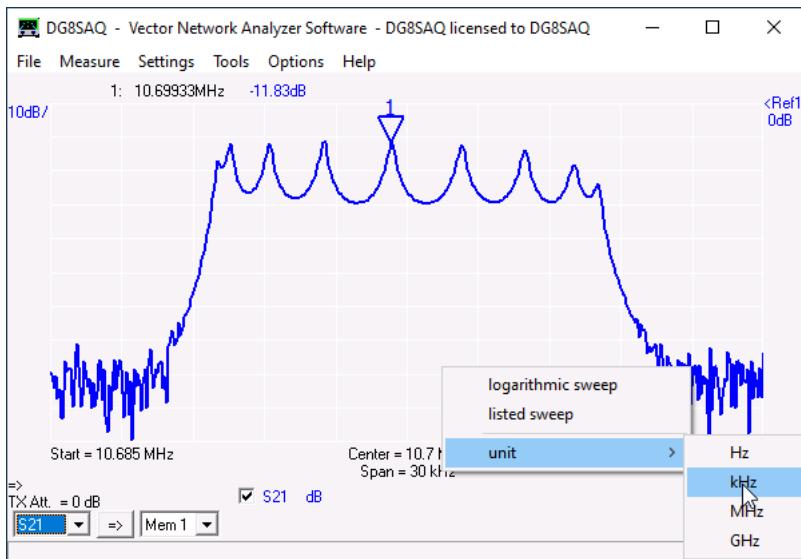
Right-clicking the copy-button on the lower left allows to copy multiple data fields at once:



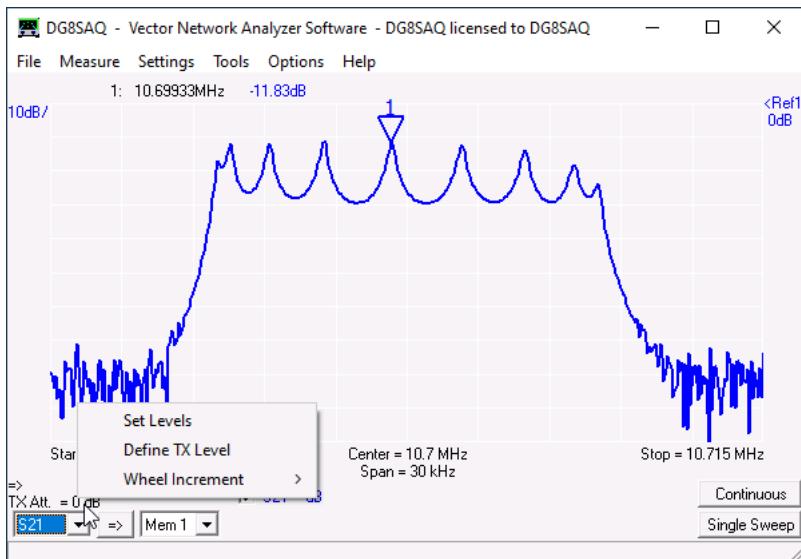
Right-clicking the Cal-label on the left allows to manipulate the calibration:



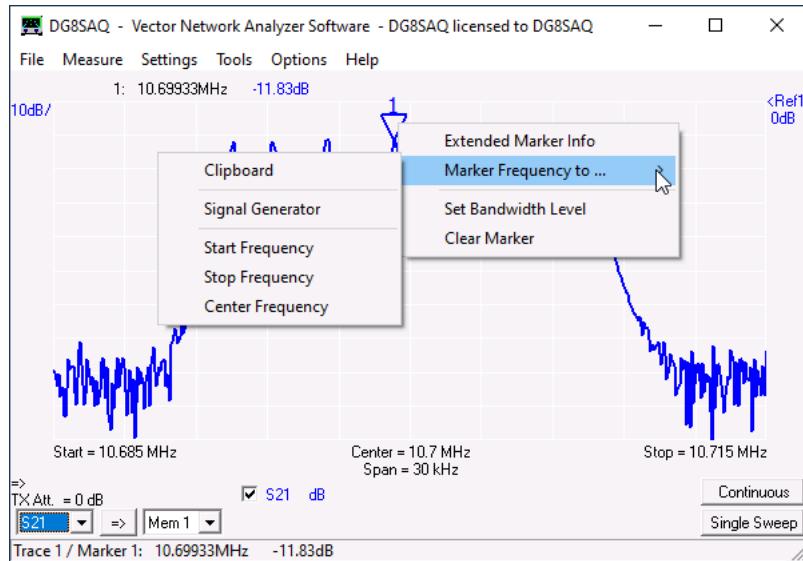
Frequency axis type and frequency units can be changed by right-clicking any of the frequency labels on the bottom:



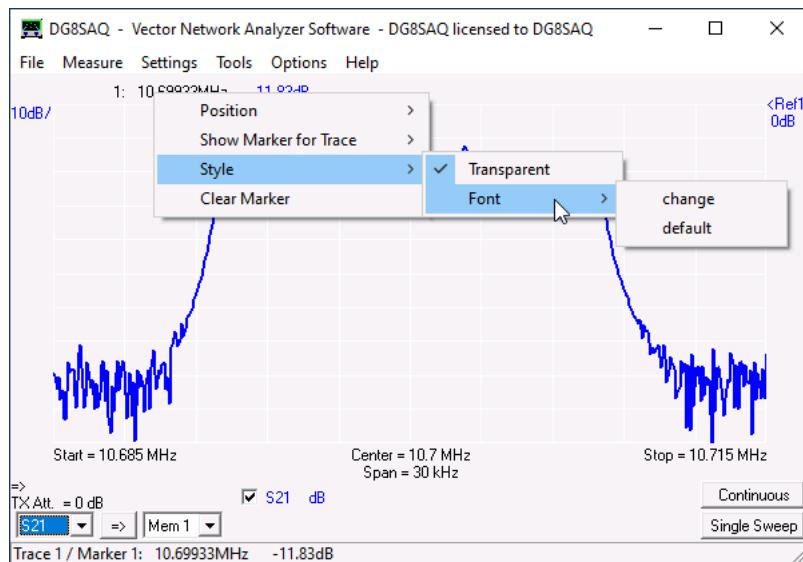
The TX power level can be manipulated by right-clicking the attenuation/level-label:



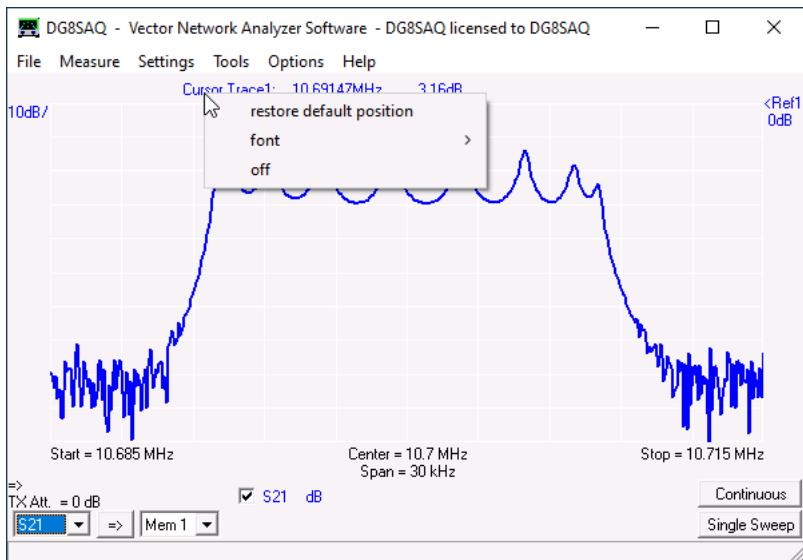
Right-clicking a marker symbol allows to access and modify the corresponding marker properties:



Right-clicking a marker caption lets you modify the marker text appearance:



If the cursor functionality is activated, right-clicking the cursor text allows to configure its appearance:



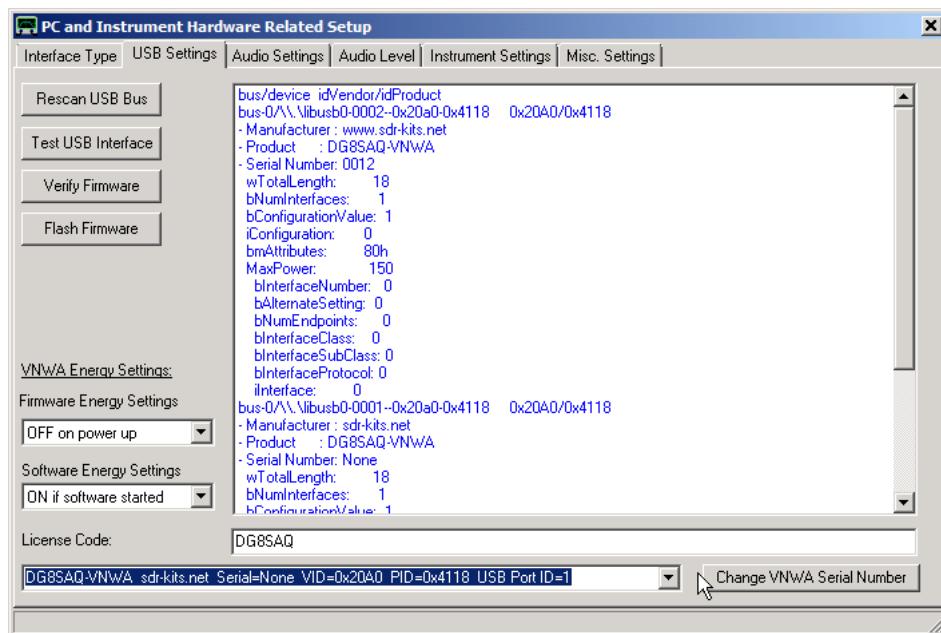
MULTIPLE VNWA SUPPORT

Using more than one VNWA may lead to two questions, which are being answered below.

1. How do I select one of several VNWAs connected to the PC at the same time inside the VNWA software?
2. How can I use multiple VNWAs with different license codes without having to retype the license code every time I change the VNWA?

1. Selecting one of several connected VNWAs

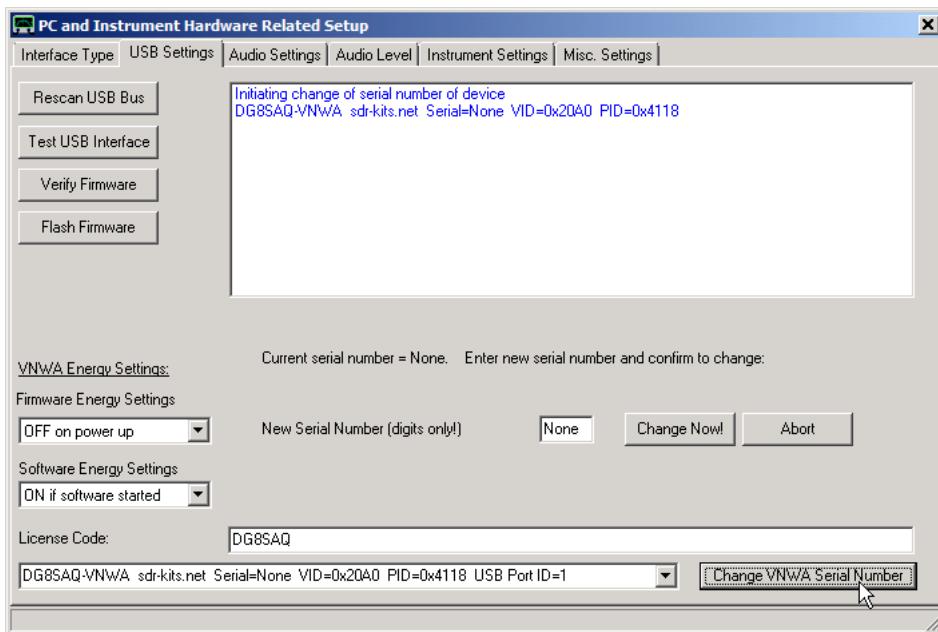
If you have more than one VNWA, you can have all VNWAs connected to your PC simultaneously. If **more than one VNWA is detected** upon program start or upon "**Rescan USB Bus**", the "Setup"-"**USB Settings**" menu will show a combo box which allows the user to select one of the connected VNWAs (very bottom, next to the mouse pointer):



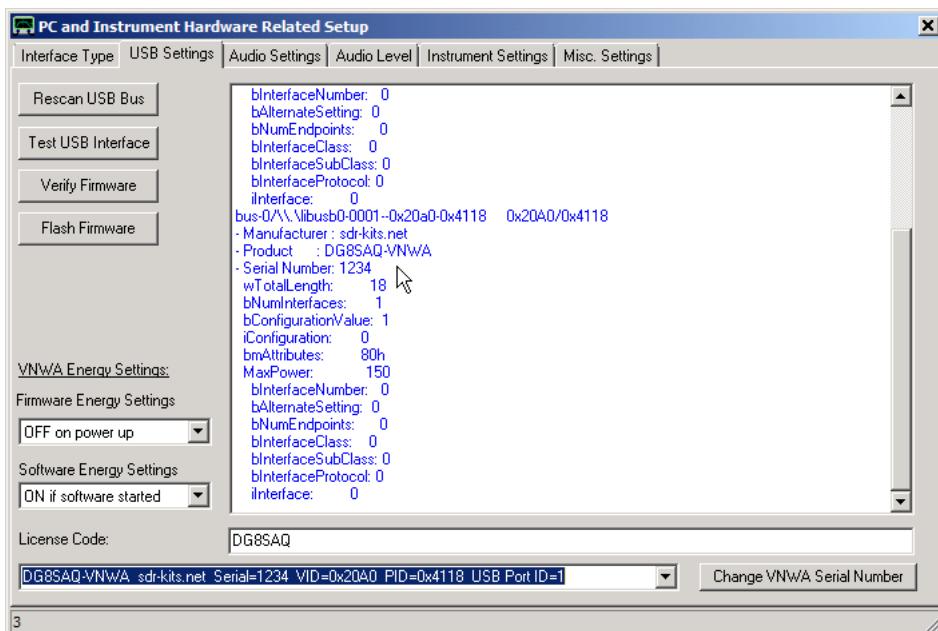
Note, that identical VNWAs can be distinguished only by their USB port ID, which is basically the order of device detection. This order might change upon reboot or replugging.

Therefore it is useful to modify the VNWAs' serial numbers, which are factory preset to "NONE". If more than one VNWA is connected, it is recommended to use consecutive serial numbers like 0001, 0002,

In order to **change a VNWA's serial number**, select the VNWA which you want to change and press the according button:



Enter a new 4 -digit serial number (e.g. 1234 in below example, you are only allowed to enter digits, no letters!) and press the "**Change Now!**" button:



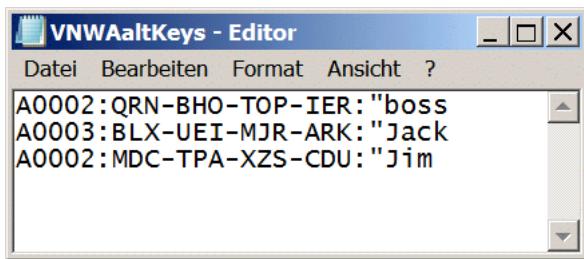
Note, that the VNWA serial has been changed to 1234 now (see entry next to mouse pointer above and also in the selection box).

From now on, this VNWA is uniquely determined and selected by its unique serial number 1234.

2. Activating more than one license code

When several VNWAs with different license codes are to be operated from the same software installation, it is quite inconvenient to manually modify the license code every time the VNWA is replaced by another. Therefore the following means has been created to let the VNWA software handle up to 30 different license codes at the same time:

- Create a text file with name **VNWAaltKeys.ini** inside your VNWA installation folder and enter all your license keys, one per line:



Note, that the part starting with " does not belong to the license code but is interpreted as the licensee's name, which is being shown in the program title bar.

- b) Save and restart the VNWA software. From now on every VNWA with a license code found in the list can be operated.

KEYBOARD SHORTCUTS

The VNWA (starting VNWA V33.z) software can be **controlled with the keyboard** to some extent.

For this purpose, some **keyboard shortcuts** are available to initiate frequently used tasks.

Note, that for VNWA36.3.1.9 and later the shortcuts marked red, are deactivated while VNWA is **sweeping** for stability reasons.

key:	action:
alt+1...9	load default instrument state 1...9 including hardware settings
alt+f	set linear sweep range
alt+g	set logarithmic sweep range
alt+l	set listed sweep range
alt+a	autosave memory spaces on/off
ctrl+alt+a	autoretrieve memory spaces on/off
alt+c	save calibration
ctrl+alt+c	retrieve calibration
alt+d	save display state
ctrl+alt+d	retrieve display state
alt+m	save calibration to mastercal
ctrl+alt+m	retrieve master calibration
ctrl+alt+r	open RC debug window
ctrl+alt+s	TCPIP data server on/off
ctrl+alt+t:	TCPIP RC servers on/off
alt+p	restore default marker caption position
alt+t	open specification tester tool
ctrl+2	import s2p-data from file
shift+ctrl+2	export s2p-data to file (dB/phase)
ctrl+c	save screen to clipboard
ctrl+i	retrieve instrument state ignore hardware settings
ctrl+j	retrieve instrument state
shift+ctrl+i	save instrument state
ctrl+a	Spectrum analyzer mode
ctrl+d	create a debug file
ctrl+e	External Bridge mode
ctrl+f	save screen to file
ctrl+g	Signal generator mode
ctrl+m	Frequency meter mode
ctrl+p	print screen
ctrl+r	RF-IV mode
ctrl+s	activate/deactivate storage screen
ctrl+t	define TX level
ctrl+v	VNWA mode
ctrl+x	clear storage screen
ctrl+alt+w	set all grid colors to white
ctrl+alt+x	activate/deactivate sound mixer (Windows XP and older only)
shift+ctrl+x	activate/deactivate SelectlsMute (Windows XP and older only)
shift+ctrl+c	report flash checksum (returns 0xFFFFFFFF if not supported)
F1	open help file
F2	measure 2 port S-parameters
F3	measure 3-port S-parameters
F5	save screen to file
F6	print screen
F10	open s1p drag-drop target
F11	save S11 into S1P-file
F12	save current S-parameters into S2P-file
spacebar	start/stop single sweep
return	start/stop continuous sweep
right arrow	measurement direction forward
left arrow	measurement direction reverse
+	add a trace
-	remove last trace
0	cursor off
1	cursor trace 1 on
2	cursor trace 2 on
3	cursor trace 3 on

4 cursor trace 4 on
5 cursor trace 5 on
6 cursor trace 6 on
a open **average** and autosave
b open **about** screen
c open **calibrate** menu
d open **display-trace** menu
g open **grid** options menu
h open **help** file
k open **cal kit** settings menu
l open **power levels** settings menu
m open **matching** tool
n open **complex calculator** tool (**numerics**)
p open **port extensions** menu
r open **realtime expression evaluator**
s open **sweep** settings menu
t open **three-port analyzer** tool
x open **crystal** analyzer tool

HIGHLIGHTING FREQUENCY BANDS

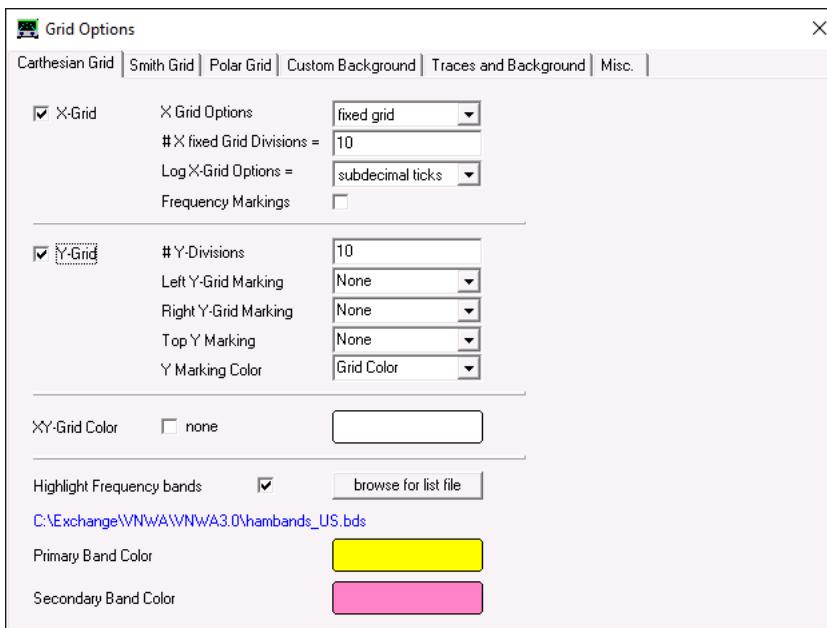
The current VNWA software allows to highlight frequency bands pre-defined in a frequency list file with extension *.bds. The current installation package contains two example files *hambands_DL.bds* and *hambands_US.bds* listing the amateur radio frequency bands for Germany and the USA.

Here is the content of one of the files:

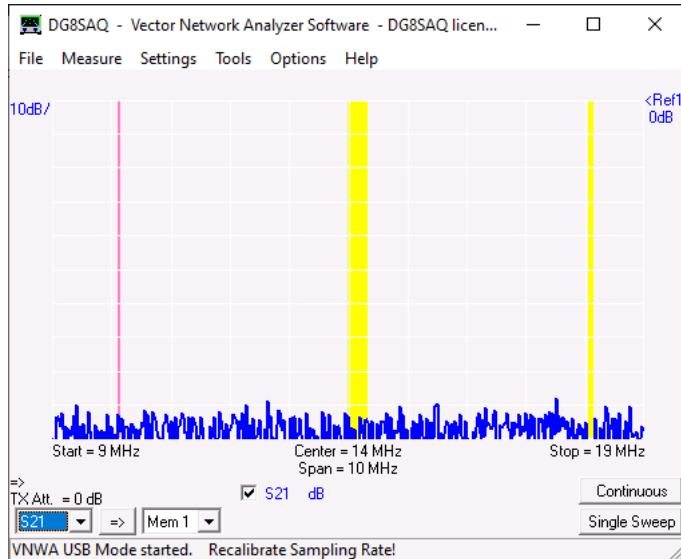
Datei	Bearbeiten	Optionen	Codierung	Hilfe	85 %
135.7	137.8	KHz	S		
1800	2000	KHz	P		
3500	4000	KHz	P		
5330	5404	KHz	S		
7000	7300	KHz	P		
10100	10150	KHz	S		
14000	14350	KHz	P		
18068	18168	KHz	P		
21000	21450	KHz	P		
24890	24990	KHz	P		
28	29.7	MHz	P		
50.	54	MHz	p		
144	148	MHz	P		
420	450	MHz	P		
1240	1300	MHz	S		
2320	2450	MHz	S		
3400	3475	MHz	S		
5650	5850	MHz	S		
10	10.5	GHz	S		
24	24.05	GHz	P		
24.05	24.25	GHz	S		
47	47.2	GHz	P		
75.5	76	GHz	P		
76	77.5	GHz	S		
77.5	78	GHz	S		
78	81.5	GHz	S		
122.25	123	GHz	S		
134	136	GHz	P		
136	141	GHz	S		
241	248	GHz	S		
248	250	GHz	P		

Each line denotes one frequency band with start frequency, stop frequency, frequency unit and priority. Priority P denotes a primary band while S denotes a secondary band. Primary and secondary bands can be displayed in different colors.

A frequency list file can be loaded via Grid-Options:



If the checkbox "Highlight Frequency bands" is checked, the bands will be shaded in the selected color:



Above, two primary bands and a secondary band are visible.

Instrument Accuracy Considerations

Note, that the VNWA does not contain user adjustable components or adjustable firmware settings. So, it does not require regular factory recalibration in order to maintain its level of accuracy.

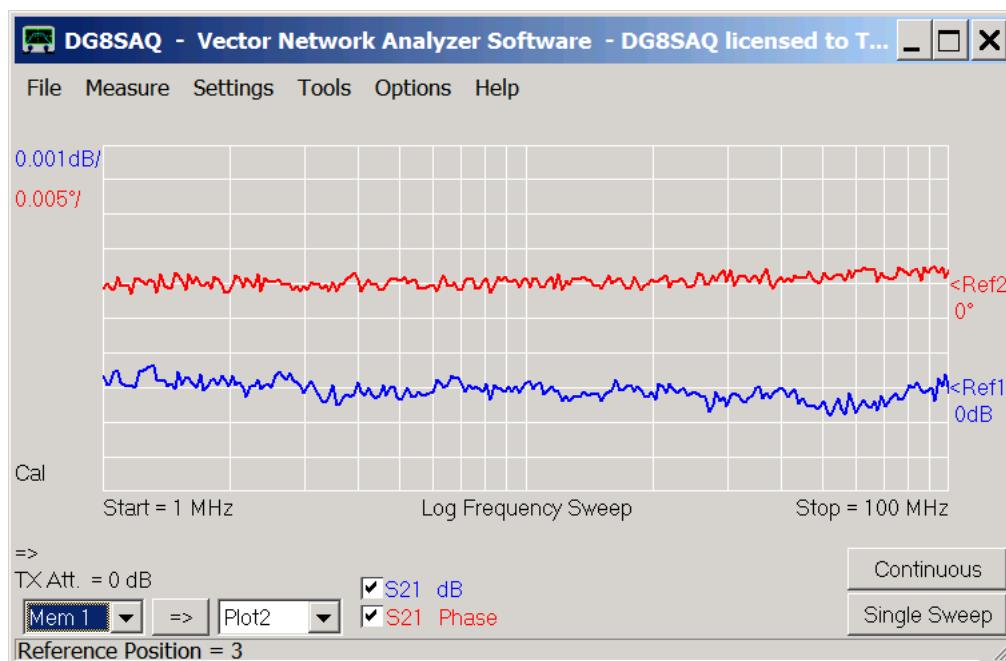
If you are in doubt of proper operation of your VNWA, you may repeat the final commissioning test and compare to the measurement sheets you have received together with your instrument upon purchase.

The test is described on page "VNWA Final Commissioning Test".

The VNWA allows for many different user settings which all influence measurement uncertainty. Moreover, the user must calibrate the VNWA prior to usage.

The quality of the calibration standards and the repeatability of the connector matings have great influence on the measurement uncertainty. Even bending the test cables will introduce noticeable changes in the measurement results. Hence, the measurement uncertainty cannot be simply specified by a number.

Yet, if settings are optimal, the VNWA allows to achieve amplitude stability of better than +/-0.001dB and phase stability of better than +/-0.005°, which is very comparable to what professional equipment can do.

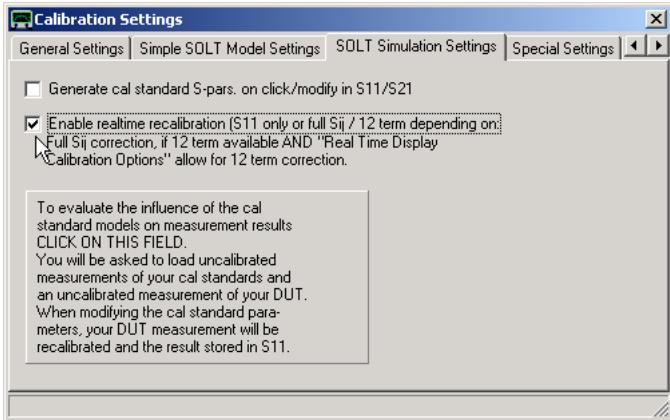


In the following all possible influences on measurement uncertainty will be discussed and demonstrated.

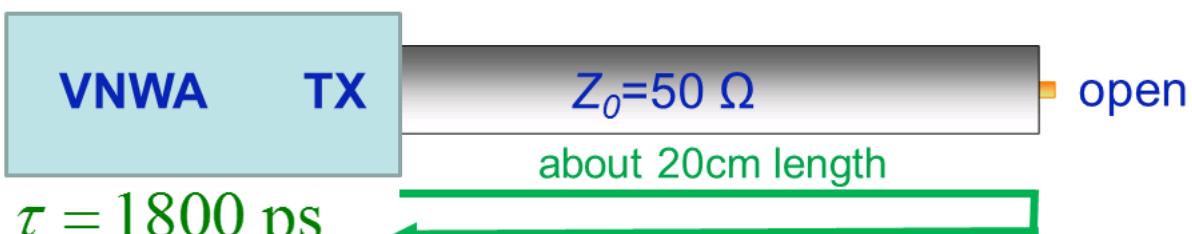
Calibration Error (Port Mismatch) and Linearity

The VNWA software can in principle **eliminate all linear hardware error terms** by a proper calibration. It is crucial though, that the software knows the properties of the calibration standards depending on the frequency. Thus, the calibration standards must be accurately characterized, modeled and the models have to be properly entered into the VNWA software. The VNWA software allows for any mathematical model you can think of, see chapter arbitrary calibration standard model on page "Calibration Standard Setup". This feature is quite unique in the field of hobby instruments. Note, that the calibration standard properties can never be known exactly. This will introduce a principal error that can be minimized but cannot be eliminated.

If the actual calibration standard properties differ from the model entered into the VNWA software, the resulting measurement error may be considerable. This error is known as **port mismatch**. The name is misleading as it has nothing to do with the VNWA TX or RX port impedances. The VNWA software can simulate port mismatch errors. We will use this simulation capability to exemplify the effect of an inappropriate calibration kit model on a reflection measurement result (S11) of a simulated 20cm piece of open ended 50 Ohm coaxial transmission line. We use the best known calibration kit parameters to perform a calibration. We do an instrument calibration in hardware, so the instrument knows the uncorrected calibration standard measurements. Next, we allow the software to re-correct raw measurement data on the fly, when the calibration kit parameters are being changed by activating the corresponding checkbox in Settings - Calibration Kit:



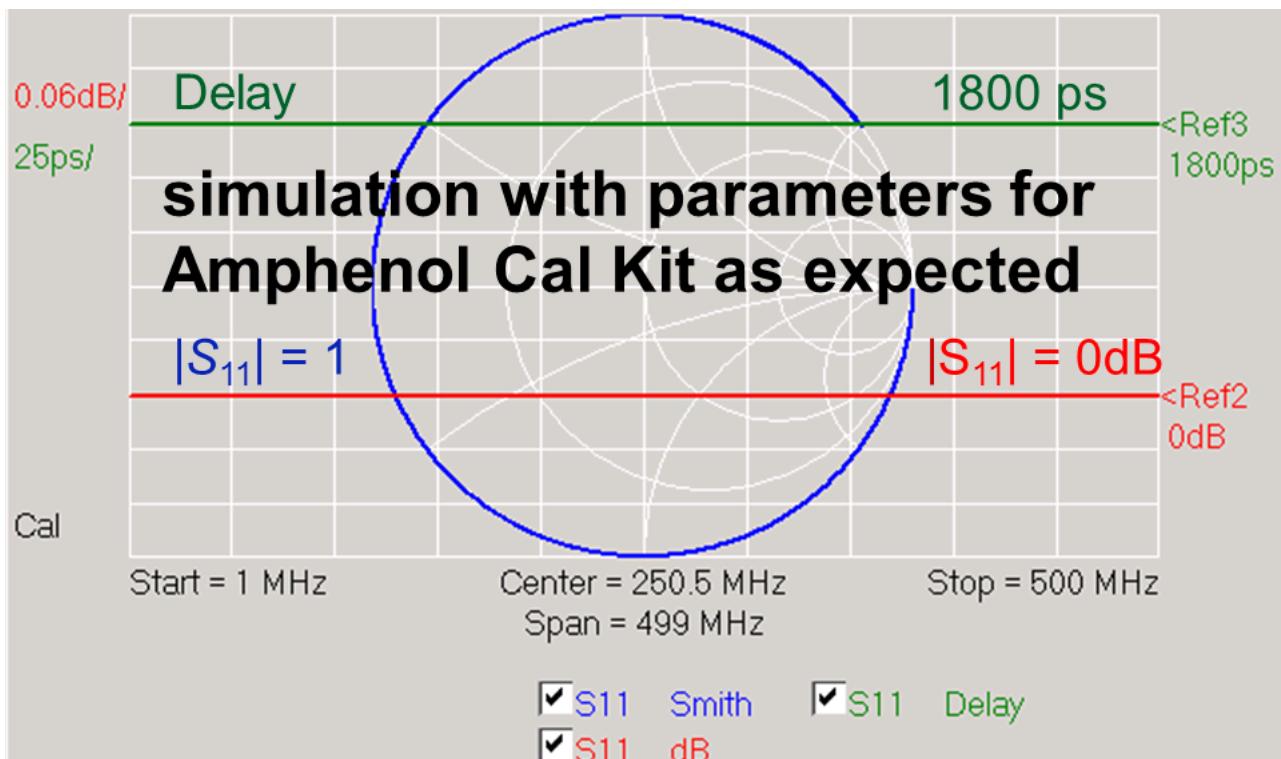
Instead of performing a reflection measurement on an actual transmission line, we simulate a measurement of an ideal transmission line.



$$|S_{11}| = 1 \quad \text{total power reflected}$$

$$\text{phase}(S_{11}) = -\omega \cdot 1800 \text{ ps}$$

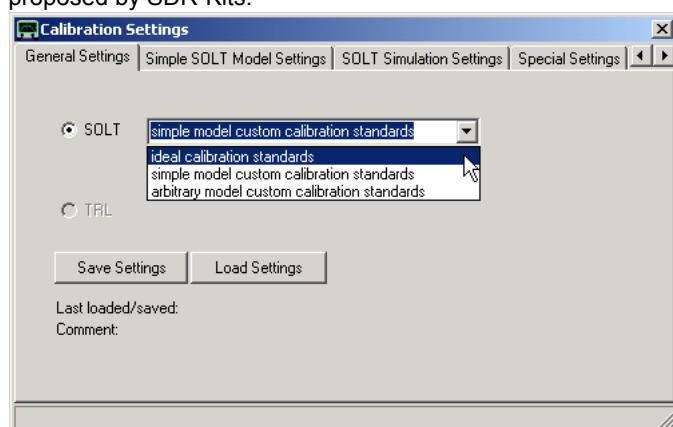
The VNWA software uses the calibration kit parameters to calculate raw uncorrected pseudo-measurement data of this simulation and recorrects it:



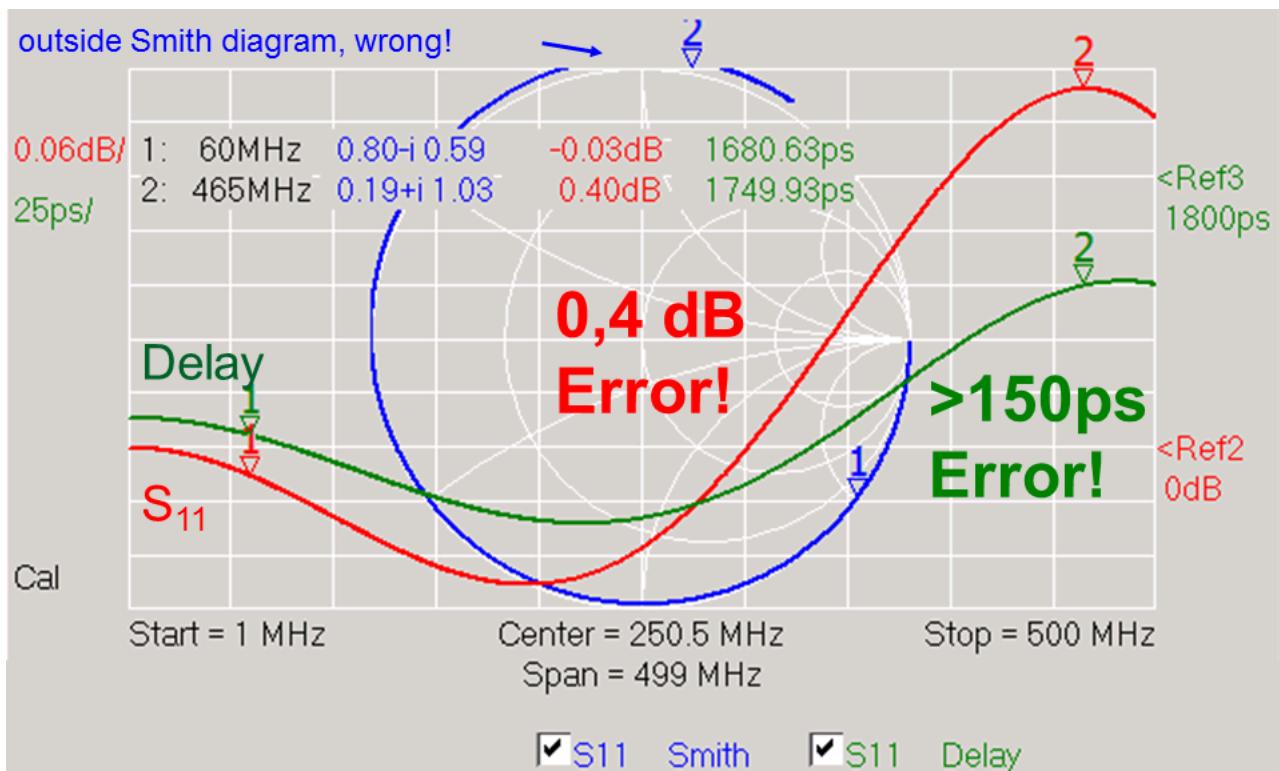
Indeed, we observe the simulated 100% reflection (=0dB) and the simulated 1800ps delay.

Now, we can change the calibration kit parameters and observe the effect on the simulated measurement result in real-time.

In the following example, we use the ideal calibration kit setting instead of the model for the Amphenol calibration kit proposed by SDR-Kits:



The simulation yields rather big deviations from the correct result, which are way beyond what the instrument can achieve if properly used:



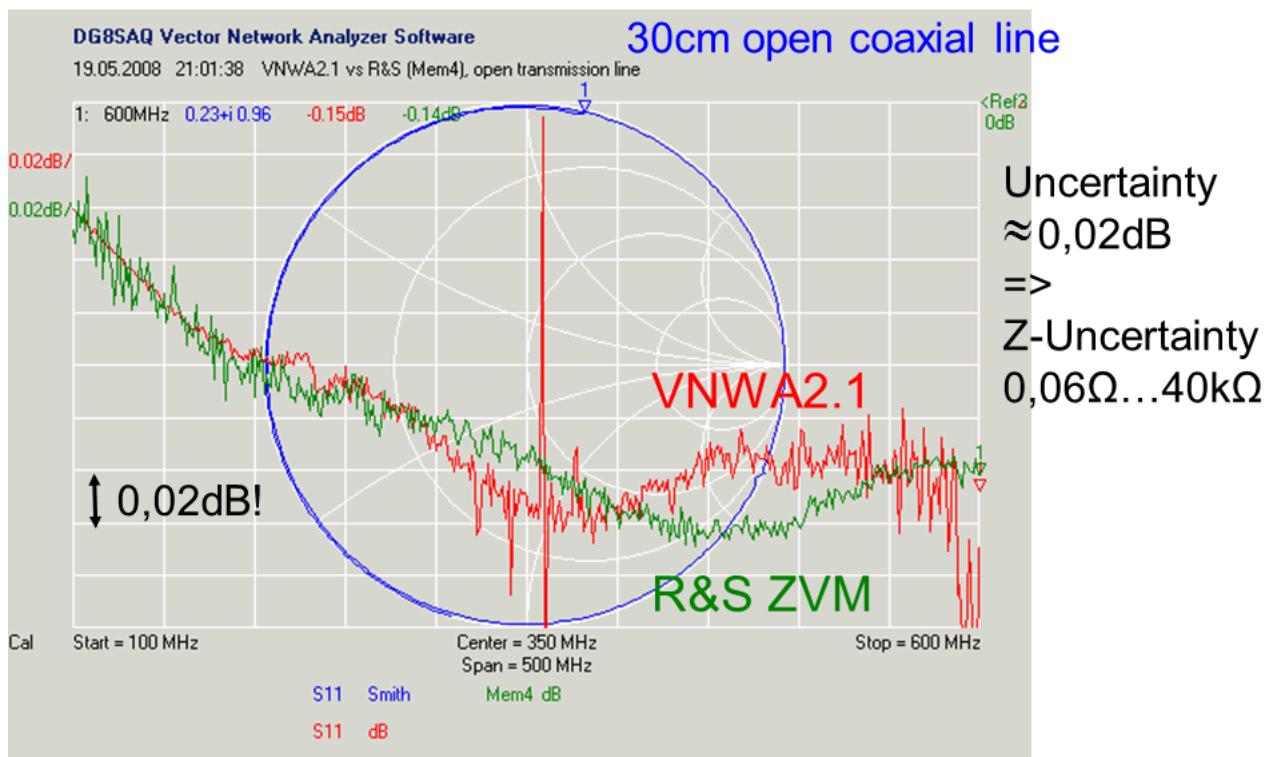
Note, that this is what you would measure when using an Amphenol cal kit and ideal cal kit settings instead of the proper Amphenol ones.

Also note, that these kinds of errors already show at relatively low frequencies!

So, remember to always **use the correct cal kit model** with the correct model parameters matching your cal kit!

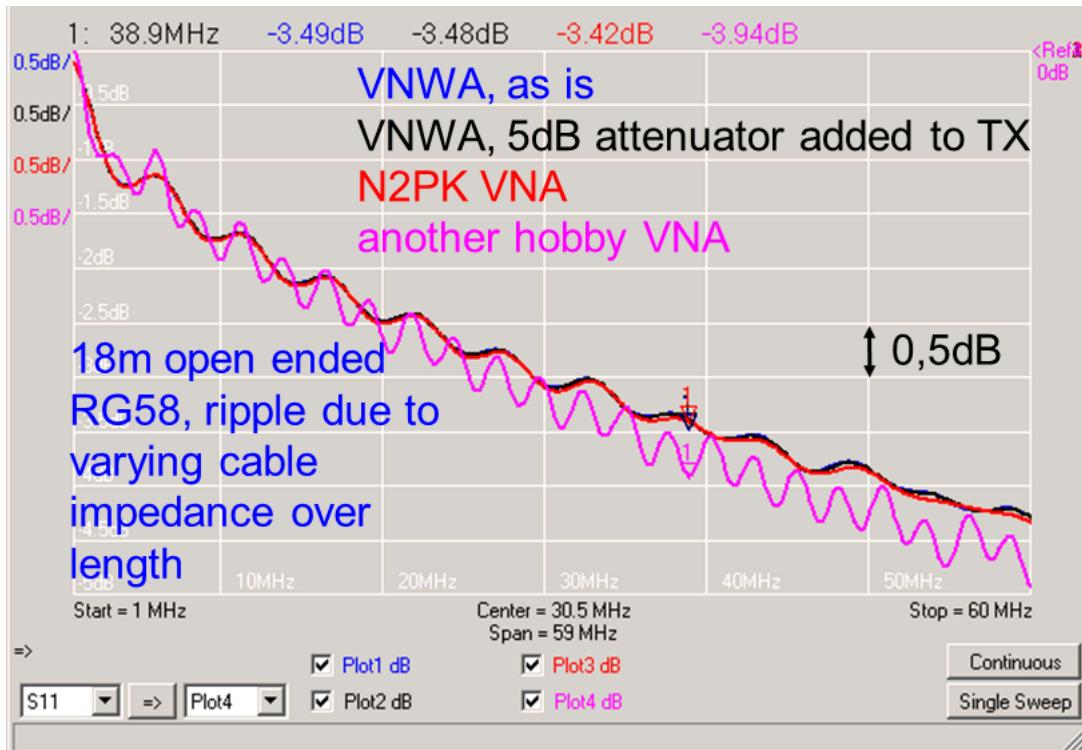
Now, that we have taken care of the linear errors, what is the influence of non-linear errors?

The best I can do to answer this question is to measure a worst case test object (real open coaxial transmission line) with a commercial reference instrument under identical conditions (same calibration kit, same calibration kit model) and compare:



Being a hobbyist instrument, the VNWA compares quite well to the high grade commercial VNA.

Here is a comparison of 3 hobby VNAs measuring the reflection coefficient of an open ended RG58 transmission line with 18m length in the frequency range that all instruments share:



While the VNWA and the N2PK VNA measure identical ripples, the third party VNA gets the ripple completely wrong due to nonlinear errors, even though it is completely compliant with its own specification.

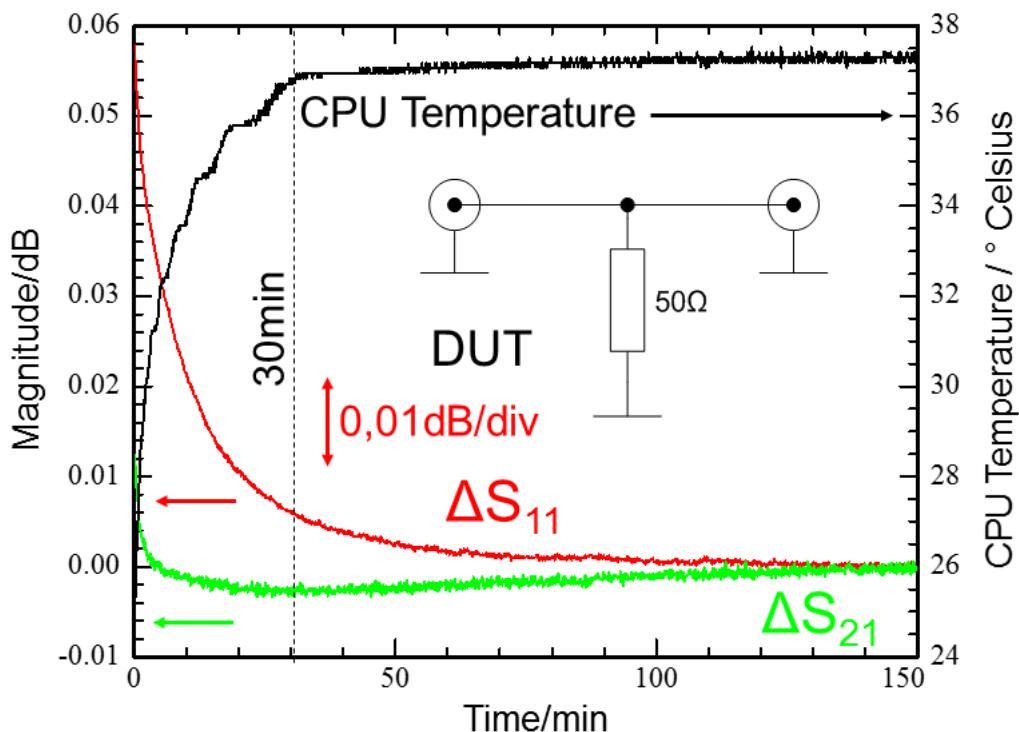
A sanity check for calibration quality (but no proof !) is the Rohde & Schwarz T-Check, which is demonstrated on page T-Check.

Warm-Up, Drift after Power-Up

A calibration can only correct errors that remain constant over time. Any instrument that is being switched on will eventually warm up, which will cause some drift of the measurement results.

In the following, a T-check DUT is used to produce reflection and transmission at the same time. Both are measured with the VNWA3E at constant frequency over an extended period of time after switching the VNWA on.

At the same time the VNWA CPU temperature is recorded by means of its built-in temperature sensor:

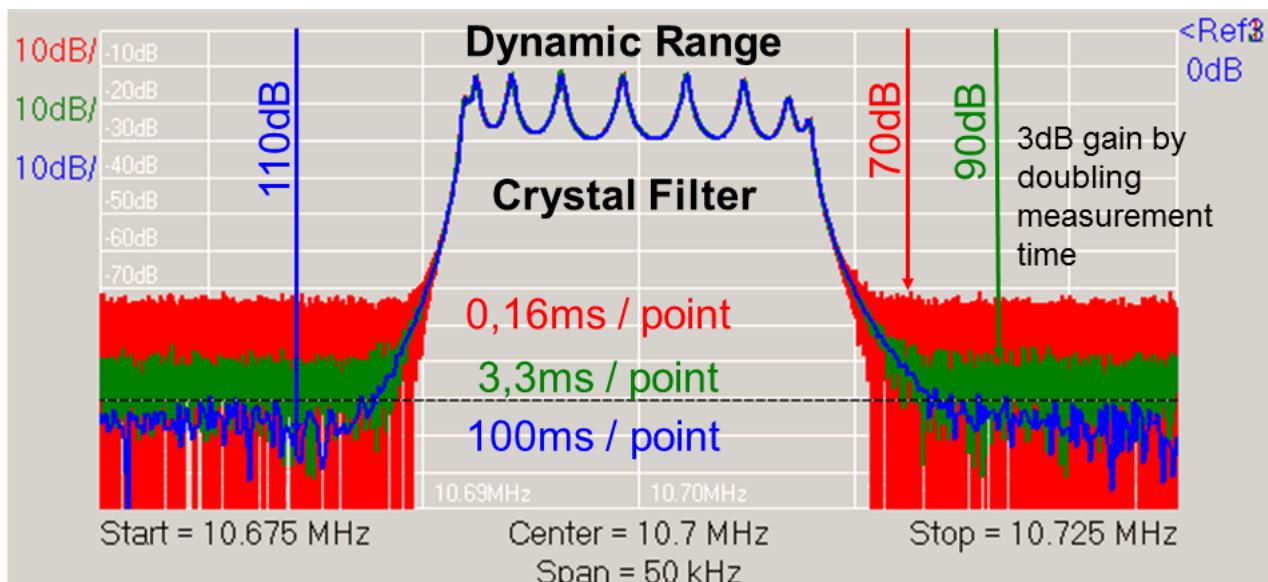


The instrument takes about 30min to warm up. After this time, all amplitudes have settled to within +/-0.01dB of their final values. But even right after switching on the amplitudes deviate less than 0.06dB, which is sufficient for most applications.

Note, that the measured phases do not drift at all within the width of noise. Therefore, the phases are not shown here.

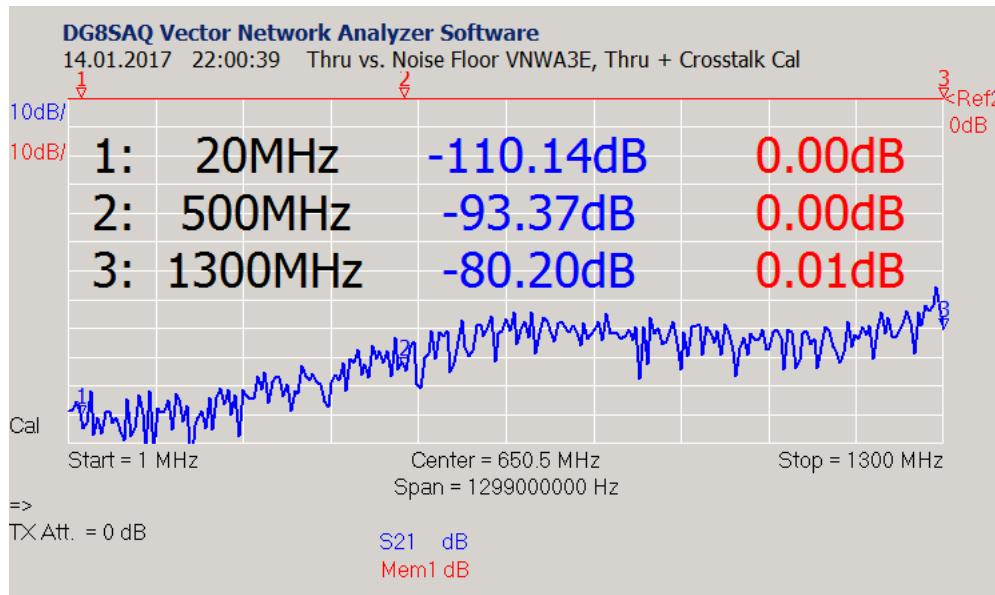
Dynamic Range and Noise Performance

The VNWA dynamic range is defined as the spacing between the maximum signal level the RX detects when the TX signal is directly passed to the RX port and the noise floor measured without RX input signal. Since the noise level depends on the receiver bandwidth, which in turn depends on the selected sweep speed (higher speed means higher noise bandwidth), it does not make sense to specify a dynamic range without stating the receiver bandwidth or the measurement time per frequency point at the same time. This is illustrated by below measurements performed with a VNWA3E on a high attenuation monolithic crystal filter. Note, that the strong passband ripple and the high insertion attenuation is a result of the high filter impedance of the order of 1 kOhms in combination with the VNWA's port impedances of 50 Ohms.



While the highest sweep speed of 0,16ms per frequency point (corresponding to 6000 frequency points per second) yields a dynamic range of 70dB, increasing the measurement time to 100ms / frequency point (corresponding to 10 points per second) increases the dynamic range to beyond 100dB, which reveals filter features previously hidden in the noise floor.

Note, that the signal level at the VNWA TX port is not constant, but decreases from -17dBm (which is also the linear limit of the RX port) at low frequencies with increasing frequency. This will lead to a decreasing dynamic range with increasing frequency. This effect is shown in below measurement, where the open response (noise floor) is compared to the thru response (0 dB):

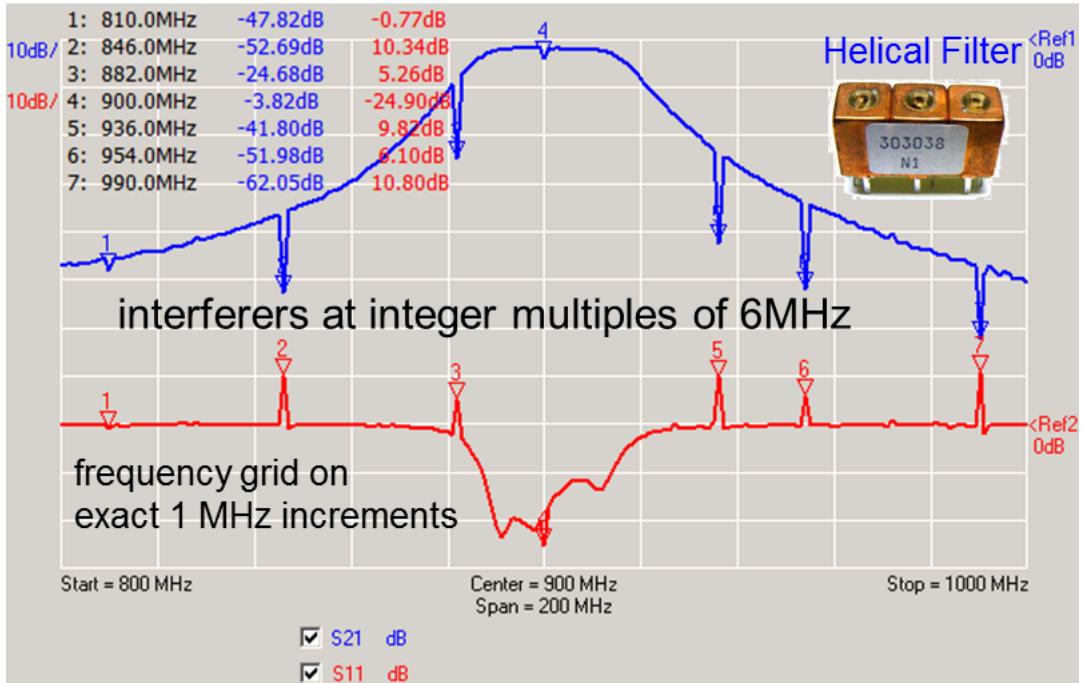


Note, that above measurement was performed at the slowest possible sweep speed of 100ms per frequency point. Both, a thru calibration and a crosstalk calibration was applied to work out the noise floor. The red thru response is flat, because the TX output level variations have been normalized by the Thru-calibration.

Spurious Responses

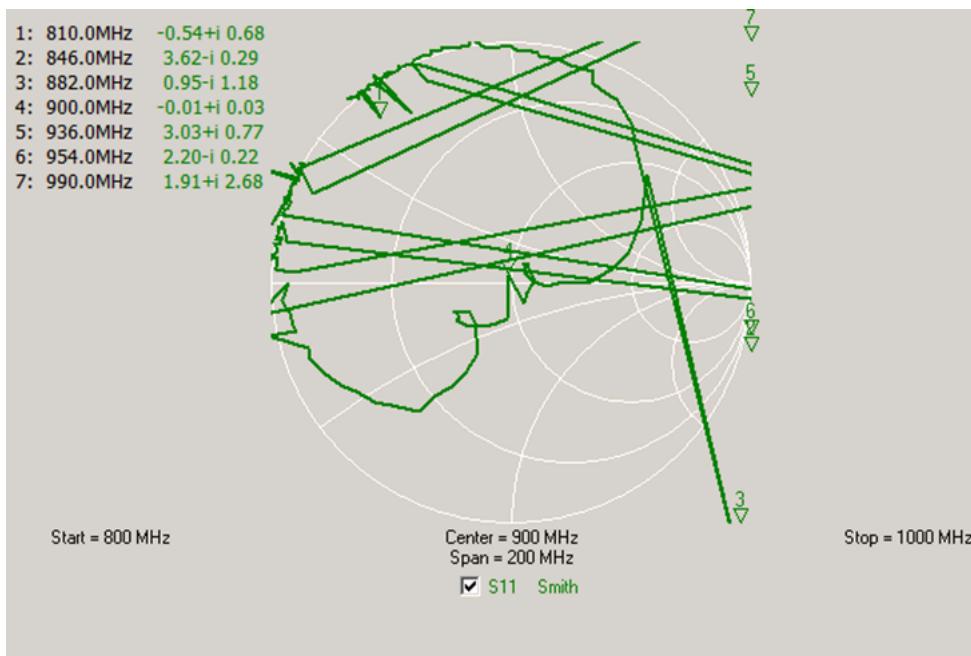
Narrow Band Interferers

In order to achieve the highest possible frequency coverage, the VNWA was designed without any frequency selective filters in the RF chain. This allows to use the DDS alias frequencies for extended frequency coverage. On the other hand, the lack of filtering will introduce some narrow band interferences. These will show as erratic spikes in the measurement data and occur at some integer multiples of half the fundamental system clock (6MHz). They are most pronounced above 500MHz and can be disturbing. Here is an example measurement on a 900MHz helical band pass filter clearly showing spikes at some but not all integer multiples of 6MHz:

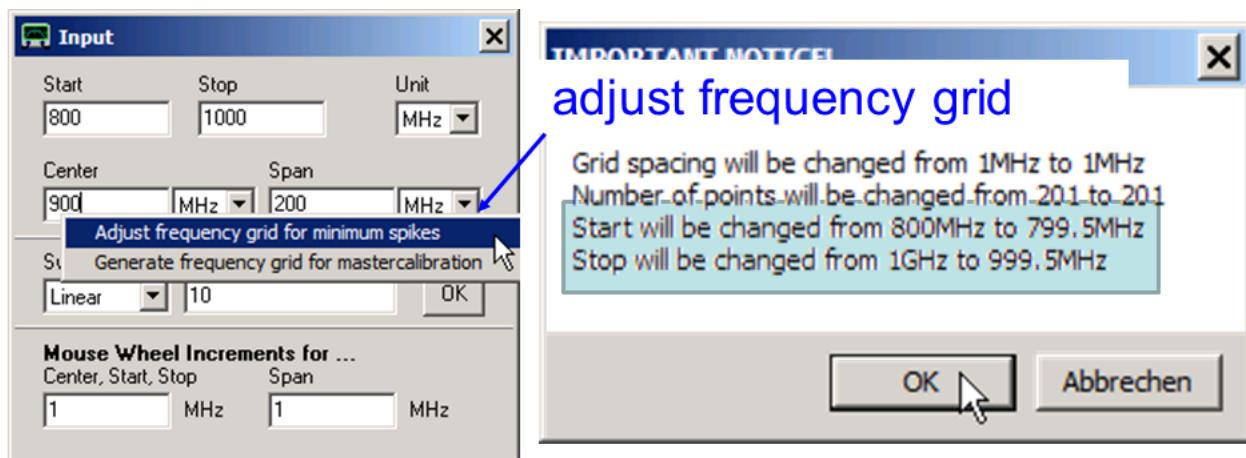


Note, that the spikes were particularly provoked by selecting the frequency grid on exact 1MHz increments starting at 800MHz.

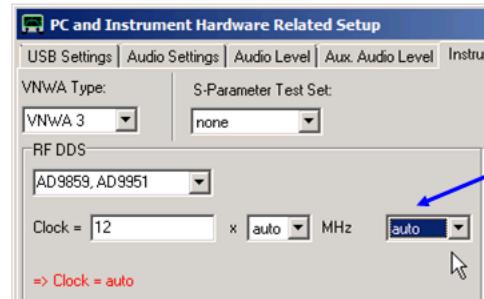
These spikes are even more disturbing in polar plots like the Smith chart, as not only the amplitude but also the phase is erratic at these points:



The way to provoke these spikes also leads the way to avoid them. The idea is simple: When sweeping, avoid integer multiples of 6MHz in the frequency grid. Moreover, the number of potential spikes can be reduced by switching to an auto-premultiplier scheme for clock multipliers. Both measures are being applied here:



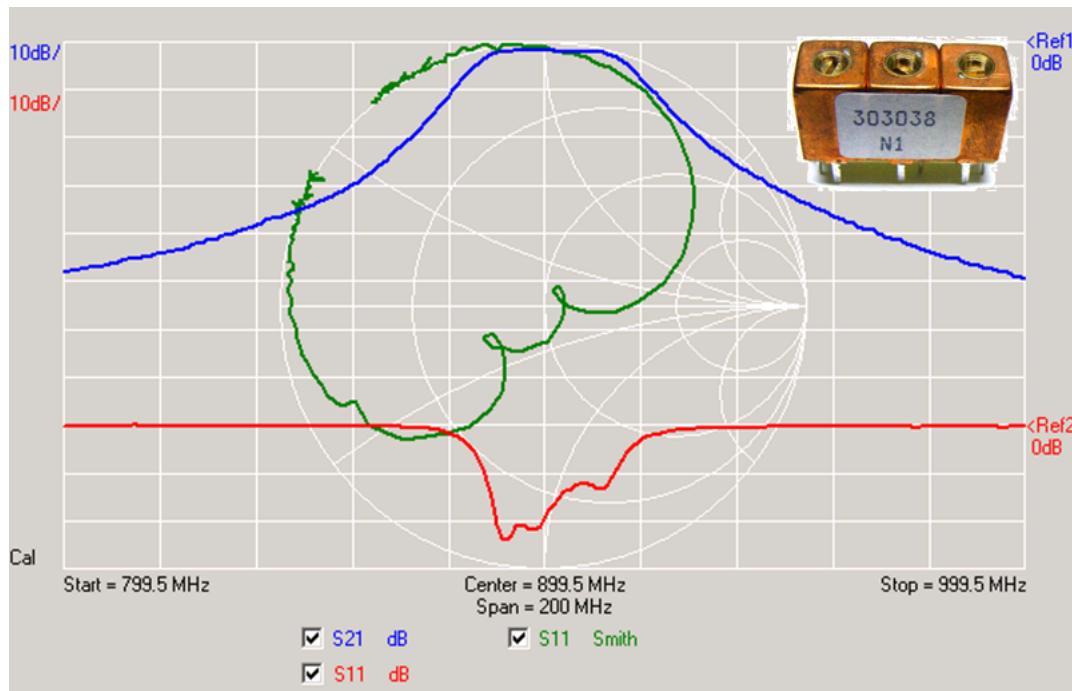
optimize clock multiplier



premultiplier auto

Note, that adjusting the frequency grid for avoiding spikes has been automated in the VNWA software. The corresponding menu can be activated by right-clicking any frequency input field above.

The result of above measures is a complete absence of disturbing spikes.



Note, that the frequency grid optimizer simply shifted the complete grid by 500kHz thus avoiding all integer multiples of 1MHz and particularly of 6MHz.

Wideband Interferers

There is also some very weak wideband interference caused by the digital synthesizers. Normally this goes unnoticed, because either the wanted measurement signal is many orders of magnitude stronger than the interferers (e.g. when doing reflection measurements) or the DUT itself filters the measurement signal (e.g. when doing transmission measurements on a bandpass filter).

However, there are a few rare cases, where the weak wideband interferers have noticeable influence on the measurement. One extreme example is a transmission measurement on a UHF notch filter or a diplexer with a deep notch level. Here, the full spectrum of the DDSes reaches the receiver with only a tiny fraction of the spectrum missing due to the notch. Now, the full spectrum of interferers will to some extent "fill" the notch, so it looks less deep than it is. An additional band pass filter or low pass filter can counteract this effect.

Kurt Poulsen OZ7OU has performed detailed analysis of this effect in the following papers:

"Measurement of a 6 cavity duplex filter of brand Procom and type DP2/6S equivalent to DP2/S6H"

<http://www.hamcom.dk/VNWA/Measurement%20of%20a%206%20cavity%20duplex%20filter%20of%20brand%20Procom%20and%20type%20DP2-6S.pdf>

"Why VNWA2 and VNWA3 does not show correct attenuation of high and low pass filters or what limitation exist for the VNWA design compared to a professional R&S or HP VNA"

<http://hamcom.dk/VNWA/Why%20VNWA2%20and%20VNWA3%20does%20not%20show%20correct%20attenuation%20of%20high%20and%20lowpass%20filters.pdf>

SWEET MODES

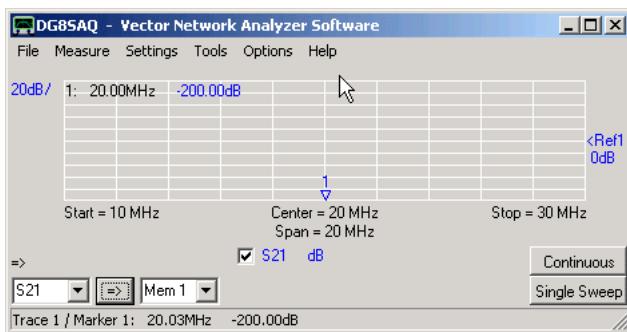
The VNWA software allows to do frequency sweeps in the following sweep modes:

- 1) **Linear sweep:** The frequency grid spacing is equidistant.
- 2) **Log sweep:** The grid spacing increases from point to point such that a logarithmic frequency axis is created. Useful for Bode plots.
- 3) **Listed sweep** The user can specify up to 20 different linear sweep segments which build up the frequency grid.

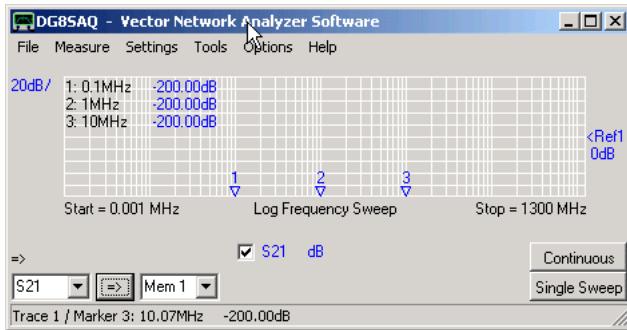
The "sweep mode" is either selected in the frequency input window opening on double-clicking the main window's frequency labels or by right-clicking the center or span label.

Examples:

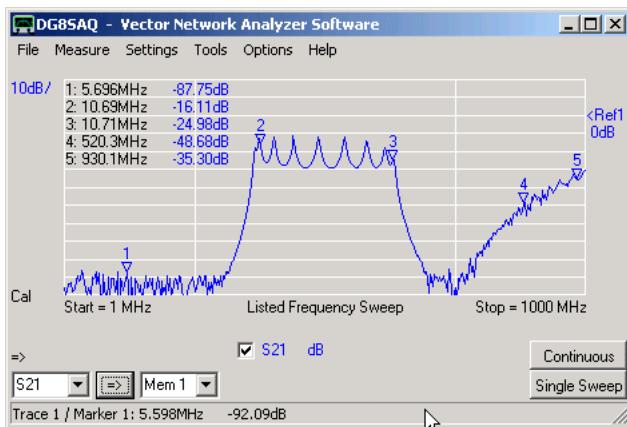
The screenshot below shows **linear sweep** settings, where the center frequency is in the middle of the screen.



The screenshot below shows a Bode type plot with **logarithmic frequency sweep** settings. Looking at the markers, it is seen that with every full frequency grid progression, the frequency increases by a factor of 10. Observe, that instead of the center frequency, "Log Frequency Sweep" is displayed. The sweep mode can be changed by right-clicking this label.



The screenshot below shows an example of a **listed sweep**. The DUT (a narrow band crystal filter) is swept from 1 MHz to 1000 MHz using 400 data points, yet the 20 kHz wide filter passband is clearly resolved (see markers). For a linear sweep with the same point number, the spacing between frequency points would be 2.5 MHz and the filter passband would not be seen.



This is achieved with the following frequency list:

Sweep Frequency List Editor				
Seg.#	start frequency	stop frequency	unit	# points
1	1	10.685	MHz	100
2	10.685	10.715	MHz	200
3	10.715	1000	MHz	100
4				

The frequency list specifies 3 frequency segments, which are swept consecutively and displayed simultaneously. Note, that in the VNWA main window segment boundaries are marked with vertical grid lines.

- »»» **Hint:** The sweep "**sweep frequency list editor**" opens automatically, when "listed frequency sweep" is selected for the first time. Right-click a line to insert or delete a segment.
- »»» **Hint:** The sweep "**sweep frequency list editor**" can be reopened by right-clicking or double-clicking the "Listed Frequency Sweep" label below the main graphics.
- »»» **Hint:** Frequency lists can be stored and reloaded to/from ***.csv files**, which can also be processed with Excel.

ULTRAFAST SWEEPING

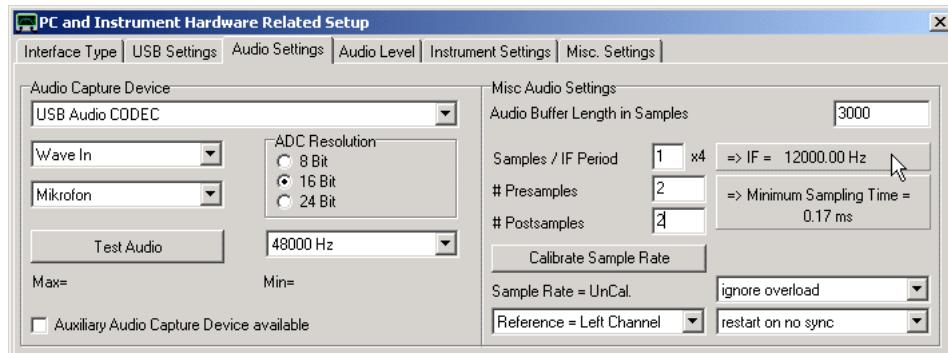
Note, that the selected VNWA IF frequency determines the upper limit of the sweeping speed. The reason is that in order to determine the amplitude and phase of a signal at least one IF period has to be sampled. With the default setting of 1.2 kHz IF the minimum measurement time per frequency point is of the order of 1 millisecond.

Also note, that the "dead" delay time between consecutive sweeps limits the sweep rate. This dead time could be reduced dramatically with software version 35.8.d. Therefore **ultrafast sweeping really only makes sense with software version 35.8.d or a newer one**, otherwise the delay between consecutive sweeps might take a lot longer than the sweep itself.

The VNWA can sweep as fast as 6000 frequency points per second. In order to use such ultrafast sweep rates, the IF frequency must be increased from the default value of 1.2 kHz to the maximum possible value of 12 kHz.

Note, that a change of IF frequency will make your calibrations invalid.

The IF can be changed in the main menu "Options"- "Setup"- "Audio Setup" tab:



The **minimum measurement time per frequency point** is determined by the settings "Samples / IF Period", "# Presamples" and "# Postsamples". Above settings are the recommended settings for obtaining the minimally possible measurement time per data point.

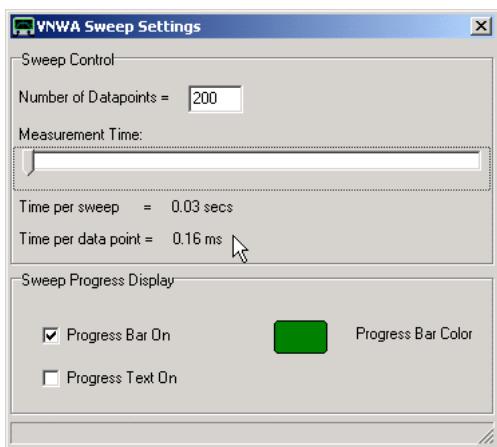
- "Samples / IF period x4" specifies the IF frequency. Note, that the entered number is multiplied by 4. So, if 1 is entered, there are 4 samples per IF period. At a sample rate of 48 kHz, this leads to an IF of 48 kHz / 4 = 12 kHz, which is displayed next to the edit field. The highest selectable IF is 12 kHz at 48 kHz sampling rate. Note, that higher sampling rates theoretically allow higher IFs, but the bandwidth of the USB sound codecs in use is limited to 20 kHz.

- "# Presamples" specifies the number of samples, that are skipped after a frequency change. This number should be as high as possible to avoid timing problems but on the other hand as small as possible to allow for fast sweeping. Setting it to 2 is a good compromise which still allows the maximum sweep rate.

- "# Postsamples" specifies the number of samples, that are skipped before a frequency change. This number should be as high as possible to avoid timing problems but on the other hand as small as possible to allow for fast sweeping. Setting it to 2 is a good compromise which still allows the maximum sweep rate.

The **sweep rate of continuous sweeps** is also determined by the delay between end of sweep and start of the next sweep. This delay amounts to minimum one audio buffer. With above settings (Audio Buffer Length = 3000 samples and 48k samples per second), this delay will be at least 3k/48k seconds or about 60 milliseconds. If the PC is fast enough, this delay can be reduced by making the Audio Buffer Length smaller, i.e. by choosing the smallest allowed value of 800 samples. Note, that a short audio buffer only works well for a small number of frequency points per sweep (e.g. 200) where the CPU loading due to plotting is low.

►►► **Note: Above setup changes will not automatically make the VNWA sweep any faster.** In order to switch on the fast sweeping mode, you must open the main menu "**Settings**"- "**Sweep**" and move the **measurement time slider** to the left. While for the default setup the minimum settable time per frequency data point is 1.3 milliseconds, above settings allow to reduce the time per frequency data point to 0.16 milliseconds:



SIMULTANEOUS CAPTURE WITH TWO SOUND CARDS

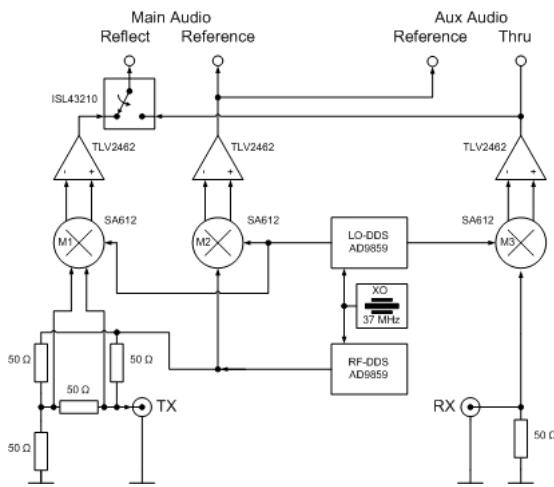
►►► **Note:** This feature is only available in USB-mode!

In its original design, the VNWA cannot measure S11 and S21 simultaneously, but with the aid of a multiplexing switch these can be measured alternately.

The reason for this lies in the fact, that three signals must be captured for a simultaneous measurement (Reference, Reflect, Thru), but a standard stereo sound device as used in the VNWA base design has only two input channels.

This restriction can be overcome by **adding a second sound device** to the measurement system, thus adding two audio channels. As the sound devices are not running synchronously in any way, both need to receive the Reference signal. This still leaves two free channels for the Thru and Reflect signal to be captured simultaneously:

Hardware Modification for VNWA2



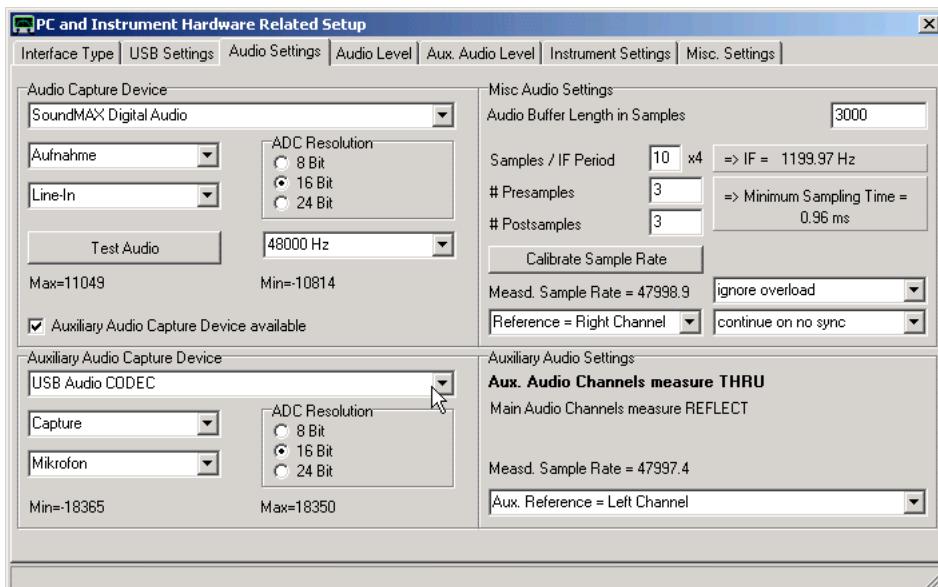
As can be seen above, the main audio capture device remains unchanged and is used to measure S11. The multiplexing switch will remain fixed in the position shown above at all times.

In order to measure S21, the RX OpAmp (behind M3) must be tapped to provide the Thru signal for the second, auxiliary sound card. Make sure to properly DC decouple the OpAmp output to the aux audio input by means of a capacitor (e.g. 10uF) if it is not DC decoupled by itself. Note, that the auxiliary sound card will also need the reference signal again, which can be tapped at the main audio connector.

►►► **Note:** VNWA3 users just need to add the VNWA3 expansion board. VNWA3E users have this expansion board already mounted.

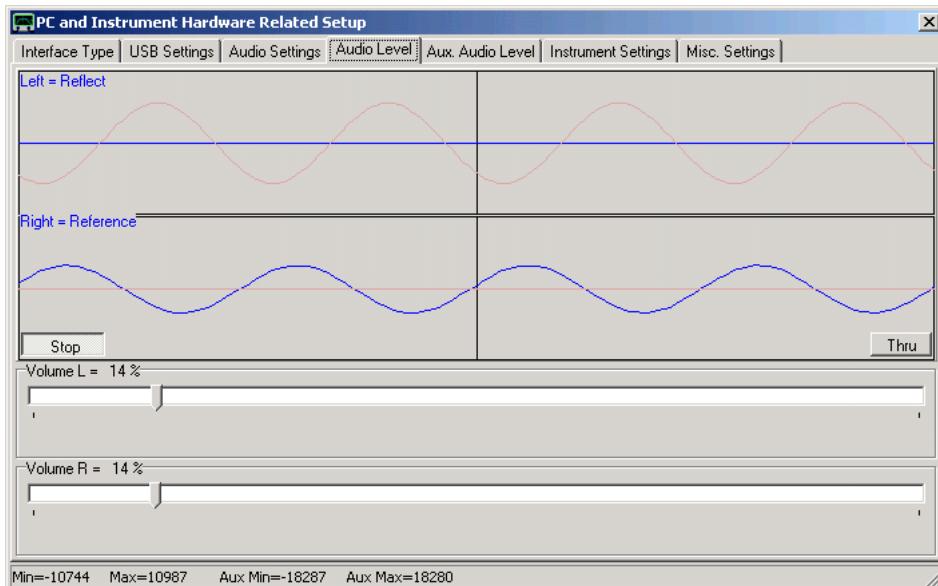
Activating and Configuring the Auxiliary Sound Device

Go to "Setup-Audio Settings" and update the settings according to your modified hardware. Don't worry about the settings on the right hand side, yet.



Note, that generally, unlike shown above, the built-in USB codec must be the main capture device, as it is hard wired to the multiplexing switch.

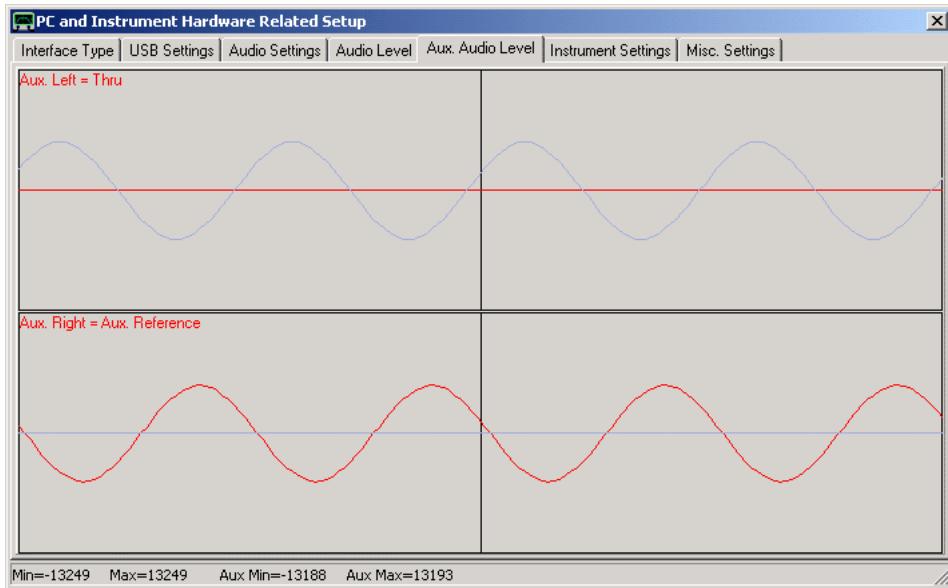
Next go to the "Audio Level" tab and press "Test Audio".



First check if you have chosen the correct main sound device by pressing the Thru/Reflect button with nothing connected to the VNWA. If one of the blue traces changes when toggling the button, the sound device is the correct one.

In Thru-mode with no cable connection between TX-port and RX port, you should see the **blue main reference signal** and a **flat blue line**. Don't worry about the faint red aux audio signals, which are also visible. If the blue sinewave signal is not marked as "Reference" like seen above, left-click onto the blue label denoting the sinewave signal. It will then change to "Reference". Thus you have selected the appropriate channel as main reference channel. Press the Thru-button to check if the reflect signal is working properly.

Next go to the "Aux Audio Level" tab.



Without TX to RX through connection, you should see the red aux reference sinewave signal only. Make sure that this is selected as aux reference signal as seen above (if necessary) by clicking onto the red label denoting the sinewave signal and observing the correct description. Next connect a coax thru connection from TX to RX port and make sure you do see an unclipped thru signal as well.

Next, go back to the "Audio Settings" tab and perform a sample rate calibration. Note, that both sound cards must be calibrated as they might have slightly differing sample rates as can be seen above.

Finally, close the setup.

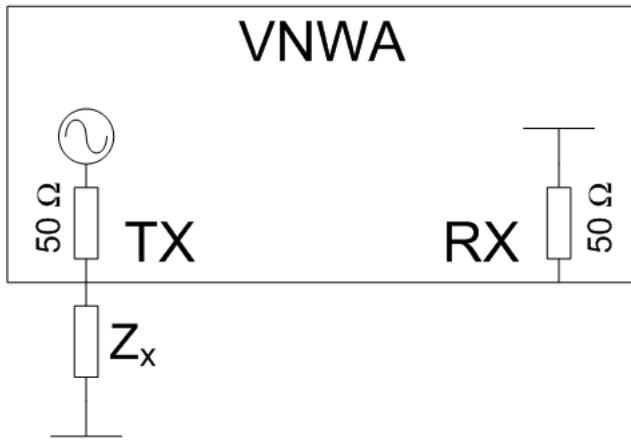
Perform calibrations and measurements in the normal fashion. From now on S21 and S11 traces will update with new data simultaneously, thus cutting the measurement time down by a factor of two.

HOW TO MEASURE IMPEDANCES - GENERAL CONSIDERATIONS

The VNWA is very well suited to measure impedances and admittances. There are several ways to accomplish this with largely varying accuracy depending on the impedance range to be measured. Therefore, it is worthwhile to pick the right method for the impedance range in question.

1. The Standard Method

The most straight forward method is to connect the impedance to be measured (Z_x) to the TX port of the VNWA:



The VNWA measures the reflection coefficient (S_{11}) of Z_x relative to its 50 Ohms environment.

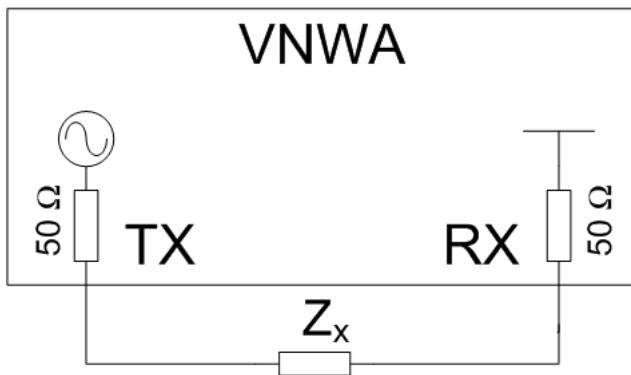
The setup can be Short/Open/Load-calibrated to remove systematic errors.

By virtue of the reflection bridge used which is tuned to 50 Ohms, this method is **most accurate when Z_x is near 50 Ohms**.

For much smaller or much bigger impedances, the test setup becomes very insensitive and thus the results inaccurate.

2. The I-Method

Here, the TX port serves as a voltage source:



The VNWA RX port is used to measure the current I through Z_x .

As S_{21} is measured, a simple Thru-Calibration is sufficient in this case.

Use the custom function $t2s(S_{21})$ to convert the S_{21} transmission data into reflection data.

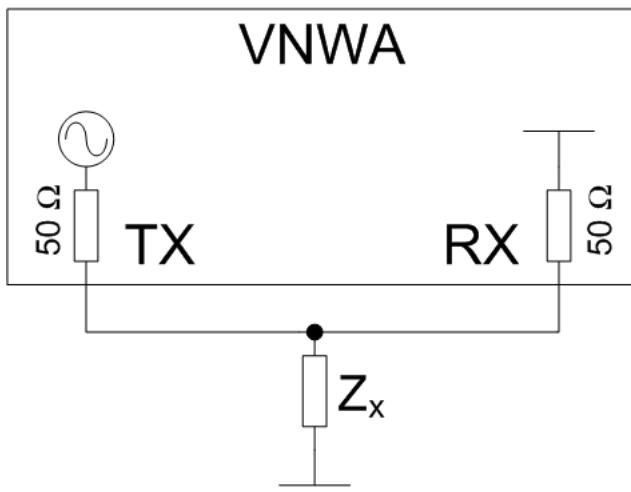
Since the TX port is not an ideal voltage source and since the RX input has finite impedance, this method becomes very insensitive and thus **inaccurate for small impedances Z_x** .

This method is well suited to measure crystals or losses in small capacitors.

►►► **Note:** None of the Z_x ports may be grounded for this method.

3. The V-Method

Here, the TX port serves as a current source:



The VNWA RX port is used to measure the voltage V across Z_x .

Use external bridge mode to perform this kind of measurement.

The setup can be Short/Open/Load-calibrated to remove systematic errors in the standard way.

Since the TX port is not an ideal current source and since the RX input has a quite low impedance, this method becomes very insensitive and thus **inaccurate for large impedances Z_x** .

4. The RF-IV-Method

The RF-IV method combines the I-method and the V-method by measuring both current through Z_x and voltage across Z_x in two consecutive measurements. Then, these measurements are combined to yield $Z_x = V/I$.

Thus, the method combines the advantages of the I- and the V-measurement, i.e. high sensitivity and thus **high accuracy for all possible values of Z_x** .

In order to use this method, an external RF-IV test head is required. See page **RF-IV Mode** for details.

SUMMARY

The following table lists the change in detector voltage for a 10% increase of Z_x for several Z_x values:

Z_x	S_{11}	I	V	V/I
0,1 Ω	-0,04%	0,01%	-9,96%	9,97%
51 Ω	480,68%	3,27%	-3,08%	6,56%
100 kΩ	0,01%	9,08%	0,00%	9,99%

Clearly, only the RF-IV method yields high sensitivity over the full impedance range. Nevertheless, the other methods will yield accurate results, if the right method is chosen for the particular impedance under test Z_x .

TIME DOMAIN MEASUREMENTS

The VNWA software allows to perform a realtime **FFT** on measured or imported data from frequency to time domain, display data in time domain, manipulate data in time domain by **gating** and perform an **inverse FFT** back to frequency domain.

A special topics section on time domain measurements demonstrates examples of both usages:

Time domain reflectometry is a useful technique to search defects in transmission lines.

The step response is useful to determine impedance variations along a coaxial cable.

Gating is used to separate responses depending on their arrival times, e.g. to separate the slow mechanical response of a crystal filter from the fast electromagnetic feedthrough of the test board.

INTRODUCTION TO TIME DOMAIN MEASUREMENTS

| Time Domain Reflectometry

Time domain reflectometry (TDR) is a straight forward and widely used technique to find faults in cables and transmission lines. The idea is to **send a short voltage impulse** into the line and **detect the reflected impulse(s)** which is the so called **impulses response**.



Every not correctly terminated end of the line will reflect some fraction of the incoming impulse. Also, cable damages like breaks or compressions will cause part of the incoming impulse to be reflected. By measuring the time delay of the impulse reflected by the damage one can calculate the location of the damage if the impulses velocity through the cable is known. The impulse in question must travel the distance from cable input to the fault and back. The impulse velocity (=speed of light * velocity factor) can be determined experimentally as well by observing the reflection from the far end of the cable and by measuring the length of the cable.

A **voltage step** is equally well suited as input signal for time domain reflectometry and it is technically simpler to generate. The reflected signal of a voltage step is called the **step response**.

In fact, every impedance change in a transmission line will cause some of the incident signal being reflected in a characteristic way:

Component	Step Response	Impulse Response
$\Gamma = 1, \text{Open}$		
$\Gamma = -1, \text{Short}$		
Resistor, $r > Z_0$		
Resistor, $r < Z_0$		
Inductor		
Capacitor		

(Image from Anritsu presentation "Time Domain Analysis Using Vector Network Analyzers" by Dr. Martin I. Grace)

| Time Domain vs. Frequency Domain

An impulse or a step signal $g(t)$ consists of a spectrum $G(f)$ of an infinite number of frequencies f . The spectrum can be calculated with the **Fourier transform**:

$$G(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(t) \cdot e^{-i\omega t} dt$$

with the angular frequency

$$\omega = 2\pi f$$

Example: An ideal Dirac impulse (width zero, infinite amplitude) will produce a constant spectrum which contains all (angular) frequencies from - infinity to +infinity with equal strength.

If the spectrum is known, the impulse shape can be reconstructed using the **inverse Fourier transform**:

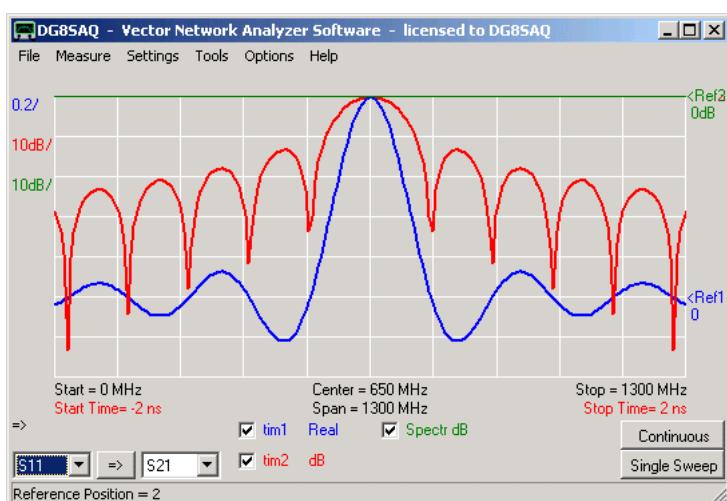
$$g(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} G(\omega) \cdot e^{i\omega t} d\omega$$

The same holds true for any time signal $g(t)$. Thus, the time signal and the corresponding spectrum or frequency signal contain the same information.

While a classical time domain reflectometer can generate impulses and measure the impulse response $g(t)$ in time domain, a VNA cannot do this. On the other hand, a VNA can generate sine waves with almost arbitrary frequencies and thus measure the frequency response $G(f)$. Since time domain response and frequency domain response are interrelated through the Fourier transform, **the impulse response of a system can basically be reconstructed mathematically from the frequency response measured with a VNA**.

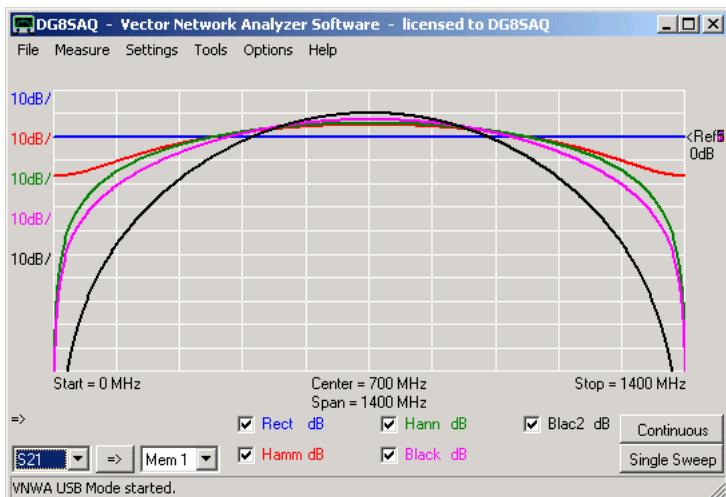
Finite Frequency Range Problem, Windowing

Any existing VNA has only a limited frequency range. Outside this frequency range a system's frequency response cannot be measured. **Reconstructing a time response from such a frequency limited response will generally produce artefacts**. The following graph shows an impulse reconstructed from a spectrum obtained by limiting the spectrum of an ideal impulse to the displayed frequency range:

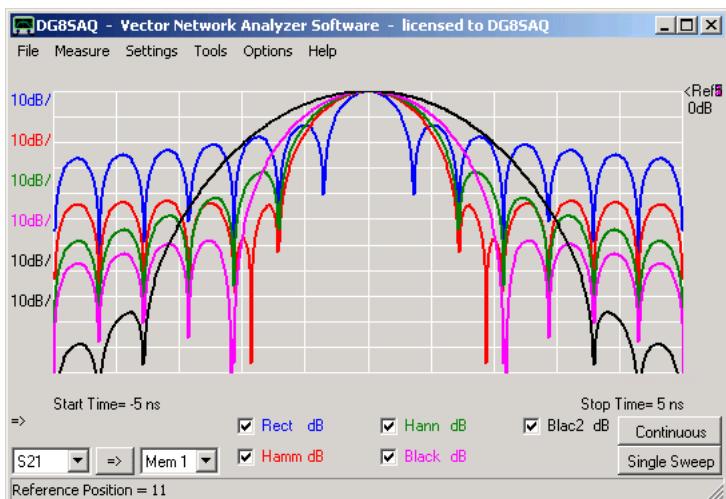


The blue impulse reconstructed from the band limited green spectrum is widened as compared to an ideal Dirac impulse and it shows sidelobes according to the law $\sin(t)/t$. The red trace shows the same reconstructed impulse in dB scale. Note that the zeros of the blue reconstructed impulse produce very distinct notches in the blue dB scale. **These notches and sidelobes are not related to any test object but are merely a consequence of the limited bandwidth**. These sidelobes might well cover up some real lower level DUT response. In order to detect such covered low level response the windowing technique was invented. By weighting the frequency response by appropriate windowing functions (e.g. by Hamming, Hanning, Blackman...) the sidelobe level can be considerably reduced. The tradeoff is a reduced time resolution, i.e. the reconstructed impulse becomes wider:

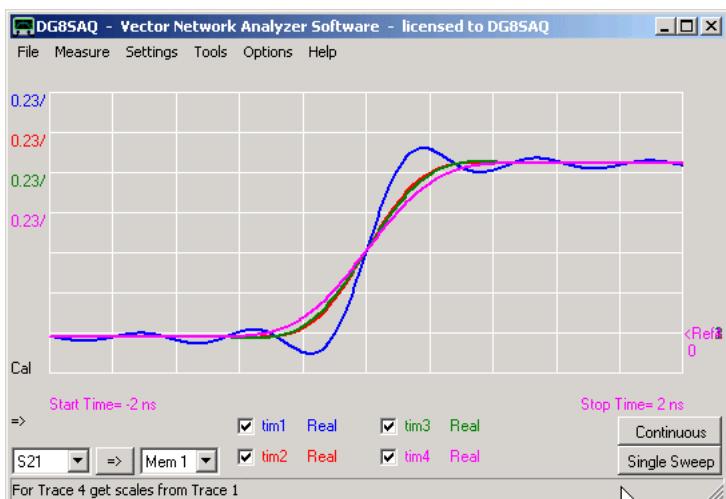
Implemented windowing functions (rectangular=none, Hamming, Hanning, Blackman, Blackman squared):



Effects of applying the above windowing functions:



Note, that the narrower windowing functions lead to wider reconstructed impulses with generally lower sidelobe levels. The same behavior can also be observed in the step response:



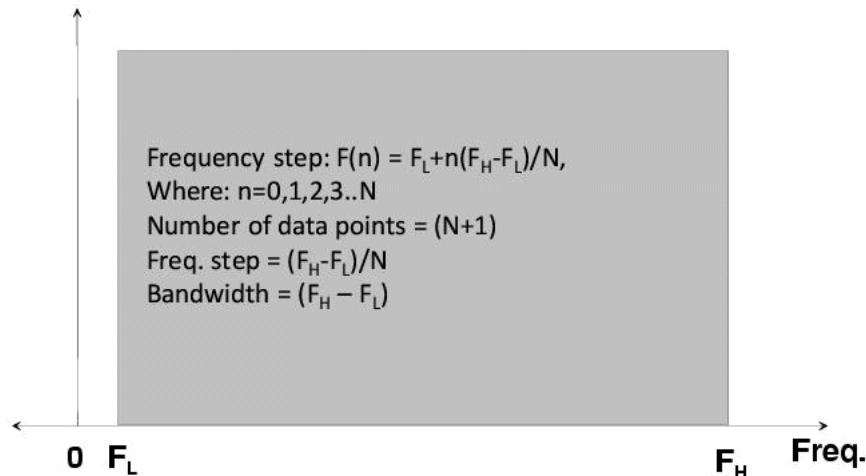
Colors in above diagram correspond to the window functions in the previous diagrams. Sidelobes and impulse width in impulse response correspond with overshoot and step steepness in step response.

Band Pass vs. Low Pass Mode

There are two possible ways to process a measured frequency response in order to obtain a time response, namely band pass and low pass processing.

Band Pass Processing:

In band pass mode, the utilized frequency range is the VNA frequency span.

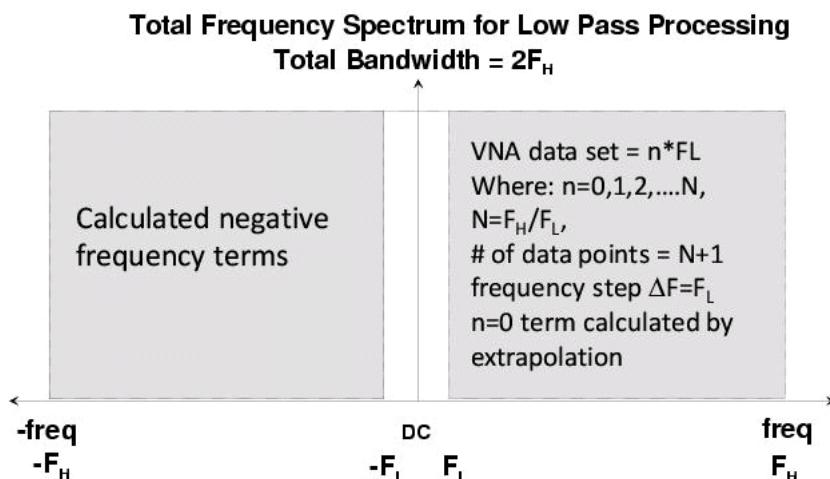


(Image from Anritsu presentation "Time Domain Analysis Using Vector Network Analyzers" by Dr. Martin I. Grace)

This mode ignores negative frequencies. As a consequence of omitting negative frequencies the reconstructed time response is unphysically complex valued. Usually, only the magnitude of the band pass response is of interest. It might give useful information for tuning filter structures, see e.g. Agilent AN 1287-10 "Network Analysis Solutions Advanced Filter Tuning using Time Domain Transforms". Note, that window functions will suppress both the high frequency and the low frequency content of the frequency response in this case. As a voltage step contains strong low frequency components, **band pass mode cannot be used for calculating a step response**.

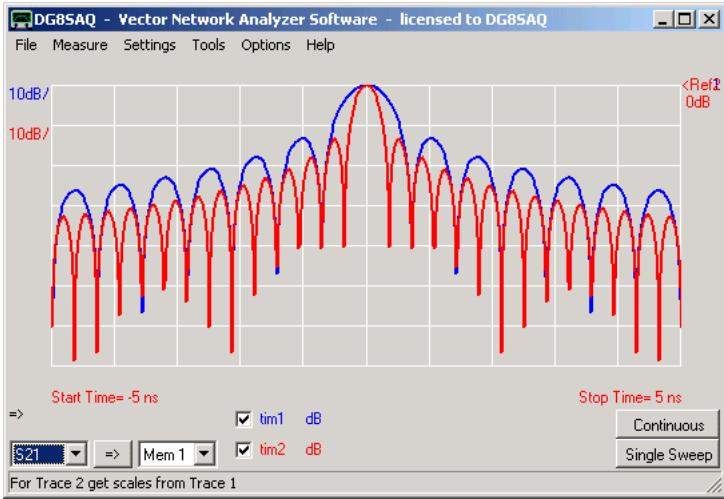
Low Pass Processing:

Low pass mode makes use of the inherent symmetry of the frequency response with respect to frequency zero. Thus, having measured the response for positive frequencies, the response for negative frequencies can be reconstructed mathematically and can be used to calculate a real valued time response, i.e. the imaginary part is zero as expected for a time response. Note that the measured frequency span should extend to (almost) zero in order to obtain good time domain results.



(Image from Anritsu presentation "Time Domain Analysis Using Vector Network Analyzers" by Dr. Martin I. Grace)

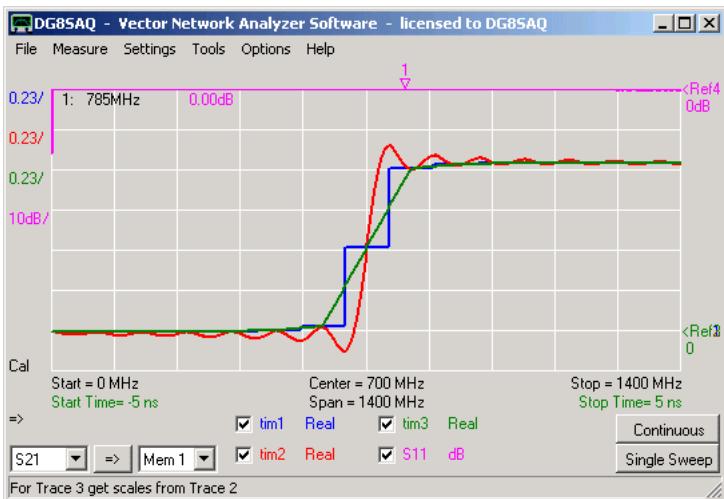
By using this trick, **the utilized frequency bandwidth is doubled and thus the time resolution is improved by a factor of two** as can be observed in the following example showing the reconstructed impulse responses of identical frequency data for **band pass mode** and **low pass mode**.



Note, that for band pass mode the window functions have their maxima located at zero frequency, which makes this mode particularly useful for reconstructing step responses which contain strong low frequency content.

[Fourier Transform Options](#)

There are many algorithms to calculate an inverse Fourier transform for reconstructing a time response from a measured frequency response. The most widely known algorithm is the **FFT or Fast Fourier Transform** algorithm by James Cooley and John W. Tukey. It is very efficient, but it can only calculate a time response at a strictly equidistant grid of times predefined by the frequency span and by the number of data points. If the times of interest lie in between this grid, FFT cannot be used. In such a case, the **Discrete Fourier Transform (DFT)** has to be calculated by some other less CPU efficient algorithm. The VNWA software offers three choices to calculate and display reconstructed time data, namely **FFT**, **FFT interpolated** and **DFT**:



Each **FFT** data point is displayed as a plateau with the point sitting in the center of the plateau.
In contrast, **FFT interpolated** linearly interpolated the FFT data points.

DFT also calculates time data in between the fixed FFT grid and uncovers the oscillations caused by the finite frequency span.

Note, that all traces must coincide on the FFT grid, i.e. at the centers of the blue plateaus.

Note, that the above data shows the step response of the open VNWA TX port reconstructed from the displayed S11 measurement without window function, i.e. rectangular window with maximum width.

[Nyquist Criterion and Aliasing:](#)

When doing a frequency domain measurement with frequency span s , which extends over n frequency points, the frequency spacing between adjacent frequency points is $d=s/(n-1)$.

When a DUT is measured with these settings in reflection, where the DUT is reflecting the incident signal with a time delay t , then the phase progression from one frequency point to the next will be $dp=2\pi d \cdot t = 2\pi s \cdot t/(n-1)$.

Now, the DUT delay t can only be reconstructed from the reflection measurement with an inverse Fourier transform, if the phase progression dp is less than π or less than half a turn in the Smith chart. This is the Nyquist criterion for the

inverse Fourier transform.

Thus, the following condition must be obeyed:

$$t < (n-1)/(2s)$$

Example:

You are measuring with a frequency span of $s=500\text{MHz}$ with $n=1001$ frequency points.

Thus the condition $t < 1000/1000\text{e}6 \text{Hz}=1\text{us}$ must be obeyed.

This means, that delays of up to 1us can be measured with these settings. Signals with longer delays will be folded back and thus become visible in the -1us...1us time range and thus produce false results. These false signals are also known as alias signals.

Time Resolution:

The time resolution of a time domain measurement obtained by a frequency sweep and inverse Fourier transform is mainly determined by the highest measurement frequency f_{max} .

If bandpass mode and no windowing (=rect. window function) is used, the first zeros of the time signal will be found at $\pm 1/f_{max}$. Generally, the time resolution is better than this because two distinct time peaks can be differentiated even if their spacing is closer than this.

As a rule of thumb, the time resolution is about $1/(10*f_{max})$.

Example:

When sweeping up to 500MHz, $1/f_{max}=1/500\text{MHz}=2\text{ns}$.

Thus, a time resolution of about $2\text{ns}/10=200\text{ps}$ can be expected when using no windowing and bandpass mode.

Note, that using low pass mode will double the time resolution as was discussed above in section Low Pass Processing.

Note, that using non-trivial windowing functions will decrease the time resolution as was discussed in section Finite Frequency Range Problem, Windowing.

TIME DOMAIN REFLECTOMETRY: IMPULSE RESPONSE

Time domain reflectometry measures the impulse response of a DUT, e.g. a transmission line. Classically, one applies a short impulse to the line input and measures the time dependent voltage at the line input. If the line is perfectly terminated, the impulse will not be reflected back to the line input and no second impulse will be measured after the initial one. If the line is broken somewhere in the middle, some of the input impulse will be reflected there. The position of the defect can then be calculated from the delay of the reflected impulse relative to the initial one, if the velocity factor of the transmission line is known.

The VNWA cannot perform impulse measurements, but it can do a wideband frequency measurement and perform an inverse Fourier transform on it, which is equivalent to an impulse measurement.

VNWA time domain reflectometry basics:

The time (and thus length) resolution depends on the frequency sweep span.

►► Hint: Do your frequency sweep with the maximum possible span starting at 0 in order to obtain the highest possible temporal and spatial resolution. With the maximum span of 1.5 GHz a spatial resolution up to a millimeter can be achieved.

The longest time (and thus length) you can measure depends on both, the number of measurement points and on the frequency sweep span.

Because of the periodic nature of the Fourier transform the maximum allowed delay you can measure is *number of points * resolution*. The resolution order of magnitude is 1/frequency span.

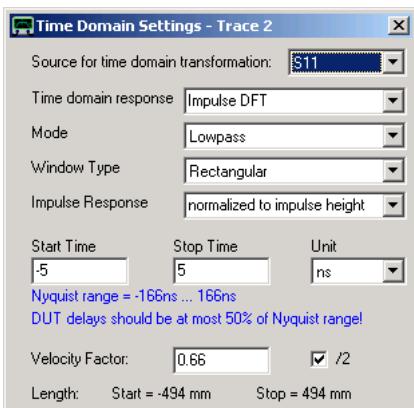
►► Hint: Make sure that you have no signals coming in later than the maximum allowed delay, otherwise those will be folded back in time and create ghost signals. Use a larger number of points if necessary.

►► Hint: If you want to measure really long delays and you have reached the maximum allowed number of points, decrease the frequency span and thus the resolution.

►► Hint: The FFT responses are computed most efficiently if the number of frequency points is chosen to be a **power of 2**. There is no such efficiency benefit for the DFT modes.

Example:

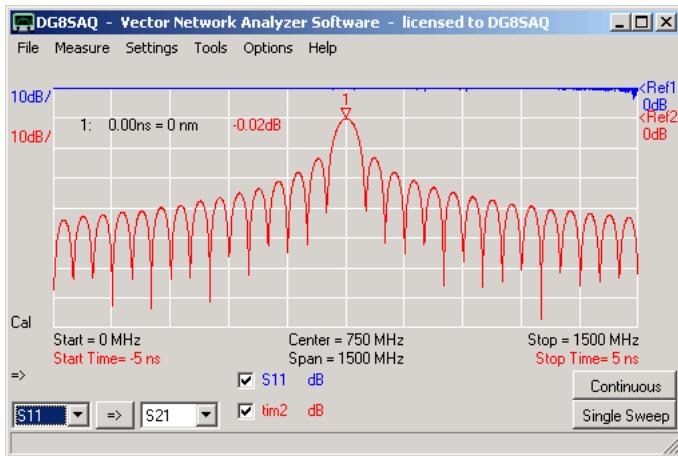
1. Calibrate the VNWA for a reflection measurement (SOL, use auto-clock multipliers). In below example I have used 2000 data points and a 4 milliseconds per data point.
2. Measure the reflection coefficient (S11) of your bare open VNWA TX output and display it in trace1. The result should be one (= 0dB) except for some spurs.
3. Add a second trace with trace type "Time" and open the "Time Domain Settings" via the Settings-Display menu or by the appropriate popup menu on right-clicking the display label.
4. Open the time domain settings window by activating the Settings - Time Domain menu or by double-clicking one of the time labels.
5. Select source **S11**, select "**Impulse DFT**", "**Lowpass**" and "**Rectangular**" for maximum time resolution and edit **start and stop times** as seen below:



Choosing "normalized to impulse height" in this example will make it possible to read the power reflected off a cable discontinuity directly in "dB". Note that the alternate choice "unnormalized" will yield the unnormalized impulse response which has the unit 1/seconds and integration of it will yield the system's step response.

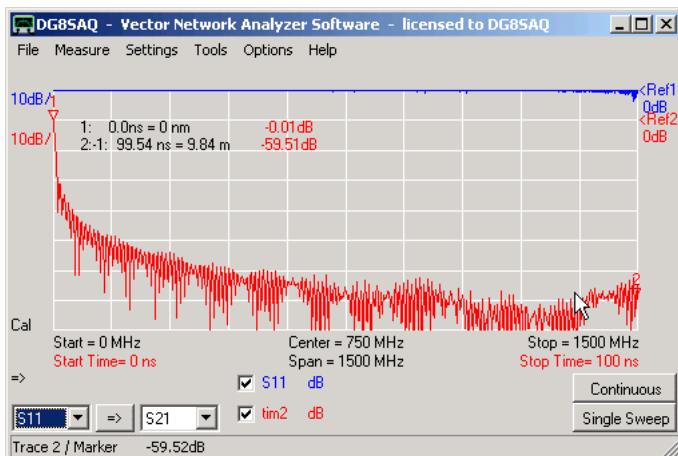
As the VNWA cannot measure below about 25kHz with default settings, "low frequency data extrapolated below 25 kHz" is chosen here, i.e. below 25 kHz the measurement data is replaced by some higher frequency data extrapolated to lower frequency. If reliable low frequency data is available by special settings or from a simulation, choose "as measured" instead of "extrapolated below 25 kHz".

6. Close the time domain settings window. After proper scaling you should see the following result:



The blue trace shows the constant 0dB reflection of the open TX port in the frequency range 0...1.5GHz. The red trace shows the Fourier transform in the time range of -5ns...5ns, which is equivalent to the length range of about -50cm...50cm. We see a sharp peak at length 0 with level 0dB (see marker caption), which means that all of the TX signal is reflected at the calibration plane. The oscillatory $\sin(x)/x$ behavior stems from the fact, that no frequency windowing function is selected.

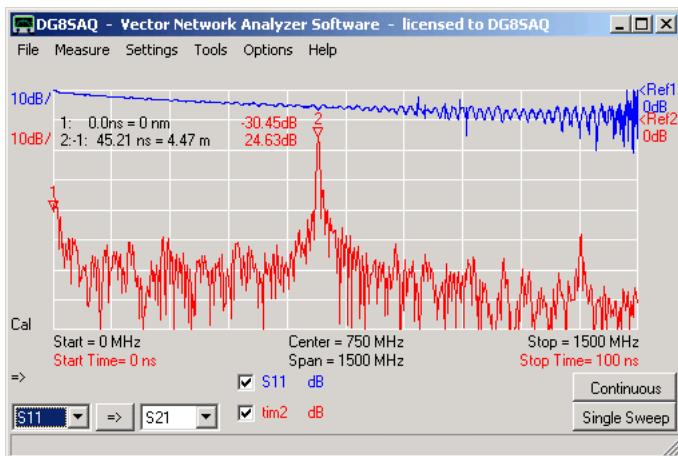
7. After adjusting for a wider time span, you will see the following.



In the above image we look at a length span of about 10 meters. We only see the signal reflected at zero length off the calibration plane.

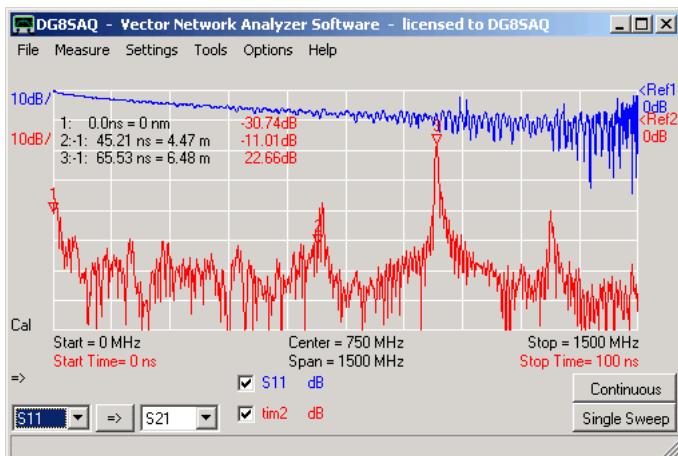
Note, that with 1.5GHz span and 2000 points a maximum reflection length of 150 meters can be measured. The maximum available 8192 data points at 1.5GHz span allow to measure reflection lengths of around 0.6 kilometers with a resolution of a few millimeters.

8. Connect an open ended coaxial cable to the TX port and sweep.



Note, that the reflected signal now comes in attenuated and with a time delay equivalent to the cable length of 4.5 meters (see time delta marker 2). We have thus measured the length of our coaxial cable.
 Note, that there is still weak reflection at zero length coming from the SMA to BNC adapter.
 Note, that there is also a signal at 9 meters delay stemming from multiple reflection.

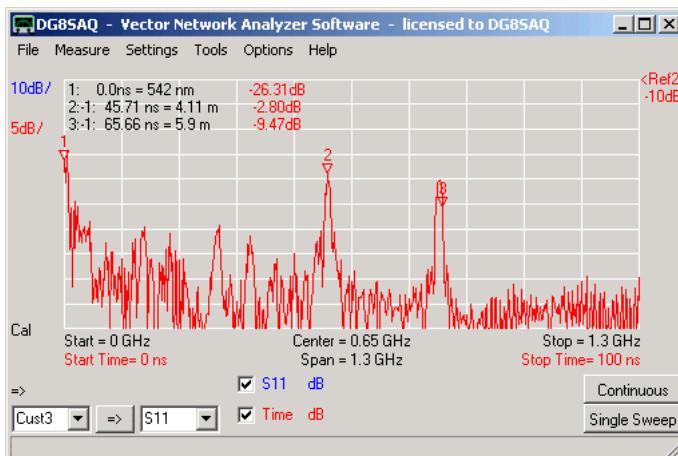
9. Next, I have connected a second open ended coaxial cable to the end of the first one. Note, that the second cable has 90 degree angled BNC connectors:



The reflection is now shifted by the length of the additional cable. The total cable length is now 6.5 meters.
 Note, that some signal is still reflected at the end of the first cable due to the imperfect wave match of the angled BNC connector.

=> **Cable defects can be detected and localized very accurately by this method.**

10. Finally, I have connected a BNC load to the end of the second cable:



Note, that we still see reflections from the connector bend in the middle and at the end of the cable. Also note, that the overall reflected signal has decreased considerably due to the cable termination.

▶▶▶ **Note:** It is possible to box-zoom into a time diagram the same way as zooming into a frequency diagram. **In contrast to frequency zooms, a time zoom cannot be unzoomed**, though. To undo a time zoom, you have to manually change the start and stop times to the original values.

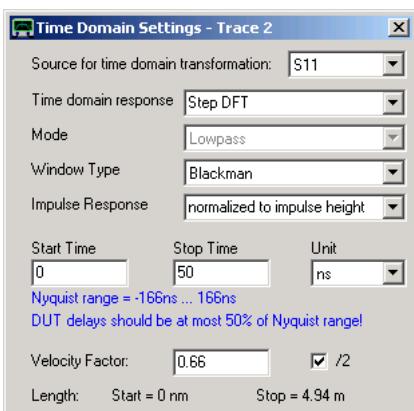
TIME DOMAIN REFLECTOMETRY: STEP RESPONSE

While the impulse response is the DUT's response to a short voltage impulse, it is sometimes useful to look at the DUT's response to a sharp voltage step which is called the **step response**. The two are closely related. E.g. in time domain, integration of the impulse response basically yields the step response. This means, every impedance change at a certain delay will produce an impulse in the impulse response at a certain delay time, which in turn will produce a step in the step response at the same delay time. The useful feature is that the step height is related to the impedance change. **In other words, the impedance change can be calculated from the step height, thus enabling one to see varying cable impedances at varying delays or positions.**

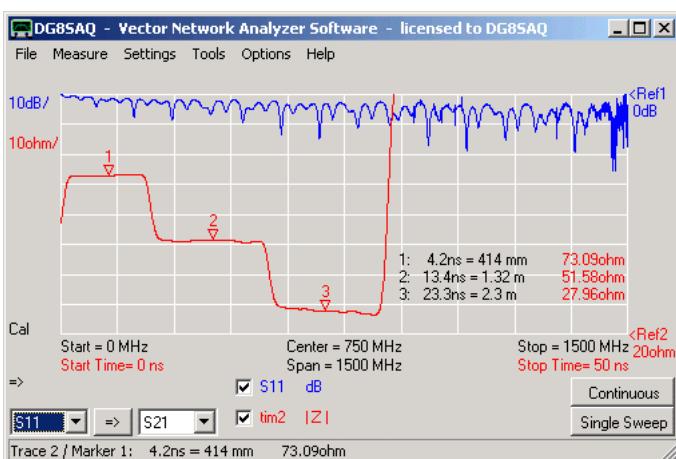
This is demonstrated in the following with a chain of coax cables of different impedances. The DUT below consists of a piece of 75 Ohms coax connected to the VNWA TX port followed by a 50 Ohms coax followed by a 25 Ohms coax the far end of which was left open.

First, calibrate your VNWA in the maximum available frequency range (0-1500 MHz) for maximum spatial resolution with a reasonably high number of points (2000 points used in this example) to separate all temporal features. Next, measure the reflection coefficient (=S11) of your DUT, so the result is visible in the S11 data space.

In order to see the step response, select **trace type "Time"** in the main window and select "**Step DFT**" from the "Time Domain Settings" menu. For displaying smooth steps without artificial overshoot, a **Blackman window function** was chosen. We use low frequency data "**extrapolated below 25 kHz**" to suppress the zero frequency artefact of S11 to enter the time domain analysis.

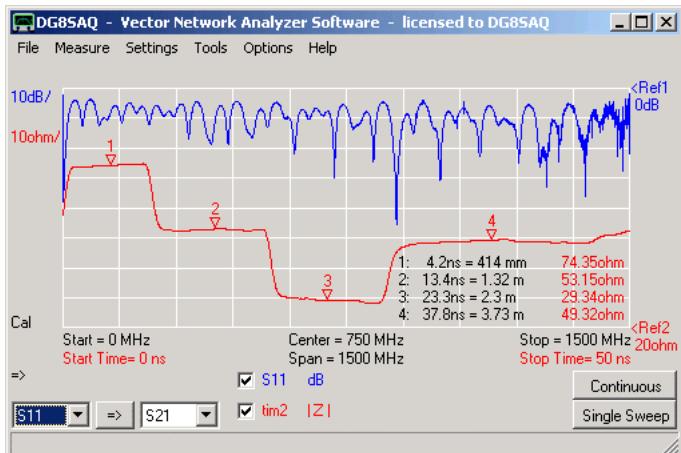


If you select to display the time data in terms of impedances ($|Z|$ in below example), you can immediately read off the varying cable impedance at varying positions:



Note, that the impedance jumps from 75 Ohms to 50 Ohms to 25 Ohms are clearly visible. Also, the jump at the open end to (almost) infinite impedance is visible.

If the far cable end is terminated with 50 Ohms instead of leaving it open, the jump to (almost) infinity vanishes and the terminating impedance of 50 Ohms becomes visible.

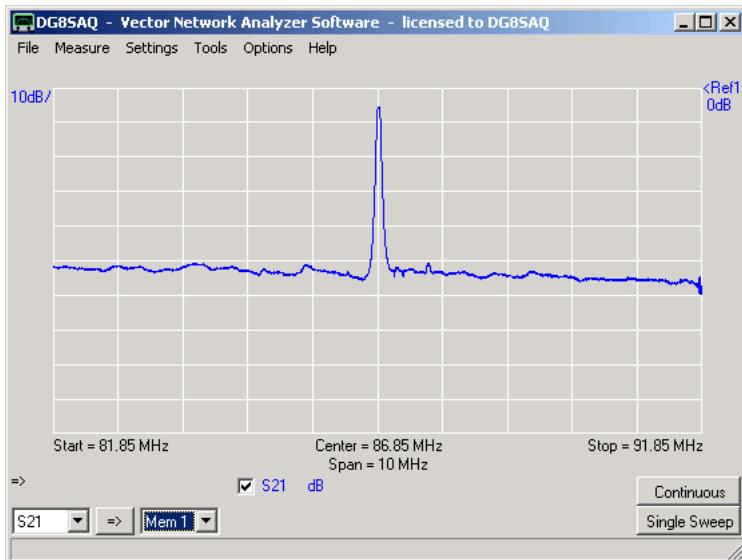


Bear in mind, that the impedance values determined this way will be influenced by multiple reflections and also by cable losses, e.g. no matter what you connect to the far end of a 75 Ohms transmission line with e.g. 100dB loss, you will always see 75 Ohms impedance at all delays.

TIME DOMAIN GATING AND INVERSE FFT

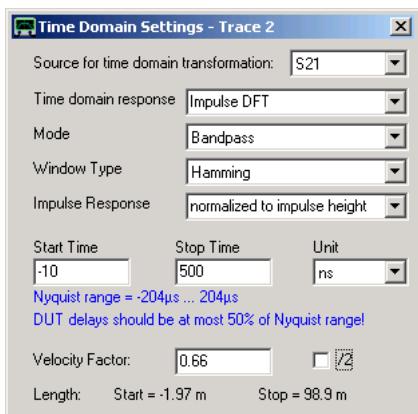
Time domain gating is used to separate responses depending on their arrival times, e.g. to separate the slow mechanical response of a crystal filter from the fast electromagnetic feedthrough of the test board. The following example shows how to use this feature.

1. Measure or import a trace to any memory space.



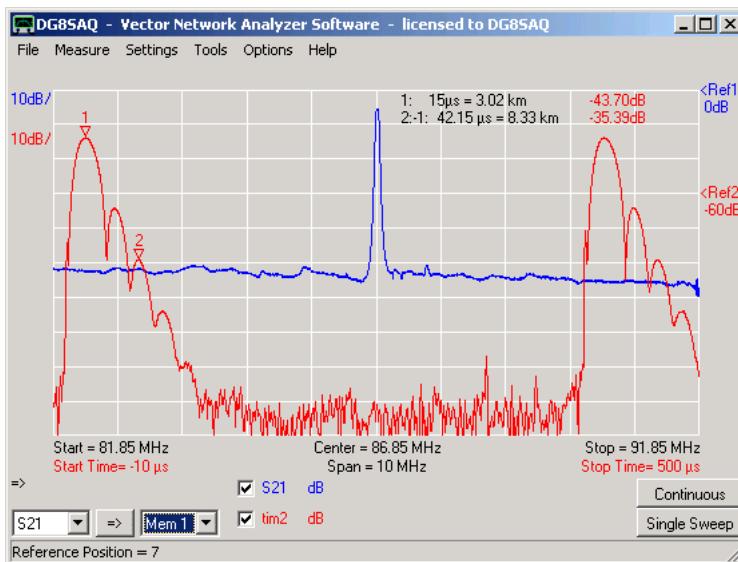
As an example you see a wideband measurement of a narrowband SAW filter as DUT with artificially increased electrical feedthrough from input to output.

2. To transform and view this data into time domain, select the main menu item "Settings"->"Time Domain Settings"->"Trace 2":



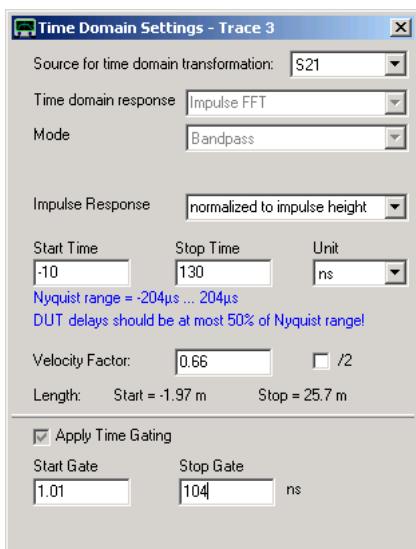
Here you can specify the signal source which is to be transformed to the time domain (here S21) and also the type and parameters of the transformation to be used. Start and stop times for the time domain diagram can be entered.

3. Close the window when done and do select on the main window a second trace with trace type "Time Domain". You can add **time markers the same way as adding frequency markers. Another way to change the start and stop times is to doubleclick the red **time labels** below the traces of the main window.**

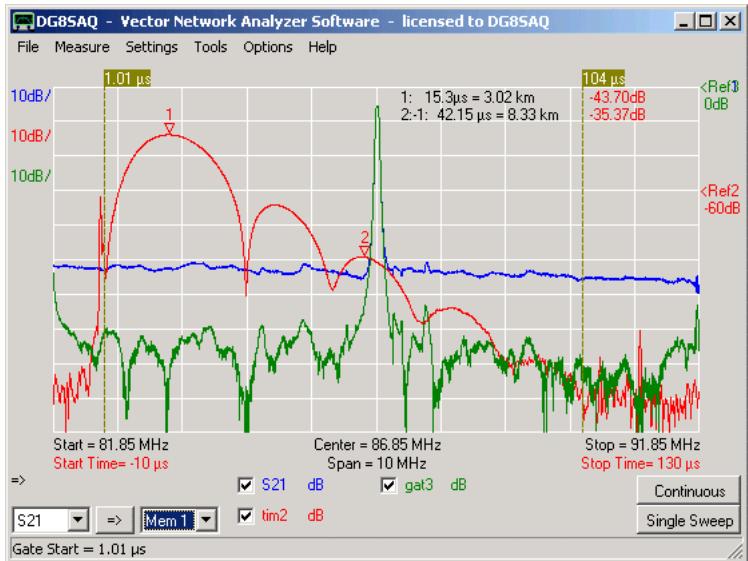


You do now see the inverse Fourier transform of your frequency data (red trace), which is by nature of the discrete Fourier Transform periodic. The part of it starting at time zero is the envelope of the DUT's **impulse response**, which you could also measure by applying a short impulse to the DUT's input. Note, that due to the DUT's high Q-value, it delays and stretches an incoming impulse considerably (filter ringing).

4. Add a third trace to the main window with trace type "**Frequency-Domain Gated**" or **Gated** and then select the main menu item "**Settings**"-"**Time Domain Settings**"-"**Trace 3**"::



5. Once you have activated **time gating** you will see two dashed vertical lines on the main window with attached labels indicating the time gate window edges (start gate and stop gate). You can move these with the mouse. Now zoom into the time domain so you see only the time range of interest:



6. Move the vertical dashed **gate lines** (or attached labels) with the mouse and observe how the green trace changes. If you look at the red time domain response, you see a sharp peak near time zero. This is the very fast electrical feedthrough signal travelling at speed of light, i.e. the fraction of the input impulse that bypasses the SAW filter due to electrical crosstalk and remains unchanged. The round features at later times is the SAW filter impulse response (=filter ringing), which travels by the nature of the Surface Acoustic Wave (SAW) with the speed of sound on the quartz crystal chip (about 1000m/sec). Due to this speed difference it is possible to zero out the electrical feedthrough in the time domain (=gating) and visualize the transfer function of the filter without electrical feedthrough in the frequency domain (green curve). The green curve shows stopband features which are hidden under the electrical crosstalk in the blue trace.

►►► **Hint:** You can change the color of the start gate and stop gate lines and labels by right-clicking them.

Summary:

This technique called "time domain gating" zeroes out the fractions of the DUT's time domain response outside the vertical gate boundary lines. Transforming the gated impulse response back to the frequency domain might reveal features previously hidden by e.g. electrical feedthrough.

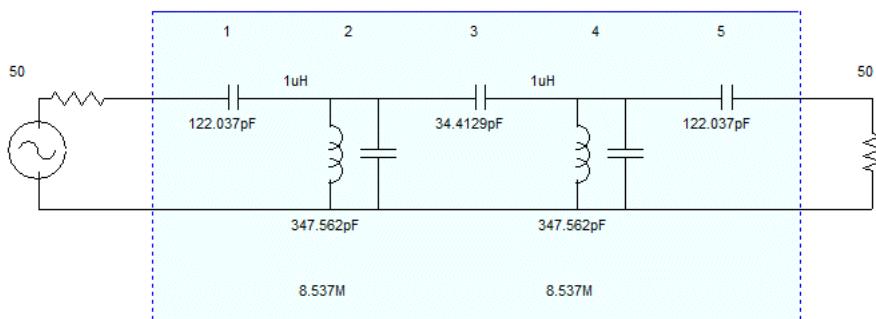
TUNING FILTERS IN TIME DOMAIN

This tutorial was provided by Roderick Wall, VK3YC. Thank you very much, Roderick!

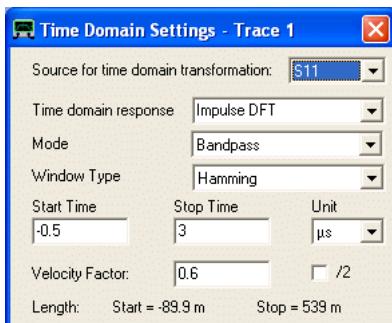
Note, that from software version VNWA35.9 on there are two new time domain settings referring to impulse response normalization and low frequency data extrapolation. Use the default settings (impulse response normalized to impulse height and frequency data below 25 kHz extrapolated) to reproduce below results.

Using Time Domain to tune variable coupling BPFs

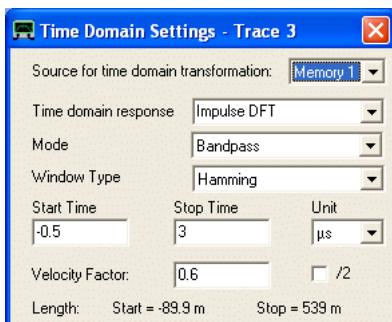
A Template is required to tune variable coupling filters. The Template can be generated from a Gold Standard Engineering filter or simulated in software. This example uses Jim Tonne's Elsie Filter design software to design a BPF and to generate a s2p Touchstone s-parameter file. The s-parameter file is imported into a memory location in VNWA to be used as a Template to tune the filter to. The example 40 meter variable coupling BPF circuit is shown below. Trim capacitors were used for the coupling capacitors. Adjustable inductors were used for the inductors.



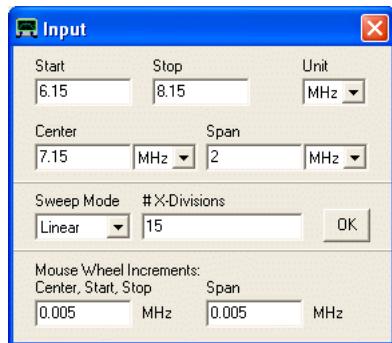
Use Elsie to design the filter. Set the centre frequency to 7.15MHz, frequency span to 2MHz and sweep steps to 500 (also refer to VNWA settings below). Generate a s2p s-parameter Template file. Import the the s2p Template file into VNWA and store S11 into Memory 1. Store S21 into Memory 2. Under "Settings" select "Time Domain" and then select "Trace 1". Set the Time Domain settings as shown below.



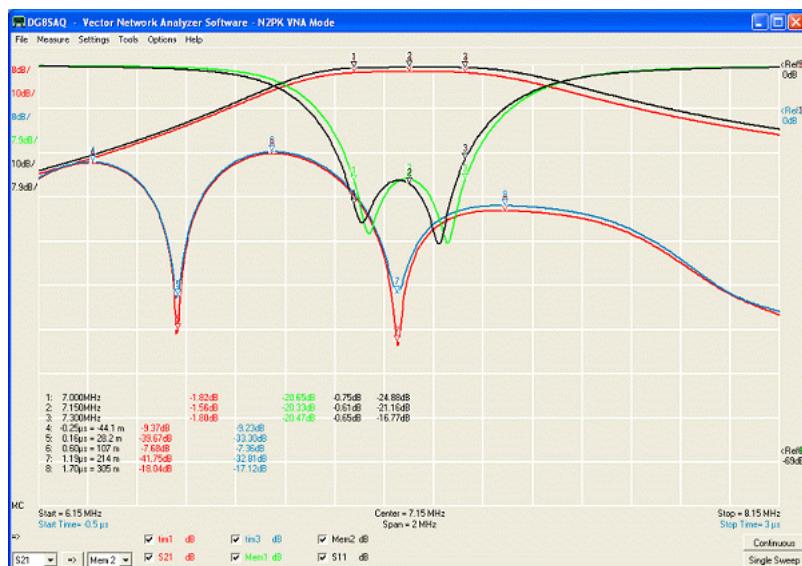
Under "Settings" select "Time Domain" and then select "Trace 3". Set the Time Domain settings as shown below.



Set the center frequency to the resonant frequency that the resonators need to be tuned to. If you don't then you will be tuning the BPF to the wrong frequency. Set the Span to two to five times the bandwidth of the BPF. Set the Mouse Wheel Increments to 5kHz.



Set the number of scan steps to 500. As shown below, set trace 2 to S21 dB, trace 4 to Mem1 dB, trace 5 to Mem2 dB and trace 6 to S11 dB.



Blue Time Domain trace = Elsie Template. Red Time Domain trace = DUT BPF.
 Green Return Loss trace = Elsie Template. Black Return Loss trace = DUT BPF.
 Black Insertion Loss = Elsie Template. Red Insertion Loss = DUT BPF.

Click "Single Sweep" to scan the DUT 40 meter BPF. Adjust trace scaling to suitable sizes. Your Time Domain trace won't have the deep dips shown above because the BPF has been adjusted using Time Domain. The two Time Domain dips are the return loss from each resonator. The three humps are the three coupling capacitors. Refer to Agilent's application notes AN 1287-8 and AN 1287-10 for the procedure to adjust the filter. Click Continuous scan to adjust the BPF.

Before assembling the BPF you may want to adjust the components to the values shown in the circuit, this will make it easier to adjust the BPF.

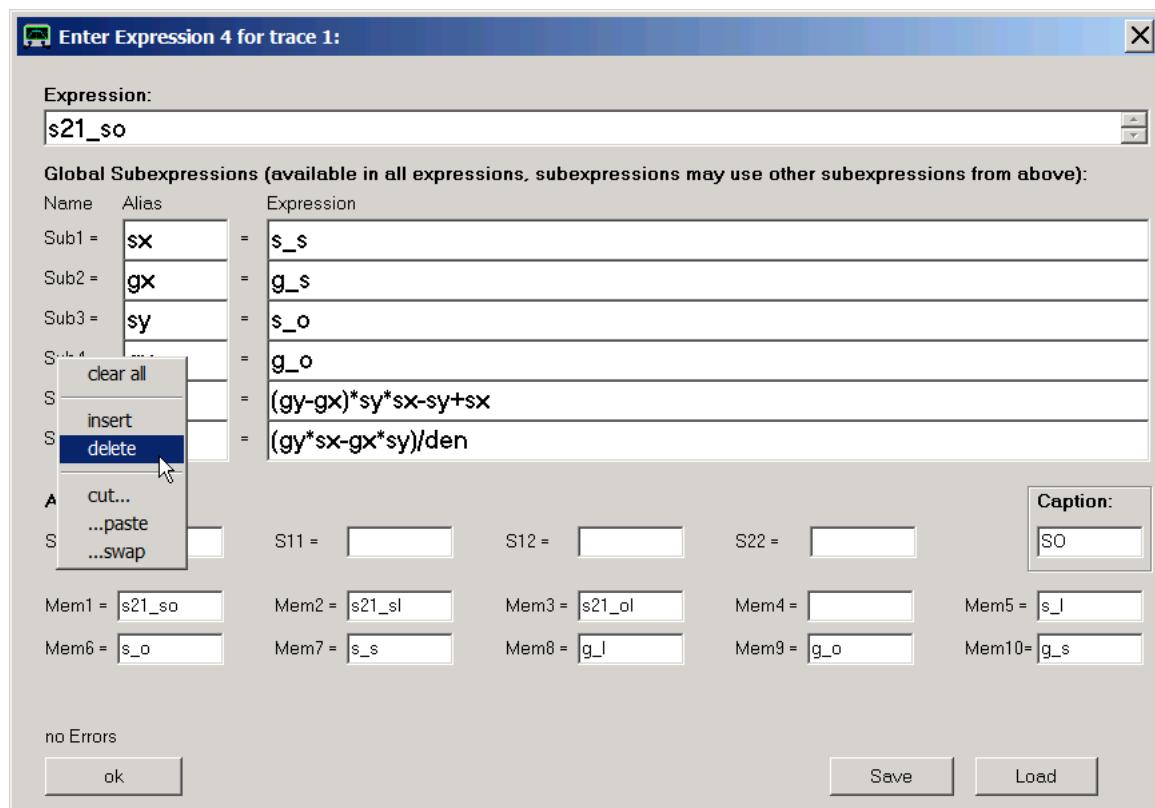
The Mouse wheel can be used to adjust the center frequency. Hold the cursor over the center frequency and turn the wheel. Adjusting the center frequency allows you to determine the frequency the resonators are tuned to. Adjust the center frequency for the deepest dip and the center frequency is the resonant frequency. Using the Mouse Frequency wheel makes it easier to adjust the resonators.

Roderick Wall, vk3yc.

CUSTOM: TRACE DATA MANIPULATION

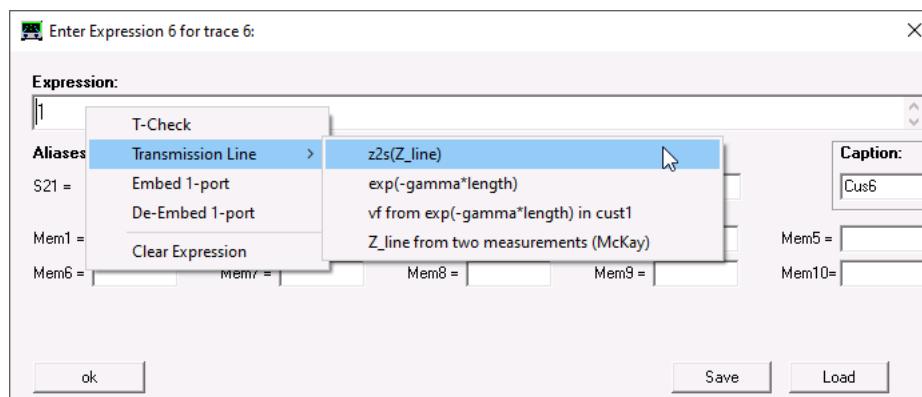
The **trace type custom** allows to simulate traces or to manipulate trace data in order to display functions of trace data.

VNWA.exe contains a **compiler**, which can generate fast code to calculate **user defined mathematical complex algebra expressions**. This code can be used for real time data manipulation during the sweep or to generate simulation data.



Note the right-click menu of the sub*-labels in above screenshot, which allows convenient expression manipulations.

For convenience, there are some pre-defined expressions that can be loaded via **right-click menu of the Expression input field**:



Currently pre-defined expressions:

T-Check as explained in chapter T-Check.

Transmission Line equations as used by the software Zplots. For more details see chapter Interfacing ZPlots.

The model details are explained in detail in
https://www.dg8saq.darc.de/HamRadio/HamRadio_DG8SAQ_2018_English.pdf.
A demonstration on how to use these transmission line expressions is shown below in Example 6.

Embed/De-Embed equations allow to "add" or "remove" a user defined 2 port described by its S-matrix to/from a 1 port measurement.

A detailed explanation and an example are given in chapter Magi-Cal as Permanent Calibrator.

►►► **Note:** This parser / compiler (own development) is different from that used in complex calculator, thus functionalities might differ slightly.

Example 1: Determining an impedance from a transmission measurement

Example 2: Unitarity relation

Example 3: Subexpressions

Example 4: Displaying the Phase of an Impedance

Example 5: Calculating various properties of an inductor or capacitor measured in transmission

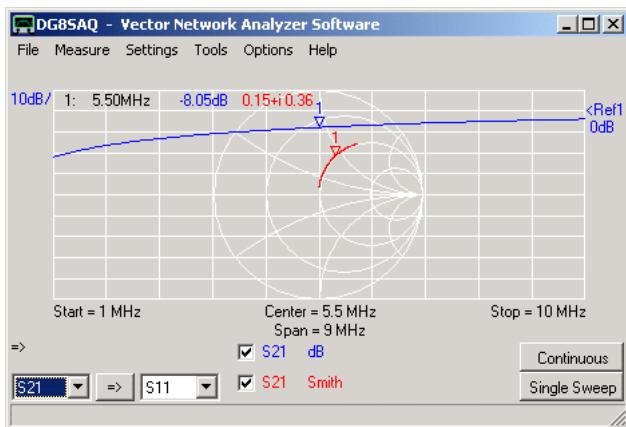
Example 6: Calculating various properties of a transmission line

Syntax Reference

Example 1: Determining an impedance from a transmission measurement

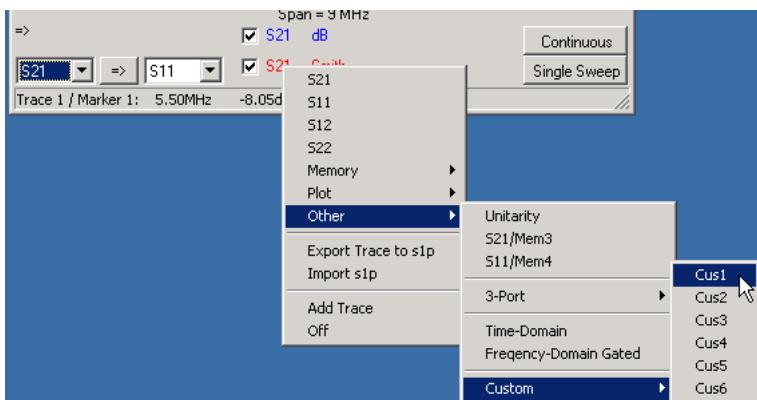
Is it possible to accurately determine an impedance from a transmission measurement with only a thru calibration?
The answer is yes and a custom trace greatly simplifies this task.

Below is a transmission measurement (S21) of a 120pF capacitor connected between the hot pins of the VNWA TX and RX port after a simple thru calibration (note, that both traces show S21):

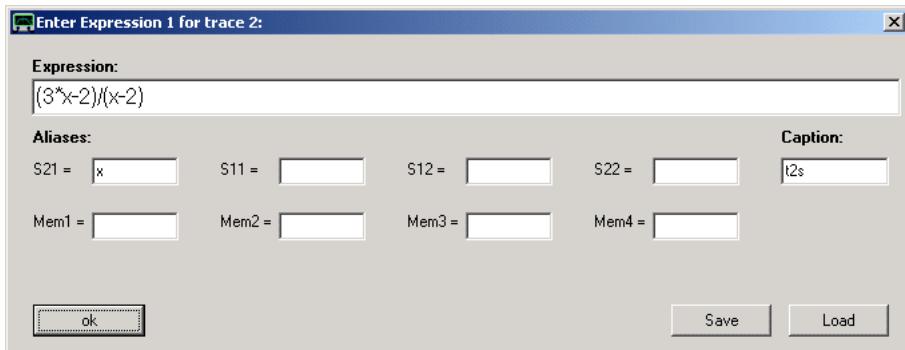


Clearly, the higher the frequency the lower the insertion loss due to the capacitor's decreasing impedance with increasing frequency.

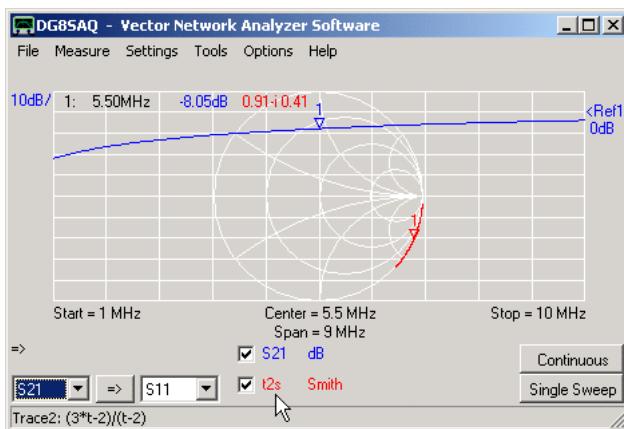
Next, we want to calculate the one port reflection coefficient (S11) of the capacitor from our transmission measurement with the aid of a custom trace. Therefore we right-click the S21 label and select trace type "Other-Custom-Cust1":



The custom trace editor window opens indicating in its header that we are editing expression 1 (=cus1) which is to be displayed in trace 2:

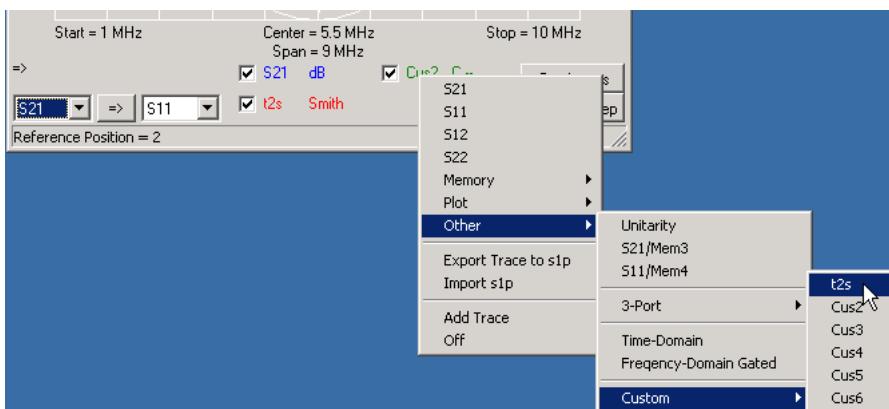


For simplicity, we assign to S21 the shorter **alias name x** and enter the expression $(3*x-2)/(x-2)$, which transforms a thru-measurement result into a reflect measurement result (Note that from VNWA version 35.9.d on this expression is implemented as function t2s()). Also, we give this expression the descriptive name **t2s** (for transmission to scattering parameter, default name is cus1), see **caption field**. Press ok and observe the updated main window:

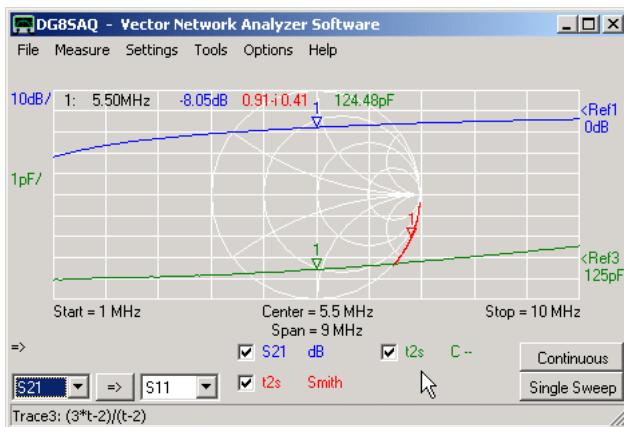


Red trace 2 is now labeled by the assigned name t2s and the data apparently shows the reflection coefficient of a quite ideal capacitor (curve running along the lower edge of the Smith chart).

We now want to determine the capacitance of the capacitor, therefore, we want to see the very same t2s data in a third trace in terms of Picofarads. We add a third trace and select again trace type "**Other-Custom**":

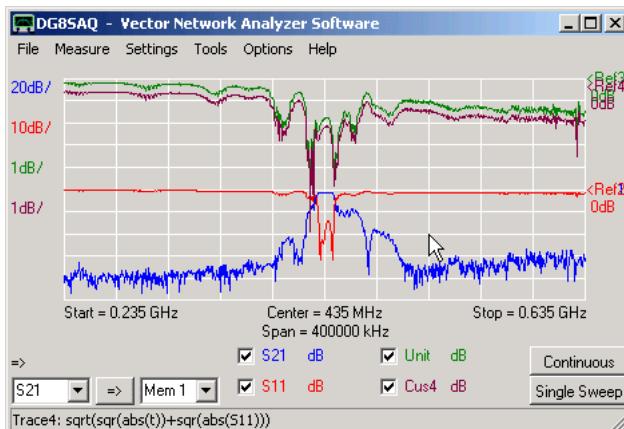


Observe, that the custom drop down menu now shows as first entry **t2s** instead of cus1 because we have assigned this name to expression 1. We select t2s. The custom trace editor opens again, but we do not need to edit anything as we have already entered the required expression before. Therefore we simply close the custom trace editor and select display type "**C--**" for trace 3:



Indeed, our capacitor shows a capacitance of 124pF, which is very close to the nominal capacitance of 120pF printed on it. Note, that the rising capacitance is due to the inductance of the long capacitor leads.

Example 2: Unitarity relation

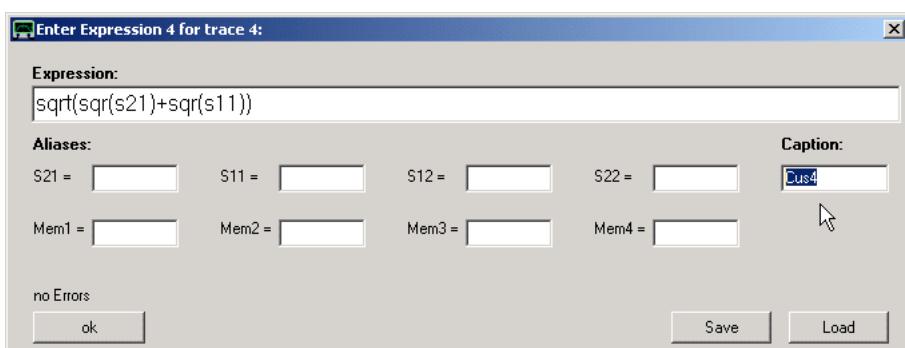


The above example shows measured S11 and S21 data of a SAW filter (red and blue traces). The green trace shows the so called unitarity condition:

$$\text{Unitarity} = \sqrt{|S_{11}|^2 + |S_{21}|^2}$$

which, when shown in dB, is the power neither reflected nor transmitted, thus the power dissipated inside the DUT.

The brown "Cus4" trace (= custom trace 4) shows exactly the same as the "Unit" trace, but is user defined.
Double-clicking the Cus4 label will open the custom trace editor window:

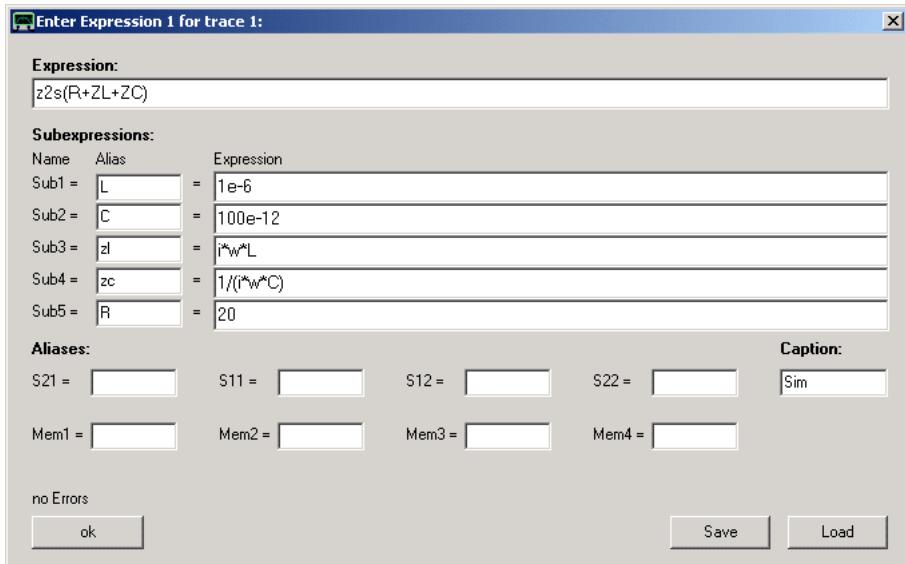


Here, the mathematical **expression** to be displayed can be entered. It is evaluated using complex calculus, thus complex S-parameters can be manipulated in a proper way. In the above example, the unitarity formula has been entered.

- ▶▶ **Note:** Alias variables can be defined to abbreviate e.g. the rather lengthy variable name S21 by the variable t.
- ▶▶ **Hint:** Typing return upon completion of the expression will invoke a **syntax check** without closing the window.
- ▶▶ **Hint:** Pressing the "ok button" will also invoke a syntax check and close the window only if the syntax is correct.
- ▶▶ **Hint:** You can **save and load expressions** to or from a Derive *.mth file.
- ▶▶ **Hint:** Expressions will automatically be saved in the VNWA ini-file upon program termination for reuse after program restart.
- ▶▶ **Hint:** By changing the **Caption** field in above window (see mouse arrow), you can modify the trace identifier label K on the main VNWA window showing up below the grid.

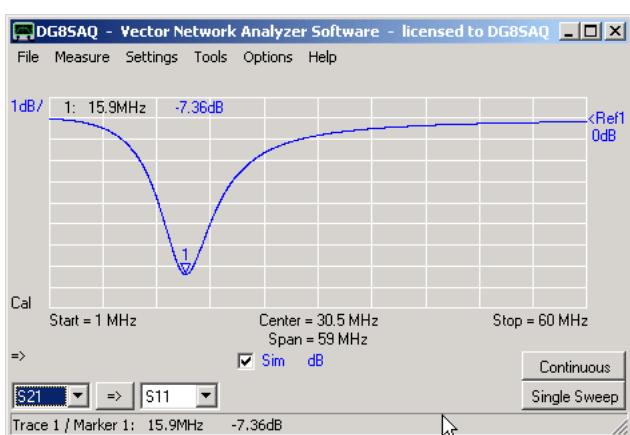
Example 3: Subexpressions

When the custom trace editor window height is increased, up to 10 input fields for **subexpressions** are being revealed:



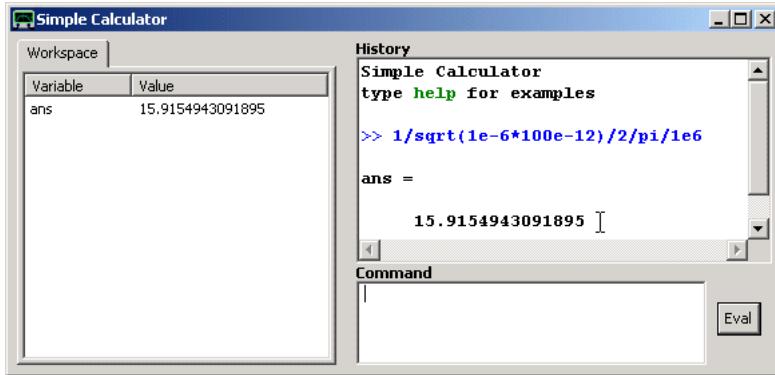
Subexpressions can be used if a parameter or term is to be reused several times in the same custom expression or simply to improve readability. In the above example, the reflection coefficient of a series resonant circuit is being calculated using subexpressions. In subexpression 1 an inductance value of 1 microhenry is being defined. Subexpression 2 defines a capacitance value of 100pF. Subexpressions 3 and 4 define the impedances of the inductor and capacitor respectively by making use of subexpressions 1 and 2. Subexpression 5 defines a resistor value of 20 Ohms.

The main expression sums up the impedances defined in subexpressions 3, 4 and 5 and converts it to a reflection coefficient that can be displayed with the standard S11 display modes on the main window:



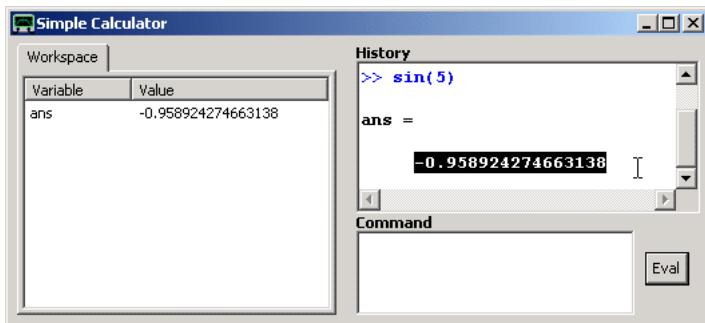
Apparently, we have simulated a series resonant circuit with a resonance frequency of 15,9 MHz. We can use the

Complex Calculator tool to check if the resonance is where it would be expected:



Indeed, the resonance is to be expected at 15.9 MHz.

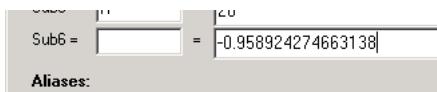
- ▶▶▶ **Note:** All 6 custom traces share the same subexpressions.
- ▶▶▶ **Note:** Subexpressions can be accessed either by the names **Sub1...Sub6** or by the user defined aliases.
- ▶▶▶ **Note:** Subexpressions may make use of previous subexpressions but not of subsequent subexpressions. Circular references are forbidden.
- ▶▶▶ **Note:** Custom traces are only being updated when the return key is pressed after editing any of the subexpressions or expression, or if the ok button is pressed.
- ▶▶▶ **Note:** You can **transfer precalculated values** from the Complex Calculator tool to one of the subexpressions by marking the result in complex calculator with the mouse ...



..., then pressing **ctrl c** to copy the marked data into the clipboard, then right-clicking into the field where the value is to be transferred to...

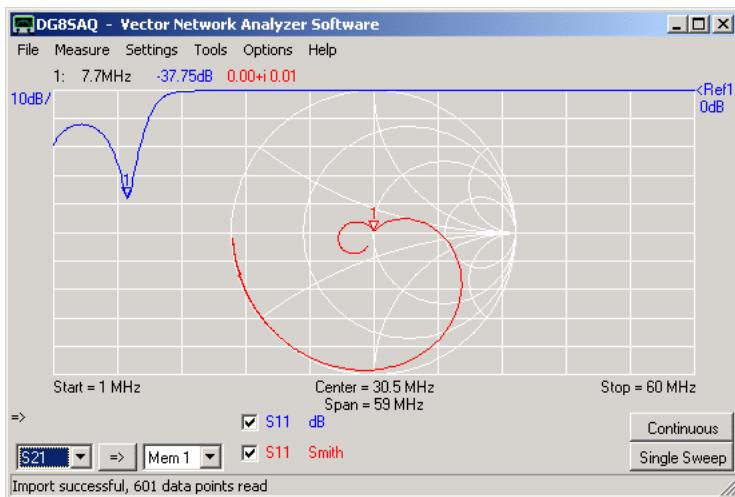


... and finally selecting **paste**:

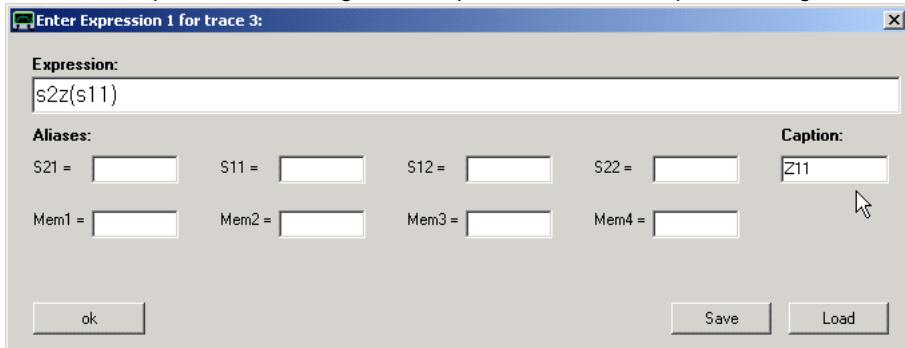


Example 4: Displaying the Phase of an Impedance

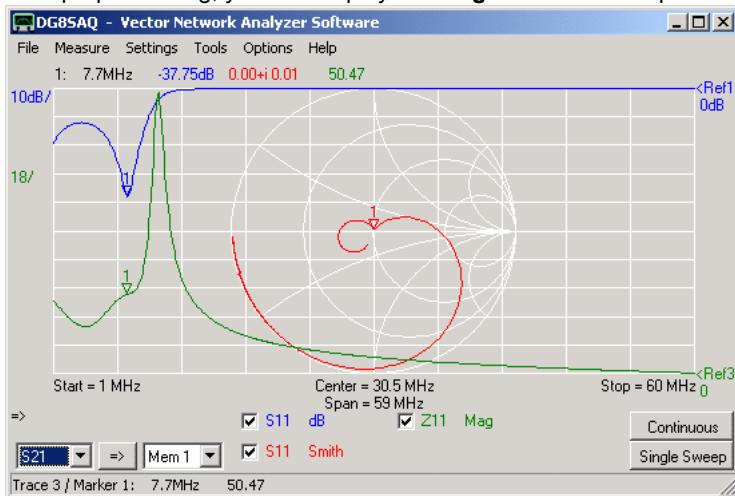
When measuring impedances, VNWA internally works with reflection coefficients and by default displays those:



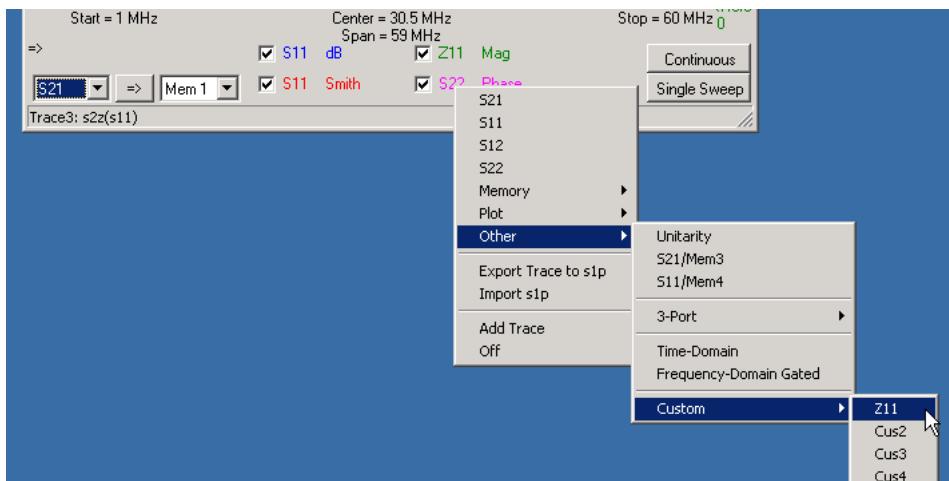
Now, it shall be demonstrated how the magnitude and phase of the corresponding impedance can be displayed using a custom trace. Add a custom trace (e.g. cust1), fill in the expression "s2z(S11)" which converts the reflection coefficient s11 into an impedance and assign a descriptive name to this expression, e.g. "Z11":



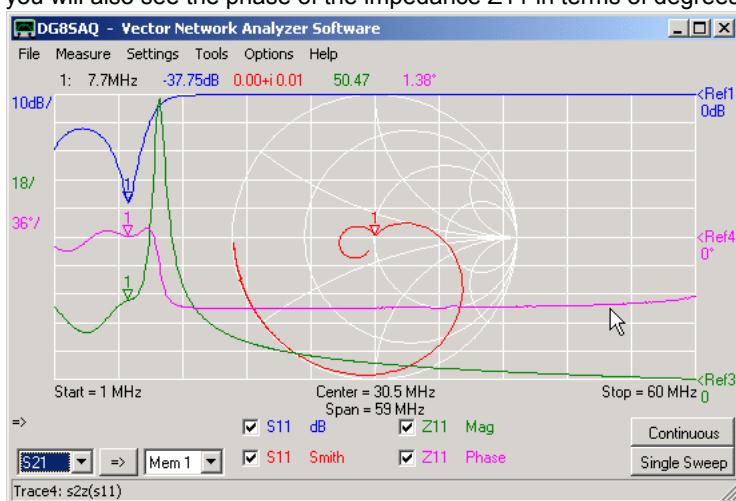
After proper scaling, you can display the **magnitude** of the impedance (green). The unit will be Ohms:



If you want to display the **phase** of the impedance as well, simply add another trace and select to display custom trace Z11 which we have defined before by assigning the name Z11 to it:

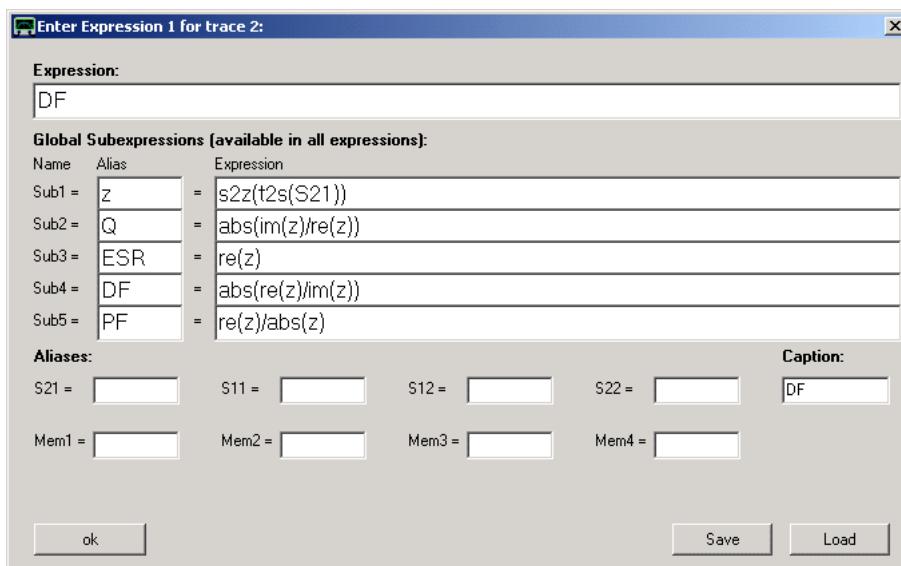


Immediately close the custom trace window again which is popping up as no changes are required. After proper scaling, you will also see the phase of the impedance $Z11$ in terms of degrees (pink):



Note, that we are using just one and the same custom expression for two different traces.

Example 5: Calculating various properties of an inductor or capacitor measured in transmission



Subexpression 1 contains the conversion from transmission measurement (S21) to reflection data (function t2s) and

subsequent conversion to impedance z (function $s2z$).

Subexpression 2 contains the Q-factor.

Subexpression 3 contains the "Equivalent Series Resistance" formula.

Subexpression 4 contains the dissipation factor formula.

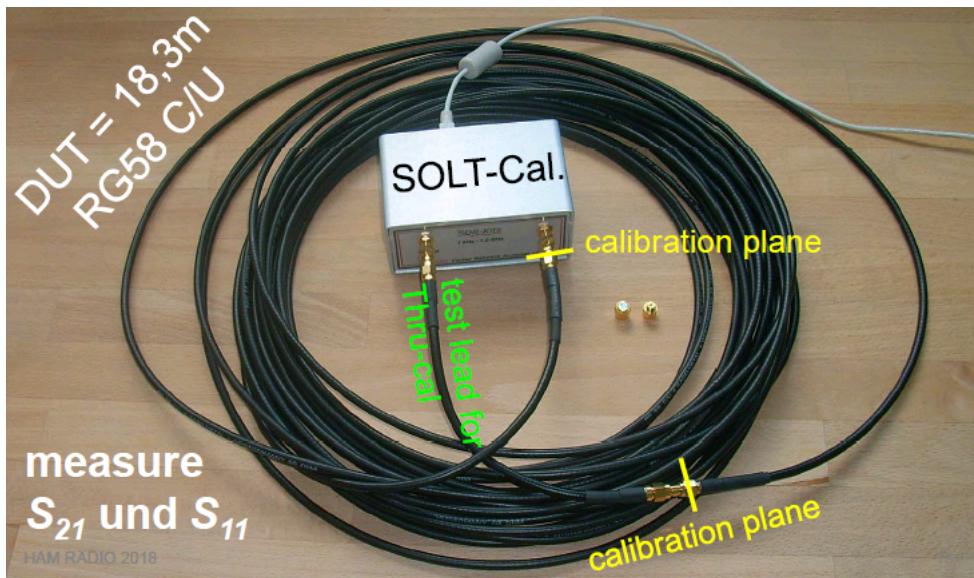
Subexpression 5 contains the power factor.

The main expression can use any of the subexpression aliases, e.g. the dissipation factor DF.

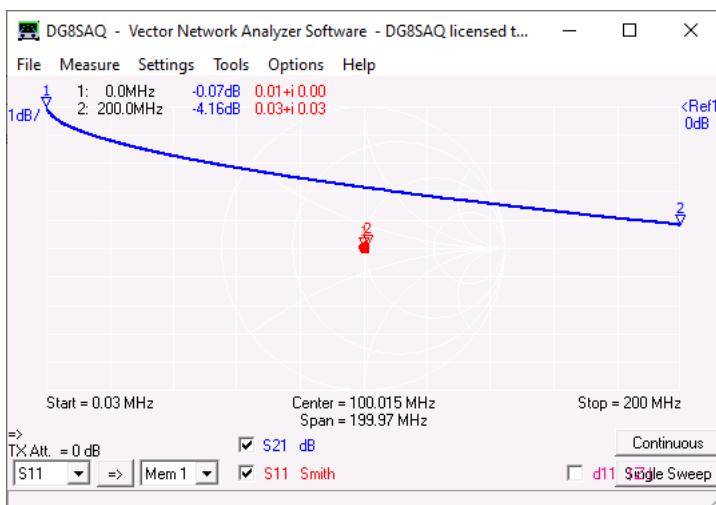
Example 6: Calculating various properties of a transmission line

This example demonstrates how to use the pre-defined transmission line equations to obtain transmission line parameters from a 2-port measurement of a piece of transmission line.

Starting point is a measurement of a piece of 18.3 meters of RG58C/U cable:



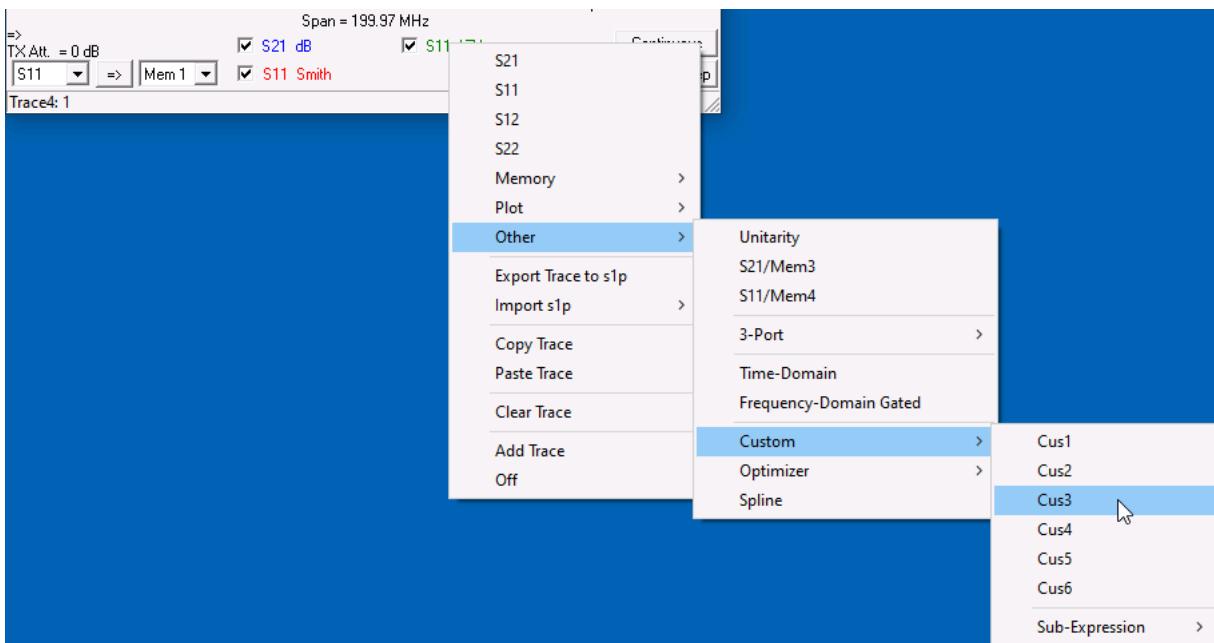
Result:



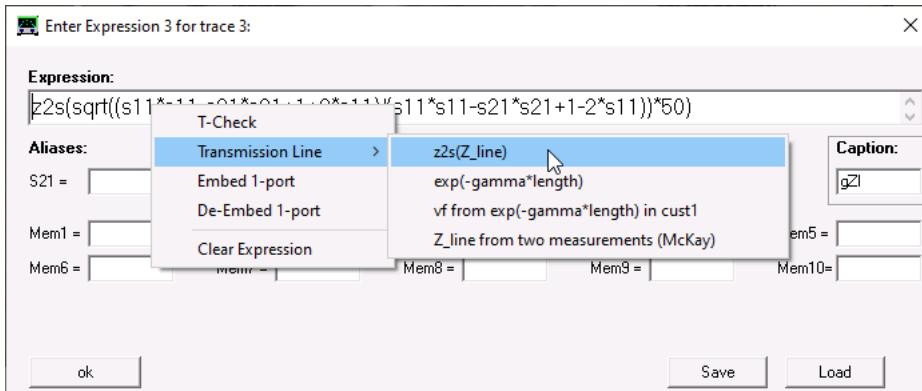
Apparently, this is a 50 Ohms transmission line. Since we measure with a 50 Ohms instrument, S21 shows us the matched line loss.

Using the transmission line equations, we can actually calculate the characteristic frequency dependent line impedance from our measurement.

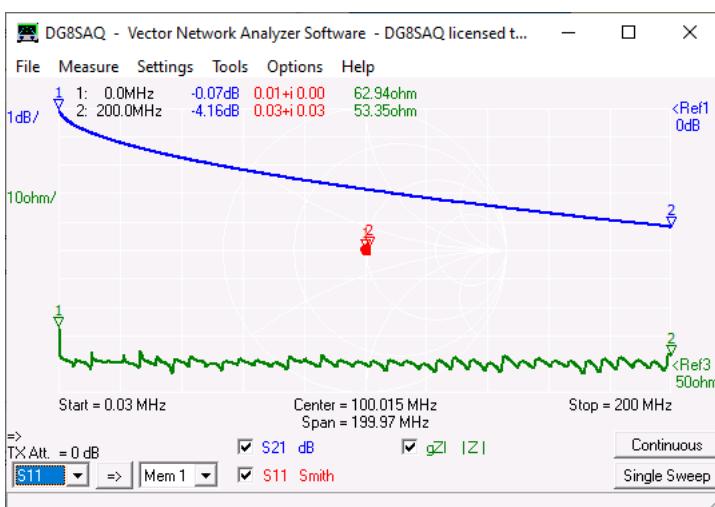
We add custom trace 3 (because we will still need custom traces 1 and 2)...



and select the pre-defined expression $z2s(Z_line)$:



As a result, we can display the frequency-dependent characteristic impedance of our cable, which is indeed close to 50 Ohms:

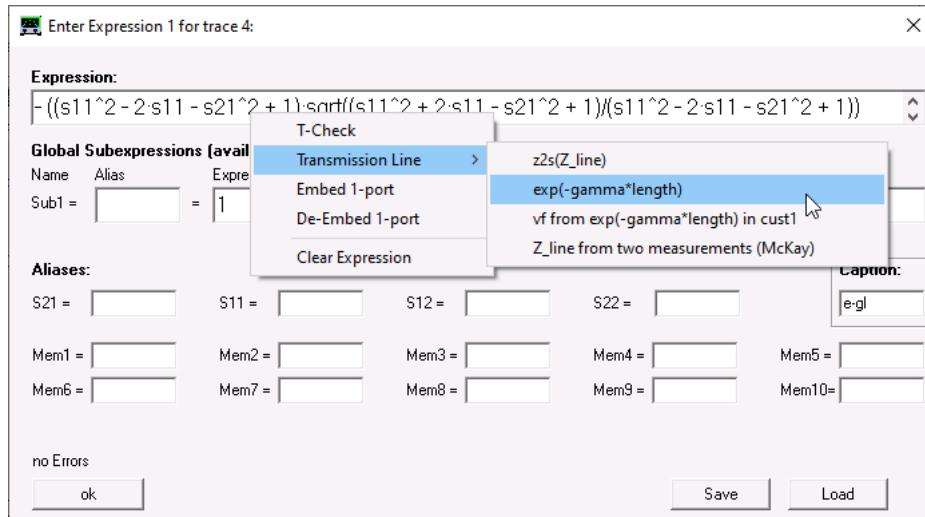


Note the considerable increase of cable impedance at very low frequencies due to dispersion. The oscillations in line impedance are caused by cable and measurement imperfections and can never be avoided.

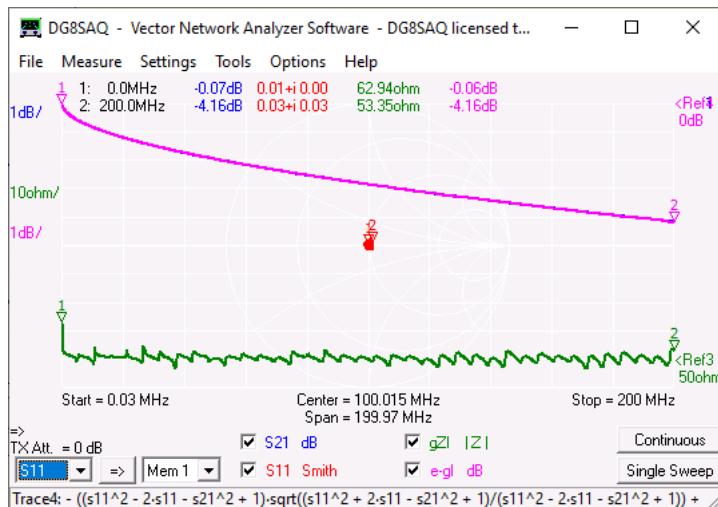
Next, we want to find out the cable velocity factor vf . This can be calculated from the cable propagation constant γ , which in turn can be calculated from our measurement.

So, we add another custom trace using custom expression 1 (cus1, later called e-gl trace). We must use cus1 for this as

the equation calculating vf uses the result of expression $cus1$.
We select the pre-defined equation $\exp(-\gamma \cdot length)$...



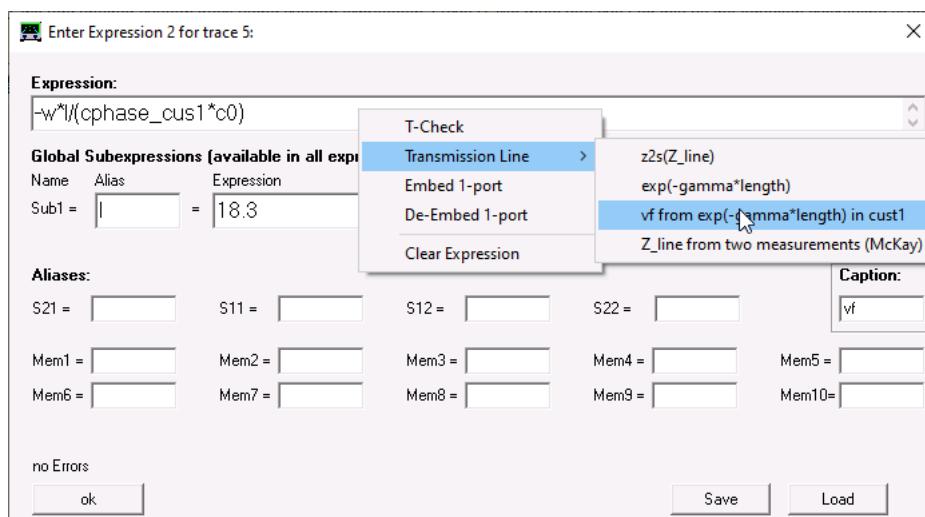
and obtain:



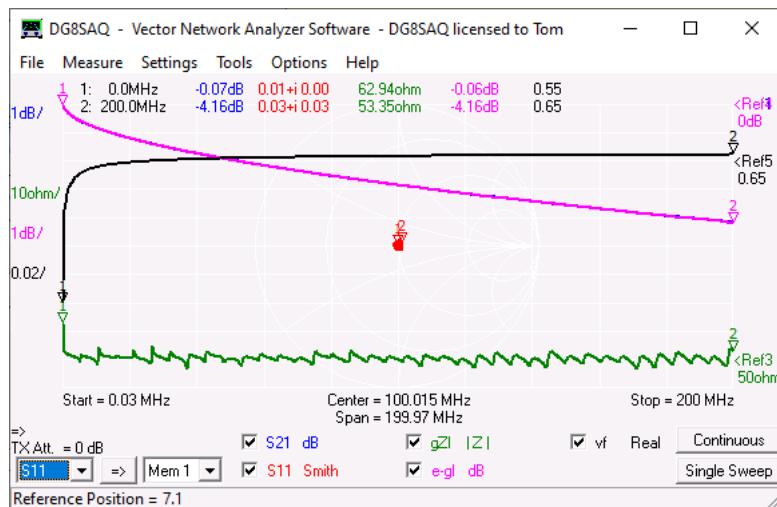
Note, that the pink $e\text{-}gl$ custom trace is identical with the $S21$ trace in this case, because our cable impedance is identical with the measurement system impedance of 50 Ohms.

If the cable was e.g. a 75 Ohms cable, $S21$ would show additional loss by reflections, while the $e\text{-}gl$ trace would still show the matched cable loss as measured in a 75 Ohms environment.

The velocity factor can now be calculated from the phase of the $e\text{-}gl$ trace by adding another custom trace and loading pre-defined equation "vf from ...".



Note, that in sub-expression 1 (Sub1) we have to define the mechanical cable length.
 Above expression yields the frequency dependent velocity factor:



Note, that vf drops considerably at low frequencies due to cable dispersion.

►► Note:

The pre-defined equations *Z_line from two measurements (McKay)* are described in McKay's paper "Measure the characteristic impedance ...", ELECTRONIC DESIGN 24, Nov 22, 1976.

They use two cable reflection measurements with two distinct terminations to determine the characteristic line impedance.

Available Syntax

Available special constants

pi = 3.14...
e = Eulers number 2.718...
i = complex unit (mathematician's choice)
j = **i** = complex unit (engineer's choice)
gamma = Euler-Mascheroni-constant 0.57721...
clock = DDS input clock [Hz]
mil = 1 imperial MIL=1 inch/1000 in meters
inch = 1 imperial inch in meters
foot = 1 imperial foot in meters
yard = 1 imperial yard in meters
mile = 1 imperial mile in meters
c0 = vacuum speed of light in meters per second
kb = Boltzmann constant in Joules per Kelvin
neper = $\ln(10)/20$, conversion Neper to dB
sa = spectrum analyzer mode offset level in linear scale

Available operators

+ = complex addition
- = complex subtraction
***** = complex multiplication
/ = complex division
^ = complex power *Hint: exp(x) is more CPU efficient than e^x*

Available data functions:

f = frequency
w = $2\pi f$
s = complex frequency = $i w$
t = time as shown in the time domain displays

S21 = measured data S21
S11 = measured data S11
S12 = measured data S12
S22 = measured data S22
Mem1 = data space Mem1 data
Mem2 = data space Mem2 data
...
Mem10 = data space Mem10 data
P1 or Plot1 = data space Plot1 data
...
P4 or Plot4 = data space Plot4 data
s_11...s_33 = 3-port measured data
gated = gated response
a_ = reflect calibration coefficient *a*
b_ = reflect calibration coefficient *b*
c_ = reflect calibration coefficient *c*
M21 = raw, uncorrected measured data for S21
M11 = raw, uncorrected measured data for S11
M12 = raw, uncorrected measured data for S12
M22 = raw, uncorrected measured data for S22
SS = VNWA source reflection coefficient as calculated from calibration data
SL = VNWA load reflection coefficient as calculated from calibration data
n = data point number
n_top = highest data point number = # data points - 1
noise = random complex numbers with real and imaginary parts between -1 and +1
delay_s21 = group delay of S21 = -d Arg(S21(w))/d w
delay_s11 = group delay of S11 = -d Arg(S11(w))/d w
delay_s12 = group delay of S12 = -d Arg(S12(w))/d w
delay_s22 = group delay of S22 = -d Arg(S22(w))/d w
delay_mem1 = group delay of Mem1 = -d Arg(Mem1(w))/d w
...
delay_mem4 = group delay of Mem4 = -d Arg(Mem4(w))/d w
cphase_s21 = continuous phase of S21 in radians
cphase_s11 = continuous phase of S11 in radians
cphase_s12 = continuous phase of S12 in radians
cphase_s22 = continuous phase of S22 in radians
cphase_mem1 = continuous phase of Mem1 in radians
...
cphase_mem4 = continuous phase of Mem4 in radians
cphase_cus1 = continuous phase of Mem1 in radians
...
cphase_cus6 = continuous phase of Mem6 in radians
Sub1...Sub6 = subexpression 1...6
x1...x10 = optimizer variables from Optimizer Tool
opt = optimizer expression from Optimizer Tool
os1...os20 = optimizer sub-expressions from Optimizer Tool
s_dm = calculates reflection coefficient of differential impedance between port 1 and 2 of two-port from S11, S21, S12, S22
s_dm_sym = calculates reflection coefficient of differential impedance between port 1 and 2 of two-port from S11 and S21 assuming S22=S11 and S12=S21
s_open = reflection coefficient of open calibration standard
s_short = reflection coefficient of short calibration standard
s_load = reflection coefficient of load calibration standard
s_thru = transmission coefficient of thru calibration standard
s_thrumatch = reflection coefficient of thru calibration standard

Available complex mathematical functions:

EXP = complex exponential function
IM example: im(x+j*y)=j*y
IMAG=imaginary part (standard definition): imag(x+j*y)=y
RE =real part
SIN =complex sine
COS =complex cosine
TAN =complex tangent
ATAN =complex inverse tangent
ABS =complex absolute value or magnitude
ABSSQR =complex square of the absolute value
SQR =complex square

SQRT =complex squareroot
CONJ =complex conjugate
DEG =conversion radians => degrees: deg(pi)=180
RAD =conversion degrees => radians: rad(180)=pi
ARG =argument of complex number in radians: arg(exp(j*x))=x
LN =complex natural logarithm
LG =complex logarithm of base 10
LOG =complex logarithm of base 10
J0 =complex Bessel function of first kind of order 0
J1 =complex Bessel function of first kind of order 1
SI =Integral Sine function
CI =Integral Cosine function
SINH =complex hyperbolic sine
COSH =complex hyperbolic cosine
HEAVISIDE=real heaviside function, imaginary part of argument is ignored

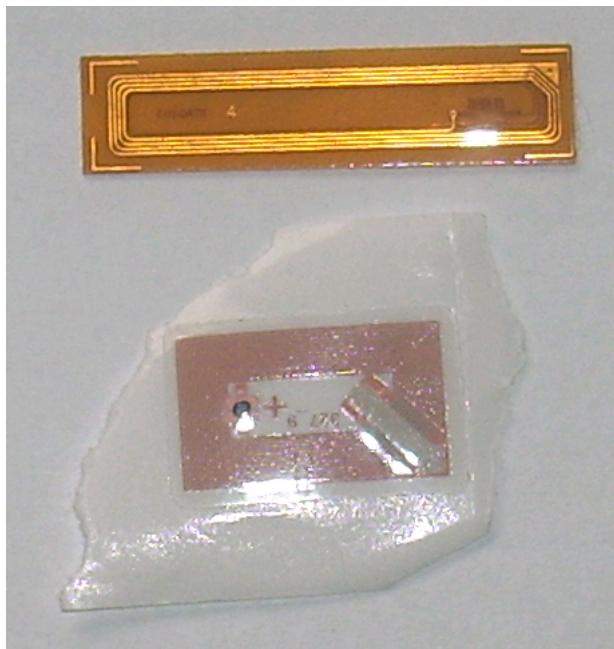
Available special functions of S-parameters:

DB =decibels: db(x)=20*lg(abs(x))
Y = S2Y =complex admittance in 1/Ohms $Y(s)=(1-s)/((1+s)^*50)$
Z = S2Z =complex impedance in Ohms $Z(s)=1/Y(s)$
Y2S =conversion from complex admittance Y(s) to reflection coefficient s, e.g. $s=Y2S(Y(s))$
Z2S =conversion from complex impedance Z(s) to reflection coefficient s, e.g. $s=Z2S(Z(s))$
T2S =convert a transmission measurement (S21) of a one-port device to reflection data (S11)
S2T =convert a reflection measurement of a one port device to transmission data
T2SS =convert transmission data of a one port device to shunt reflection data
SS2T =convert shunt reflection data to transmission data of a one port device
T2ZS =convert transmission data of a one port device to complex shunt impedance data
ZS2T =convert complex shunt impedance data to transmission data of a one port device
CP =parallel capacitive part of complex admittance in F (as function of reflection coefficient S)
LP =parallel inductive part of complex admittance in H (as function of reflection coefficient S)
RP =parallel resistive part of complex admittance in Ohms (as function of reflection coefficient S)
QC =Q-value of complex admittance
CS =serial capacitive part of complex impedance in F (as function of reflection coefficient S)
LS =serial inductive part of complex impedance in H (as function of reflection coefficient S)
RS =serial inductive part of complex impedance in Ohms (as function of reflection coefficient S)
QL =Q-value of complex impedance (identical with QC, but calculated from Z instead of Y)

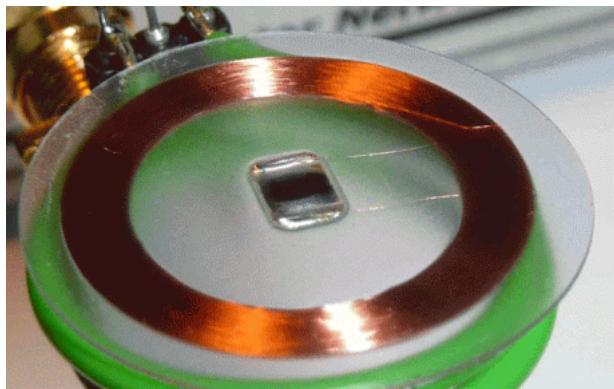
►►► **Note:** The parser is NOT case sensitive!

MEASURING RF-ID TAGS

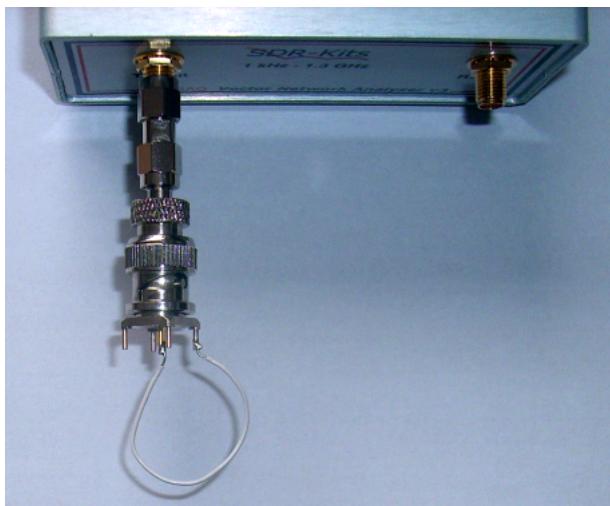
In the following simple method for testing RF-ID tags using the DG8SAQ VNWA by SDR-Kits is demonstrated using 13 MHz NFC RF-ID tags ...



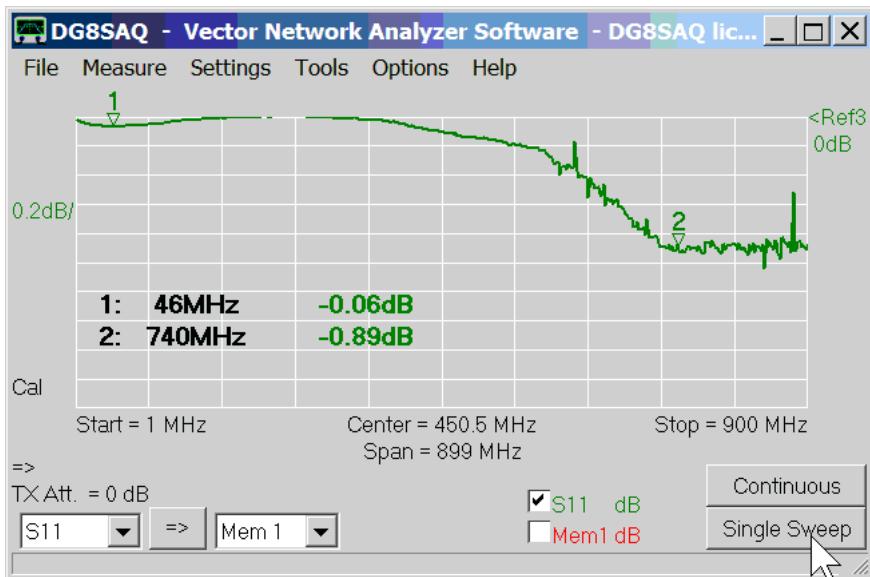
... and 125 kHz ID tags:



The VNWA is operated in reflect mode (measuring S11) using a single turn detector coil with about 2cm diameter connected to its TX port:

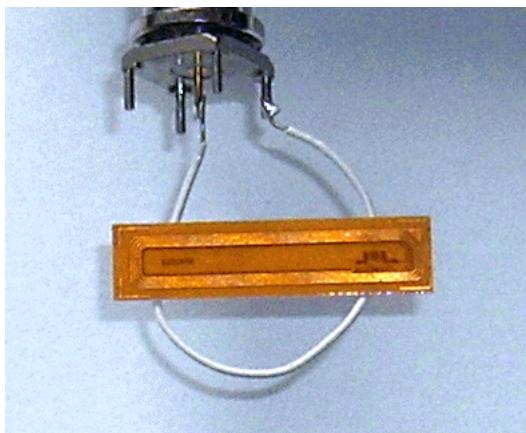


The detector coil shows only very shallow resonances:

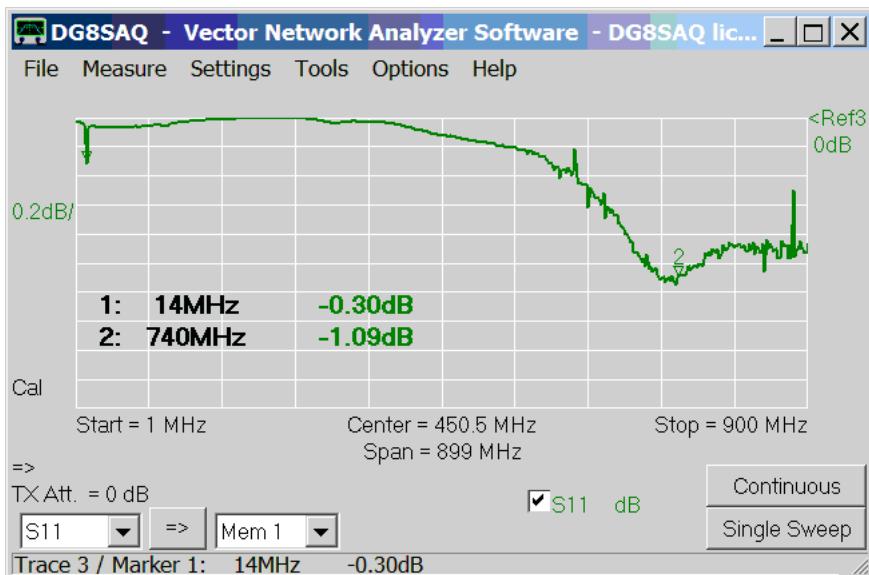


Note, that the coil is damped by the 50 Ohms output impedance of the VNWA TX port, thus it is not likely to detune a resonator by resonant coupling.

Placing an RF-ID tag on top of the detector coil ...

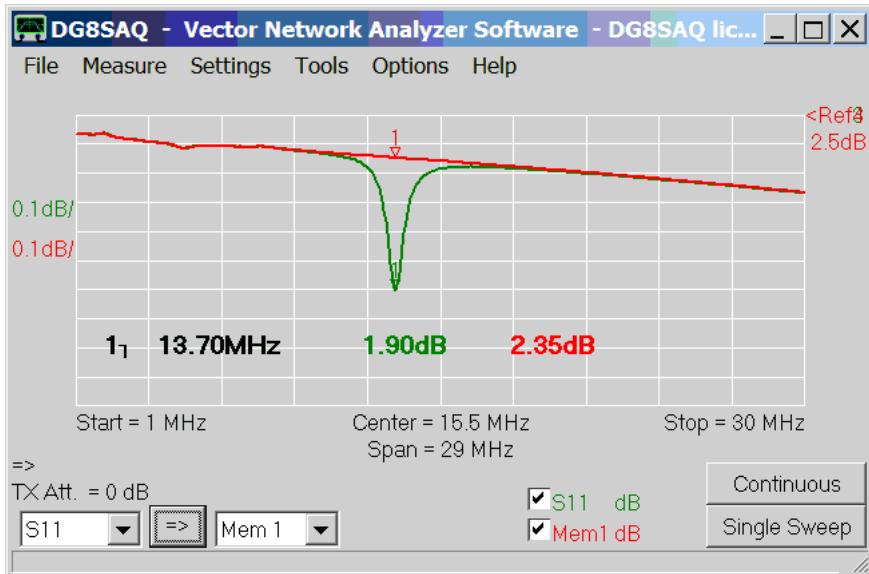


... we can see its effect on the coil showing a sharp but tiny resonance at about 14 MHz:



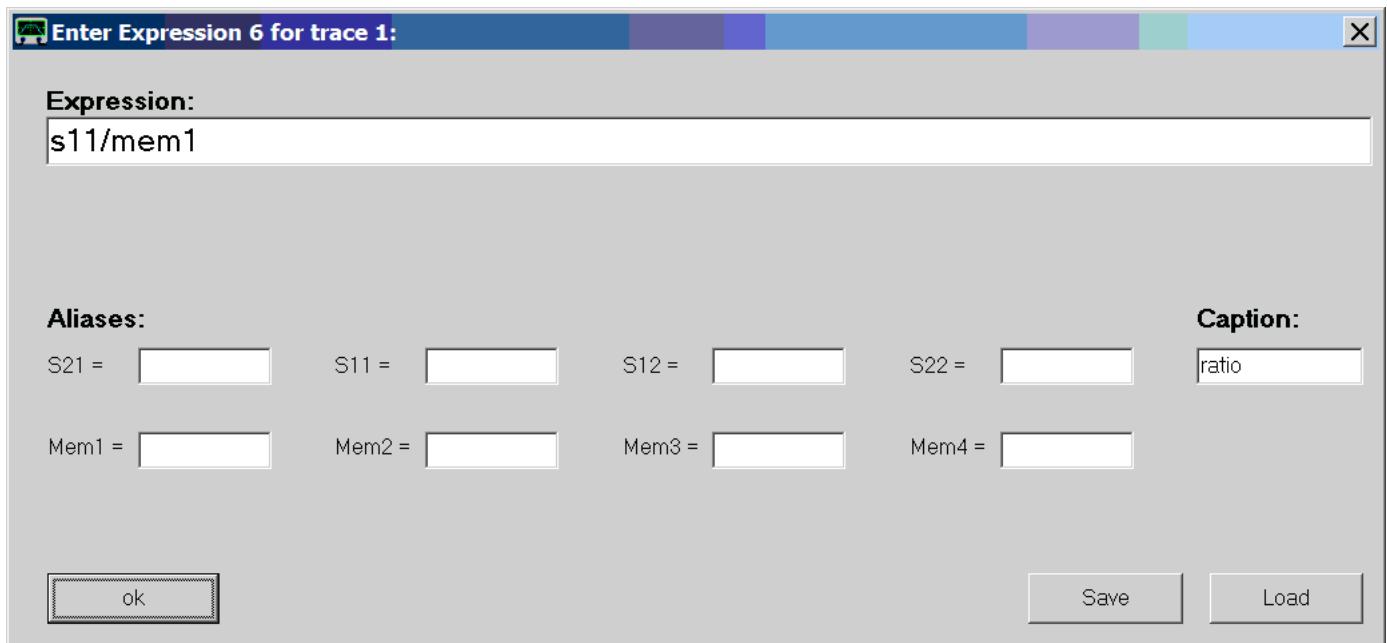
Next, we want to zoom into the tag's resonance frequency and separate the tag's response from that of the coil by a simple calibration step. The idea is to mathematically divide the tag measurement by a measurement of the coil without tag. Using this method, we do not even need to SOL calibrate the TX port. All following measurements will be performed without calibration. The measurement of the coil without tag serves as calibration measurement.

Here are the uncalibrated S11 measurements without (red) and with (green) tag in a narrow frequency range:



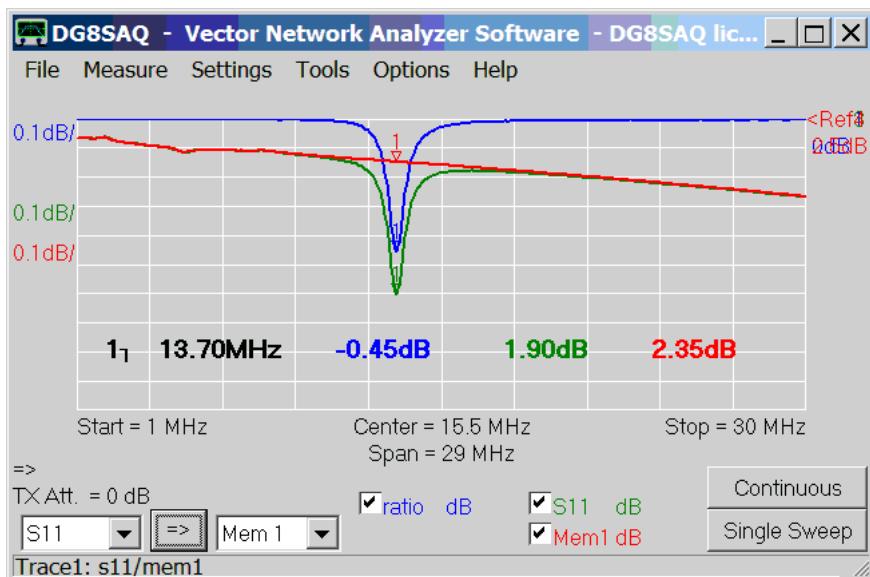
We can clearly see the tag's resonance and find its frequency with the aid of a minimum marker at 13.70 MHz.

Next we perform our special "calibration" step by means of a custom trace:



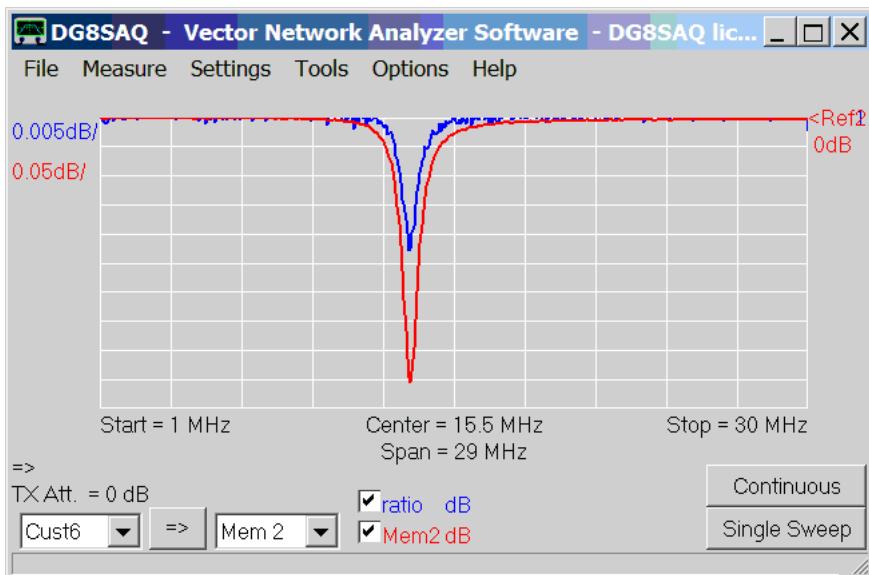
As mentioned above, we divide any reflect measurement (S_{11}) by the reference measurement without resonator in memory space Mem1. We assign the name "ratio" to this trace.

The resulting blue "ratio"-trace nicely shows the tag resonance alone, i.e. how much RF power is being absorbed by the tag as a function of frequency:



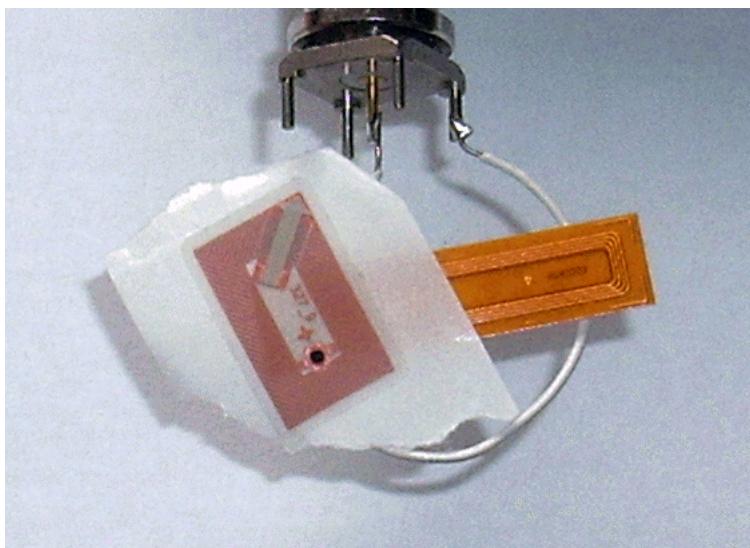
Note, that with every subsequent S_{11} measurement the ratio trace is being recalculated and updated in real time during the sweep. So, it indeed conveniently serves as a calibration method.

Varying the distance between coil and tag does not change the tag resonance frequency:

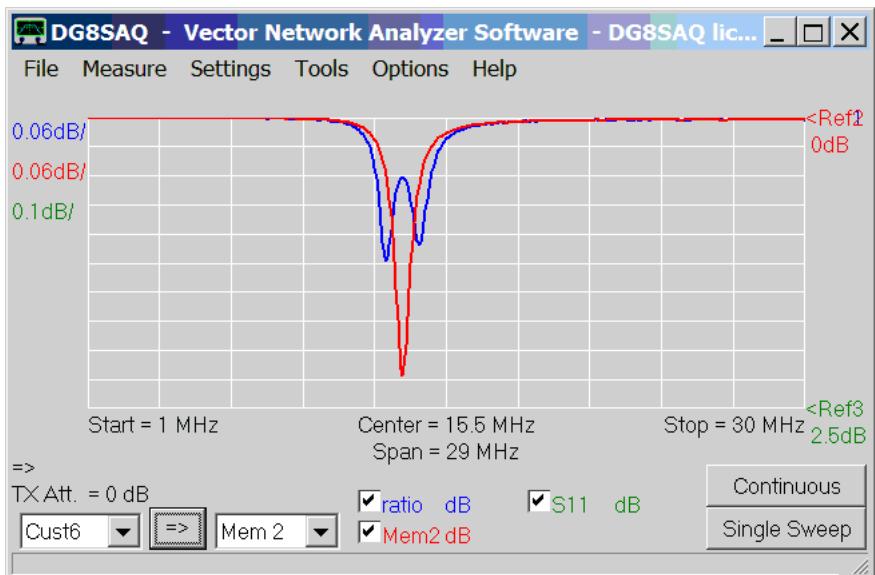


While above red trace was taken with the tag positioned right on top of the detector coil, the blue trace was taken with the tag being spaced 2 cm above the coil.

For the fun of it, we place a second tag on top of the first one:



And sure enough, we see resonant detuning through the coupling of the two tag resonators to each other:

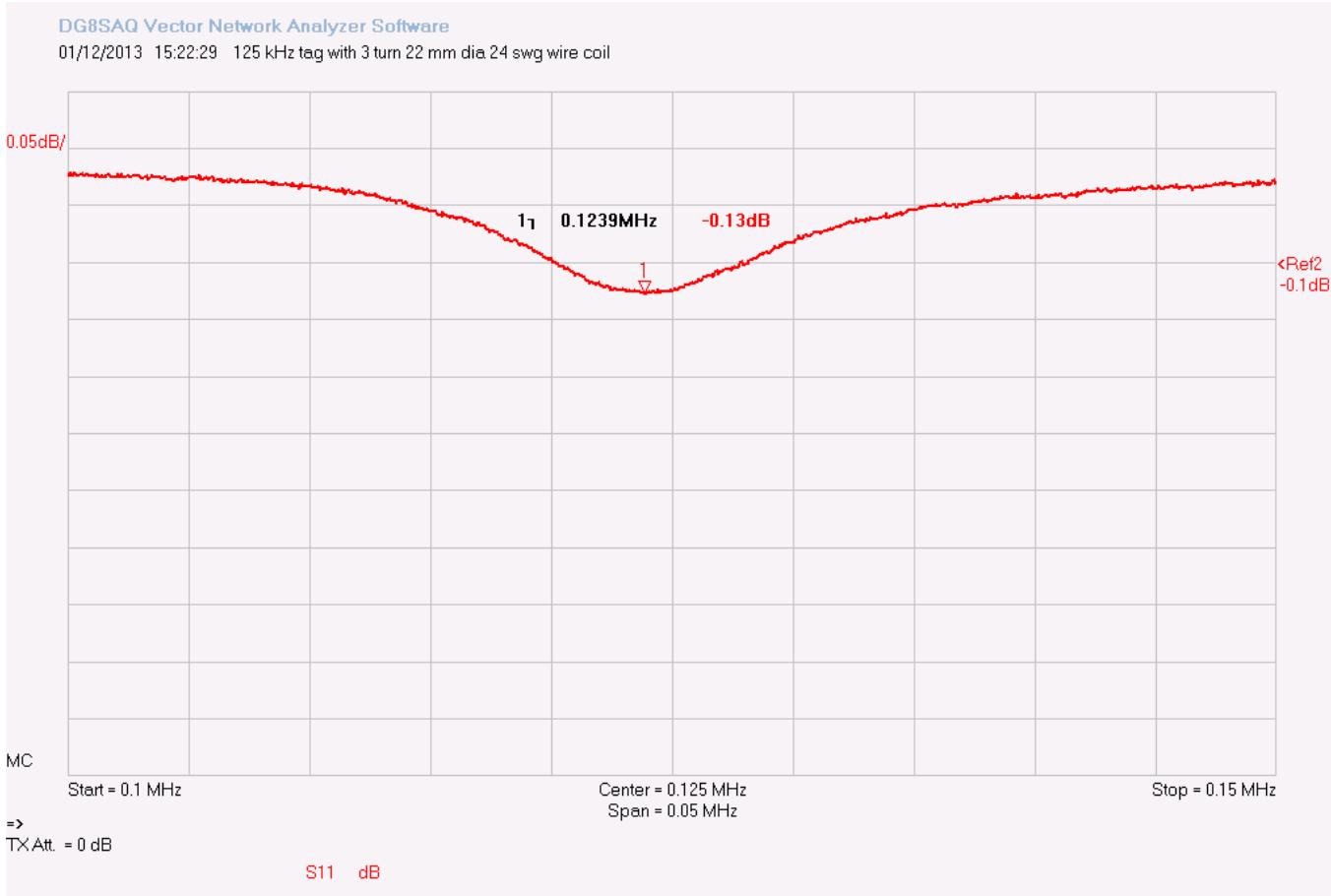


Above red trace shows the response of the lower (brownish, lengthy) tag alone, while the blue trace shows the response of the stack of both tags.

The same test method can be applied to LF RF ID tags operating at 125 kHz. Below image shows a 125 kHz RF ID tag placed onto a pickup coil with 3 turns (green wire):



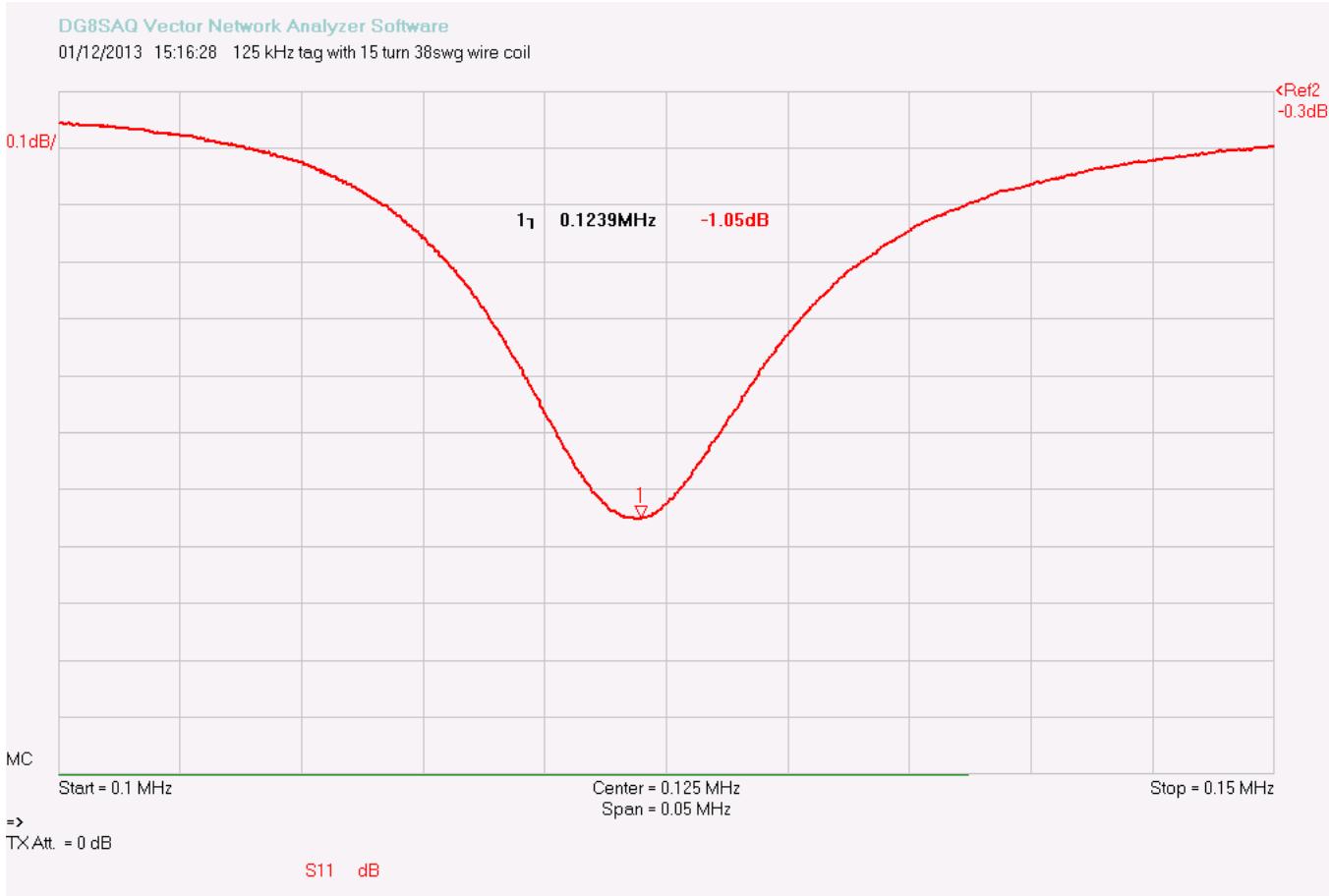
The absorption signal is weak but clearly detectable:



The signal to noise ratio can be improved by using a detector coil with a higher number of turns, e.g. 15 turns as in the picture below:



The absorption signal becomes a lot stronger:



Now, what is the optimum pickup coil for a certain RF ID tag?

On the one hand the coil should have many turns in order to maximize the magnetic field for good coupling to the tag.

On the other hand, the natural parallel resonant frequency of the coil (coil inductance parallel to parasitic coil capacitance) should be well above the tag's resonant frequency, to avoid detuning of the tag by resonant coupling to the pickup coil.

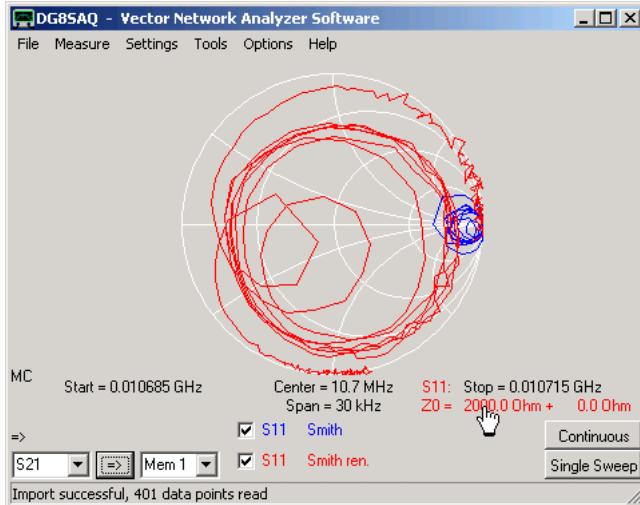
The pickup coil is best optimized experimentally starting with a large number of turns and removing turns until the parallel resonance of the coil alone is well above the tag frequency, e.g. by at least a factor of two.

RENORMALIZED SMITH CHART

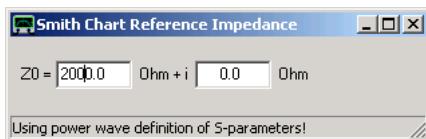
The VNWA software can recalculate **one port reflection coefficients** to an arbitrary **reference impedance** and display it as **renormalized Smith** data.

Most Smith charts display reflection data normalized to a reference impedance of 50 Ohms. If a different reference impedance is chosen, e.g. 75 Ohms, the reflection coefficients will change. For example, a 75 Ohms resistor will have a reflection coefficient of zero if normalized to 75 Ohms reference impedance or in other words in a 75 Ohms system.

An Example is shown below:



S11 is the input reflection coefficient of a crystal filter terminated with 50 Ohms on output. The blue curve near the open circuit point in the Smith chart shows, that the input impedance is much higher than the reference impedance of 50 Ohms. This changes, if a reference impedance of 2000 Ohms (see red **impedance label** under hand cursor). The magnitude reflection coefficient is still quite high in this case, as the power is still reflected at the mismatched 50 Ohms termination on output. To modify the reference impedance, double-click the red impedance label. The reference impedance window will open, where new values can be entered.



►►► **Hint:** You can also change the real part (imaginary part) of the reference impedance, by holding the mouse over the real part (imaginary part) of the red impedance label in the VNWA main window and turn the **mouse wheel**.

►►► **Note:** The definition of a reflection coefficient is only unique as long as the reference impedance is purely real. For a complex reference impedance I have chosen to implement the **power wave** formulation, which allows to interpret the reflection coefficient in terms of reflected power like of the real reference impedance case.

►►► **Hint:** To renormalize **two port S-parameters** to a new reference impedance, use the matching tool.

IMPORTING AND MANIPULATING A BACKGROUND IMAGE

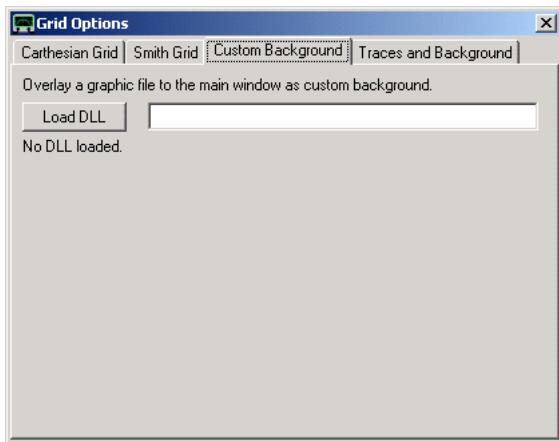
The custom background options allow to **load an image to appear as background on the VNWA main window**. Image file import is implemented in an optional external dynamic link library (dll) to avoid incompatibilities with older Windows versions (Windows98 and Windows2000 do not support GDI, which is used to read and display *.svg-files). Two different dlls are provided for reading image files:

svg_dll.dll supports import of scalable vector graphics files (*.svg)

gfx_dll.dll supports import of most pixel graphics files like *.bmp, *.jpg, *.png... .

Moreover, the source code of a simple dll-example for reading *.bmp files is provided in order to document the programming interface.

Before importing an image, an appropriate dll-file must be loaded by pressing the "Load DLL" button and selecting the dll-file:



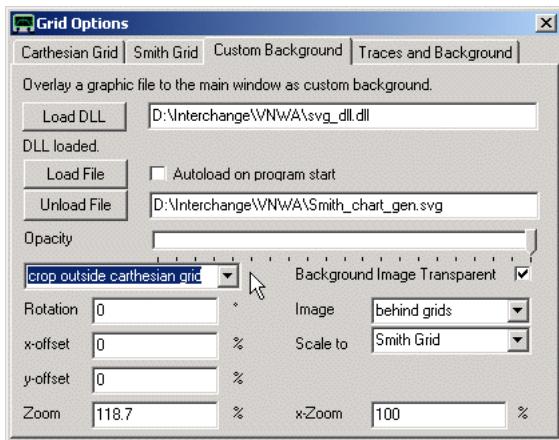
Once the dll is loaded, the controls for loading and manipulating an image appear and an image file can be loaded.

Example 1: Loading a detailed Smith chart on top of the standard one

Wikipedia offers a detailed Smith chart in svg-format:

http://upload.wikimedia.org/wikipedia/commons/7/7a/Smith_chart_gen.svg

The svg or scalable vector graphic format has the advantage of offering constant high display quality on all magnification scales. In order to load this file, **svg_dll.dll** must be loaded first. Then the image can be loaded by pressing the "Load File" button:



There are several possibilities to manipulate the displayed image:

Opacity ...allows to blend the image more and more into the background until it disappears

cropping ...means cutting off the image portions that are extending outside the selected grid

transparent ...if checked, all portions of the image having exactly the same color as the lower left image pixel become transparent, i.e. invisible. Particularly useful for black and white graphs to make the white portion transparent

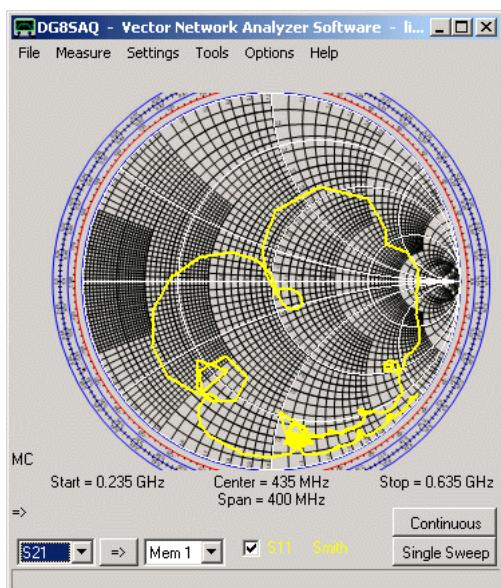
The **image** can be placed in front of or behind the grid lines

It can be **scaled** to one of the grids (i.e. if the grid is resized, the image will be resized accordingly or it can be displayed with fixed size in one of the main window corners)

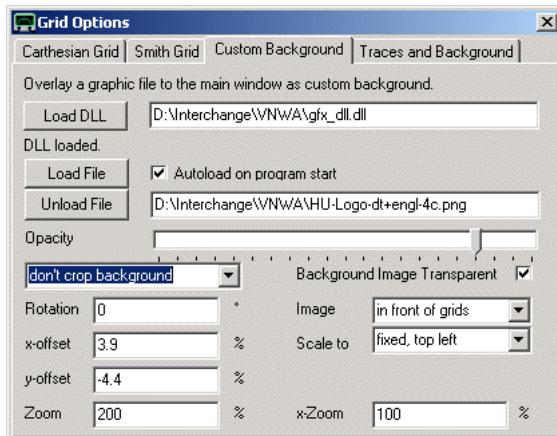
Moreover, the image can be **rotated**, resized by a **zoom-factor**, stretched by an **x-zoom-factor** and repositioned by specifying **offsets**.

The image can be chosen to **autoload** upon program start.

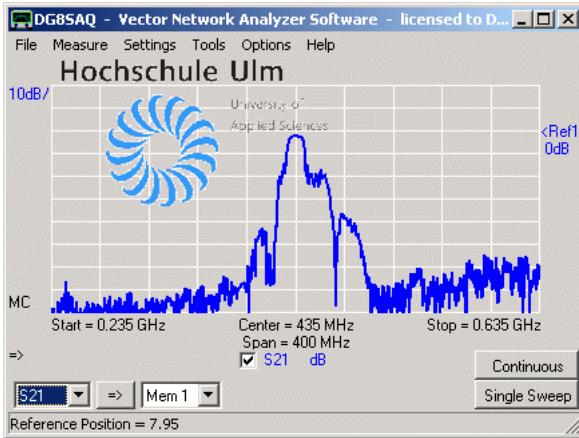
With above settings, the image displays like this:



Example 2: Loading a company logo

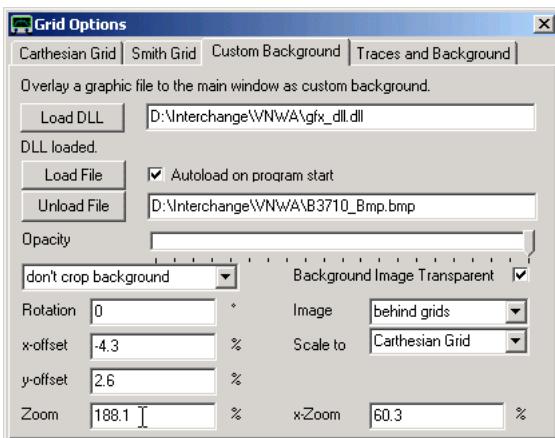


Here a logo in *.png format has been loaded using the gfx_dll.dll and the image has been fixed to the upper left corner of the main window:

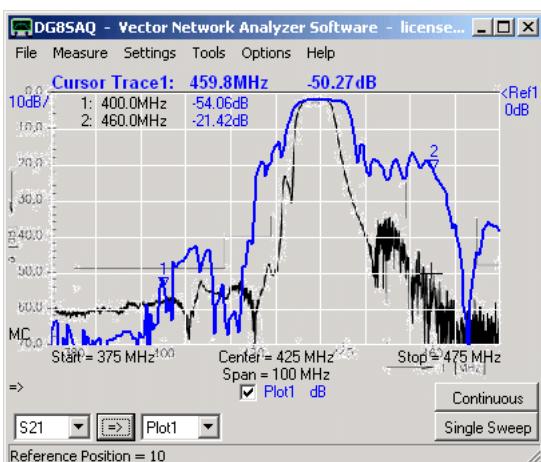


Example 3: Comparing a measurement with a graph from a print or a graphic file

Very often one comes into the situation where one would like to compare an own measurement result with a measurement result published in the literature. Usually, the literature result is not available as S-parameter file, but only as print or pdf-file. In this case, an image file of the literature result can be created by either scanning the paper or extracting the image from the pdf document. In the following, I have extracted an image from a data sheet of a similar filter and loaded it using the gfx_dll.dll:

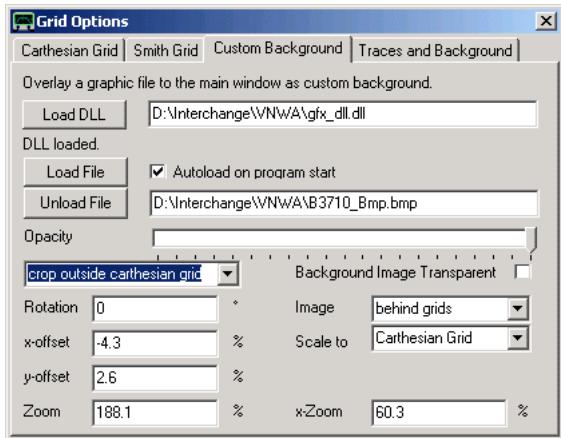


After properly shifting and scaling the image and adjusting the VNWA grid and frequency scale, the image scales match the VNWA scales. As the image is scaled to the cartesian grid, the scale match will be maintained even if the window size and aspect ratio is changed. Now, a measurement can be overlaid. The cursor can be used to compare image values with measurement values:

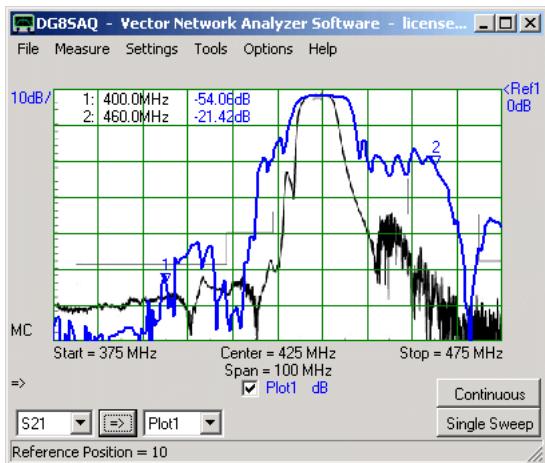


Note, that in above example, the image has some portions that are extending outside the cartesian grid, which might obscure VNWA features. Also, near the image curve transparency is not perfect, which is due to color interpolation in the original image format.

Here, it might be nicer, to make the image non-transparent yielding the white image background and cropping the image parts extending outside the grid:



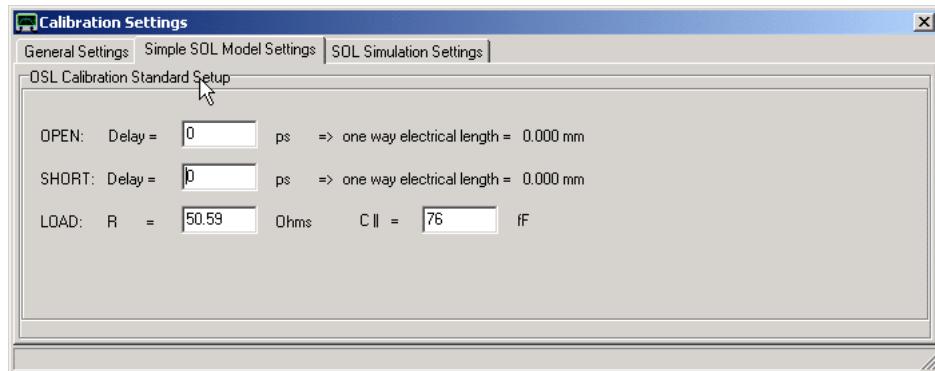
With above settings, the result looks like this:



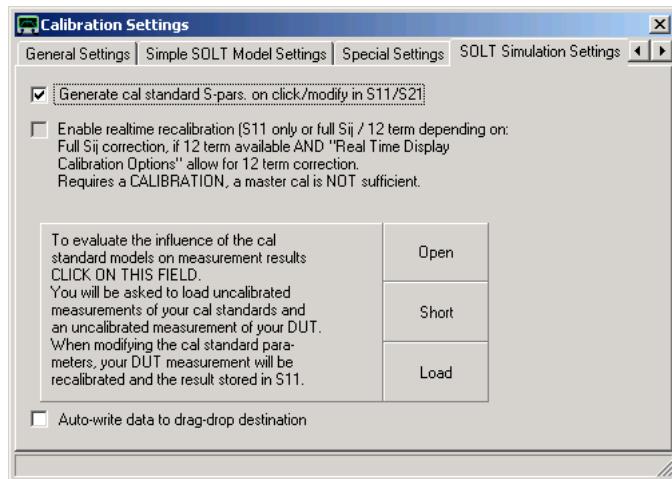
Here you can clearly see that my filter is a lot worse than the literature example.

EXTRACTING CALIBRATION KIT PARAMETERS

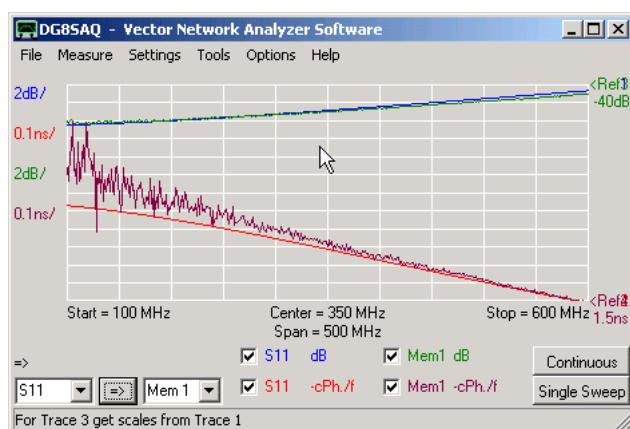
In order to obtain accurate measurement results, a good knowledge of the used **calibration kit parameters** is mandatory. These must be entered into the "Calibration Kit Settings" menu:



Extracting parameters from reference measurements



When activating "**Generate S-pars. on click/modify in S11**", the reflection coefficient of the modified cal standard will be simulated and stored in S11. Note, that the parameters can be "tuned" with the mouse wheel and the changes will be visible on the VNWA main window in real-time. This can be used to fit the calibration kit model to a reference measurement:



The above example shows the reflection coefficient of my load as measured on a R&S ZVM (Mem1) compared to the

model generated with above parameters. Note, that the model reacts very sensitive on changes in model parameters.

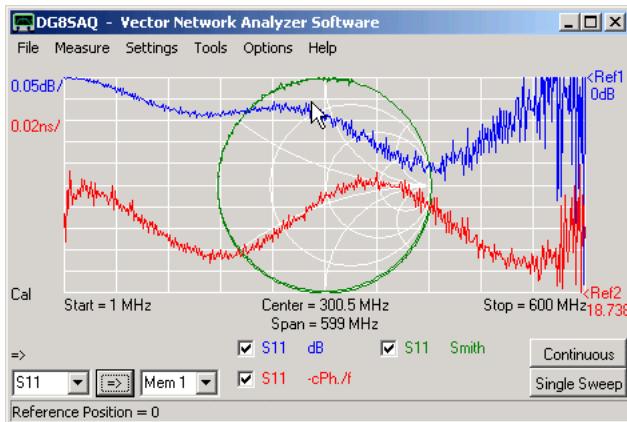
Extracting parameters with the aid of an open transmission line

If no reference VNWA is available, then there is still a possibility to extract model parameters. To do so, do the following:

1) In the "Calibration Settings" window select **custom calibration standards**, but enter the values for ideal calibration kits (Load: 50 Ohms real, all delays zero).

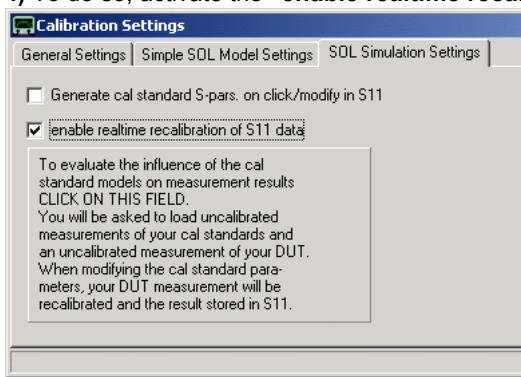
2) Perform a **reflection calibration** (SOL, S11 only) over a frequency range of 1...500 MHz with your unknown calibration kit. Calibrate the VNWA output without a test cable for best accuracy.

3) Measure the input reflection coefficient of 30cm straight high quality semi-rigid 50 Ohms transmission line with its output open or shorted. The result should ideally be a reflection coefficient of magnitude 1 circling around the edge of the Smith chart:



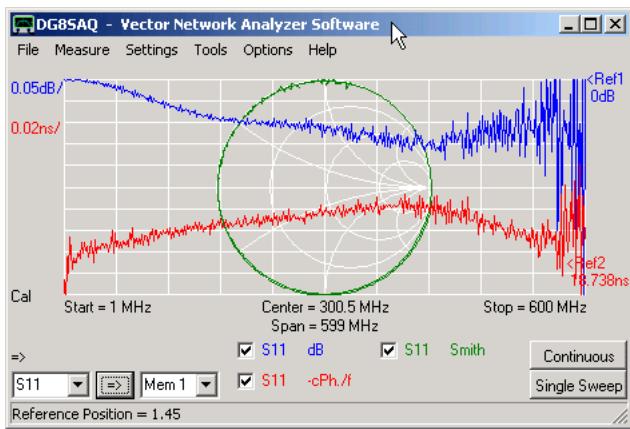
Basically, this is the case. But looking at it a bit closer, the magnitude of the reflection coefficient must decrease with increasing frequency due to the increasing transmission line loss. This is observed, but **superimposed, an oscillatory behavior** with increasing amplitude is visible (blue trace), which may lead to unphysical reflection coefficients greater than unity (outside the Smith chart). Similar oscillations are observed in the phase progression (red trace). These oscillations are not real, but a mathematical artefact arising from the fact, that the real calibration coefficient parameters are different from those used in the correction calculations. This effect is known in the literature as **port mismatch**. Knowing that much, it is straight forward to tune the calibration kit parameters such that the oscillations on both curves vanish in the best possible way and the phase progression becomes as constant as possible (constant "circling speed" around the Smith chart).

4) To do so, activate the "**enable realtime recalibration**" checkbox in the "SOL Simulation Settings" tab:



5) Keep the open delay at constant zero value. This will fix the calibration plane.

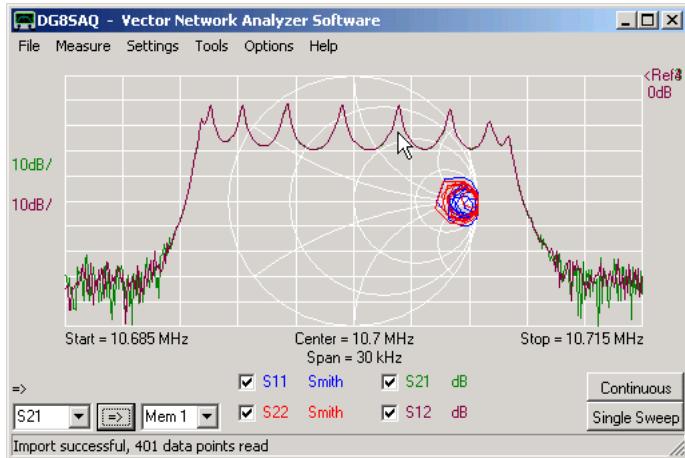
6) Tune short delay, load R and load C|| with the mouse wheel and observe the change to the recalibrated S11 in the VNWA main window. A good choice with reduced port mismatch is seen below.



MATCHING TOOL: PERFORMING A MATCHING ANALYSIS

Determining the impedance environment for optimum DUT performance

Note, that the VNWA measures **S-parameters**, that are normalized to **50 Ohms source and load impedance!** This might not be the desired operating conditions for your DUT. E.g. a narrowband crystal filter is usually a high impedance device and performs very poorly, when terminated with 50 Ohms at input and output. The picture below shows a measurement of a 10.7 MHz telecommunication crystal IF-filter.



Note, that the filter S21 shows a passband ripple of close to 20dB. Note also, that like for all reciprocal DUTs (all passive devices except isolators and circulators) $S_{21}=S_{12}$.

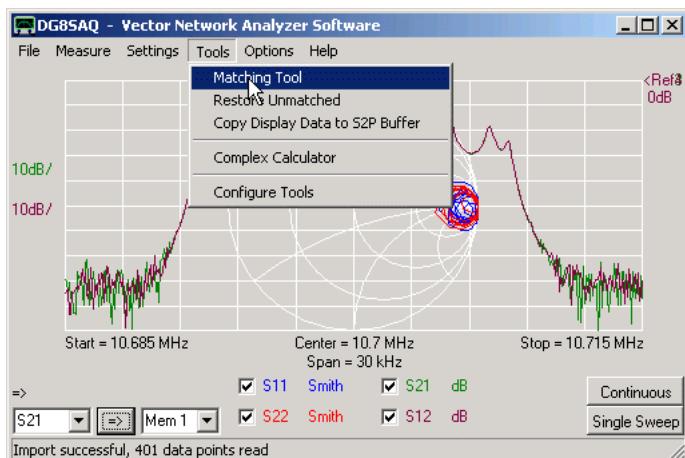
As you can see from S11 and S22, the filter is severely mismatched, as its impedances are near the open circuit point in the Smith chart.

When building the filter into a receiver, the high filter impedances must be transformed with inductors and capacitors to match the impedances of the surrounding circuitry. This is called **matching**. Most the time, it is desirable, that the output impedance of one section (e.g. driving transistor) equals the complex conjugate of the following section's input impedance. This condition is called **power match**, as it ensures, that all power is transferred from one stage to the next. Sometimes, different matching conditions are used, e.g. noise match, to optimize the signal to noise ratio.

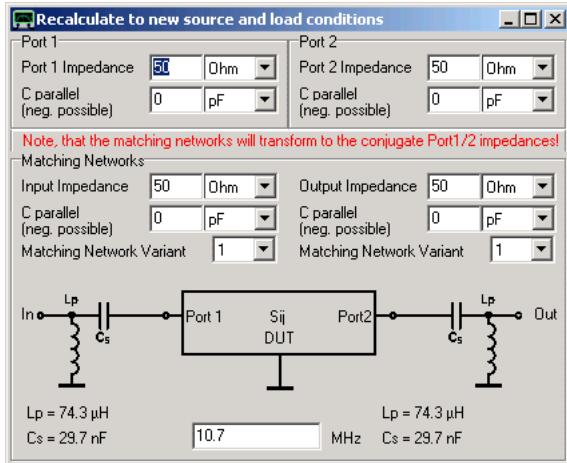
The VNWA's built-in matching tool allows to simulate the DUT's transfer characteristics under arbitrary impedance termination conditions.

►►► **Note:** You must have a complete set of four valid S-parameters (S11,S21,S12,S22) in order to use the matching tool.

To open the matching tool, select "**Tools**"-"**Matching Tool**":

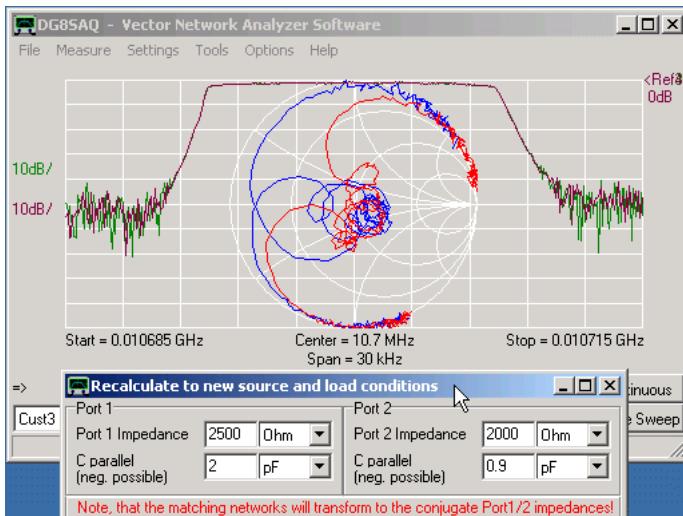


The matching tool window will pop up.



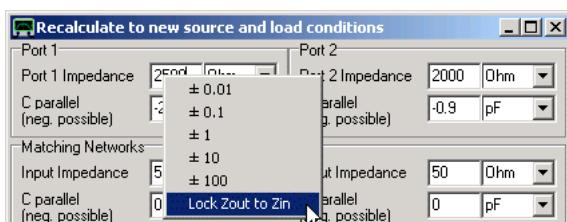
Place this window in such a way on the desktop, that you can see both, the VNWA main window with your measurement curves and the matching tool window.

In the upper part of the matching tool window, you can specify the **your DUT's impedances**. At startup, these are real 50 Ohms on input and output (=impedances of the VNWA). If you modify them, you will find, that the S-parameters on the main window change. The crystal filter of our example has impedances of about 2000 Ohms resistance parallel to about 1pF capacitance (thus impedances represent slightly capacitive loads) on input and output. Entering these impedances yields a perfectly flat passband and a good power match:



Note: The mathematical transformation from the measured 50 Ohms impedance level to calculated 2 kOhms imposes a crucial test on the accuracy and consistency of the four S-parameters ($S_{11}, S_{21}, S_{12}, S_{22}$). If the passband shows spikes, this is a sign that there was some measurement error, e.g. wrong calibration, inconsistent reference planes, no 6 term correction.... .

Hint: You can also modify the impedances with the **mouse-wheel**. To change the increment, right-click onto the according input field. Here, you can also **lock input and output impedances**. This might be useful, if you have a symmetric DUT and you want to change input and output impedances synchronously:

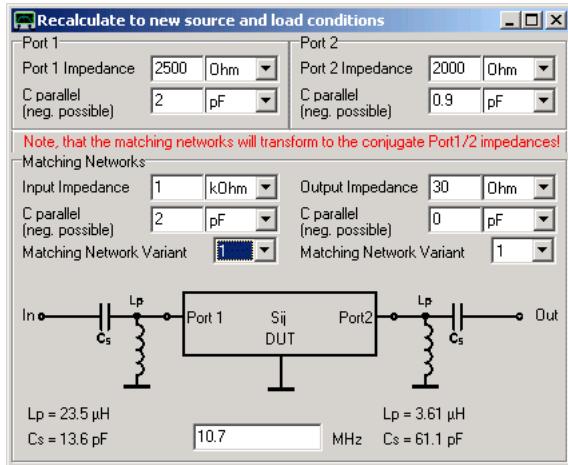


By now, we have determined the impedance environment, our IF filter (=the DUT) wants to see for best performance, namely the **complex conjugates** of the port 1 and port 2 impedances.

Determining the matching networks

If we want to actually place the above filter in between a mixer output and an amplifier input, we need to **match the mixer output to the filter input and the filter output to the amplifier input with two matching networks**.

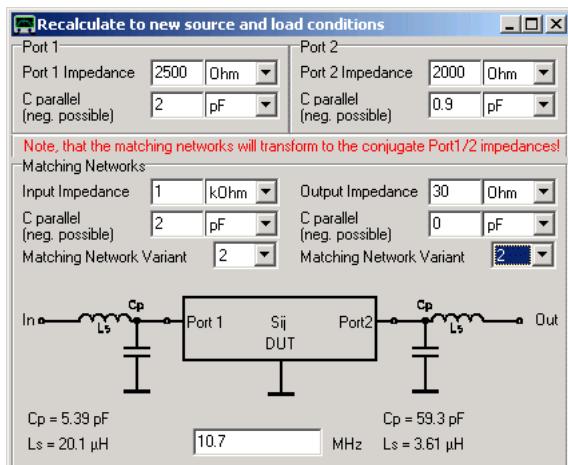
The matching networks can be calculated in the lower part of the matching tool window. Assume, the mixer output has an impedance equivalent to $1 \text{ kOhm} \parallel 2 \text{ pF}$ and the subsequent IF amplifier stage has an input impedance of real 30 Ohms (i.e. $\parallel 0 \text{ pF}$). Enter these impedances into the lower part of the matching tool window:



The calculated matching networks are shown in a little schematic. With the displayed matching networks, the filter will yield optimum performance now, if the **In-terminal** sees $1 \text{ kOhm} \parallel 2 \text{ pF}$ and the **Out-terminal** sees 30 Ohms.

►►► **Note:** Calculations are performed with ideal components with infinite Q-values.

►►► **Hint:** Identical impedance transformations can be obtained with up to 4 different variants of matching network topologies. You can select the one you like with the combo-boxes:



►►► **Note:** The impedance transformations are **calculated at a single frequency**, by default the center frequency. If this doesn't coincide with your filter's passband, you can change the frequency used by the simulation in the edit field at the very bottom.

MATCHING NETWORK TUTORIAL

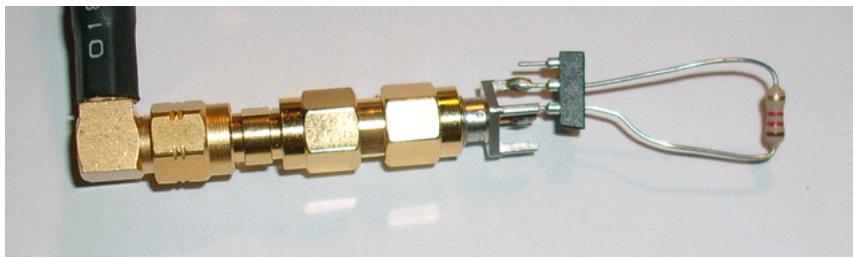
This tutorial describes, how the VNWA can be used to **generate and optimize an impedance matching network**, that transforms a given source impedance (in our example 50 Ohms) to a given load impedance.

►►► **Note:** For all measurements, the **VNWA must be properly calibrated**. Some of below screenshots show the VNWA in uncalibrated state. These screenshots have been created from earlier recorded S-parameters. All S-parameters have been measured with the calibrated instrument.

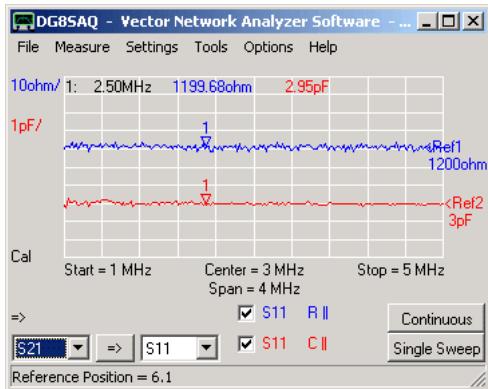
►►► **Note:** If you repeat below experiments in the VHF/UHF range, a proper setting of the calibration plane using port extensions will be most important. Since below example is performed at 2.5MHz (Wavelength >100m), a few mm shift in the calibration plane will hardly be detectable.

1. First, we need to know the source and load impedance between which we want to achieve a power match, i.e. a matching condition with minimum power loss.

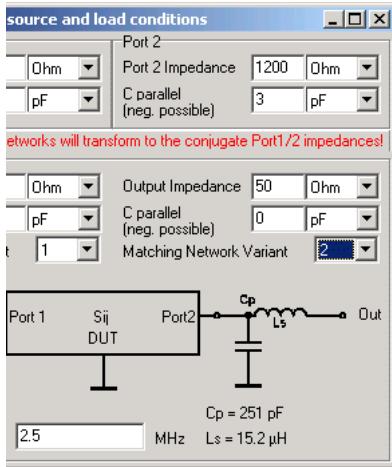
For simplicity, we use 50 Ohms as the impedance of our source. This could be the output of a transmitter. Our load impedance in this example will be a 1,2kOhm wired carbon resistor:



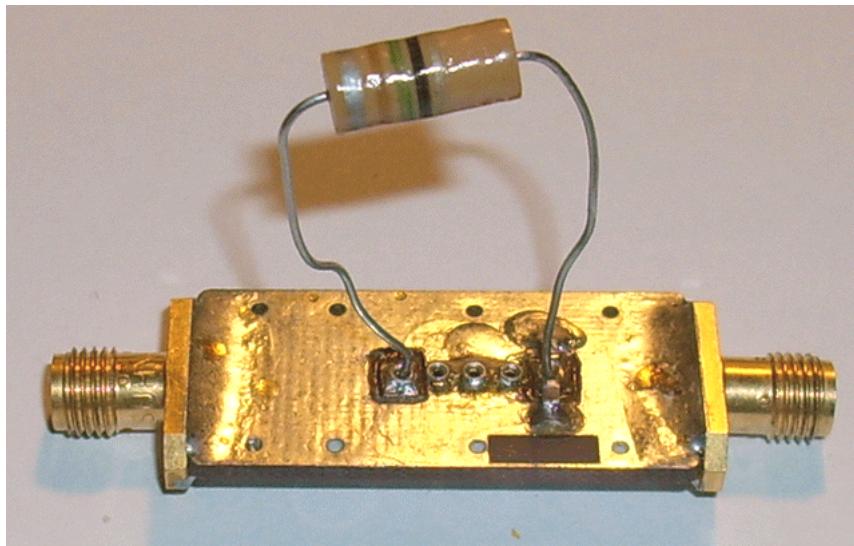
This could be a model of a high impedance antenna. To be on the safe side, we measure the impedance of our load in the frequency range of interest:



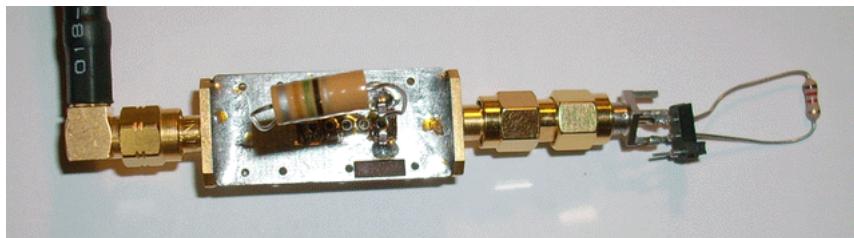
2. We want to achieve a power match at the marker frequency of 2.5MHz, therefore we calculate a matching network for this frequency. This can be done with the VNWA matching tool. We only need half of it. We enter our source impedance (50 Ohms) and the measured load impedance (1200Ohms || 3pF), the frequency where the matching condition is to be achieved (2.5MHz) and select a matching topology (#2):



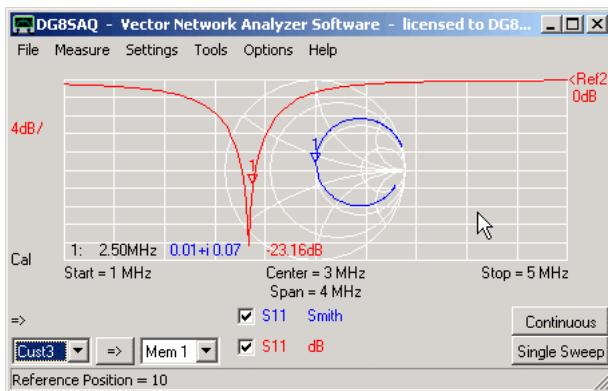
In order to build up the matching network we will need a 15uH inductor (standard value) and a 251pF capacitor which we will approximately create by connecting a 150pF and a 100pF capacitor in parallel. All components are connected on a test board:



3. We connect our high impedance load to the matching network output ...

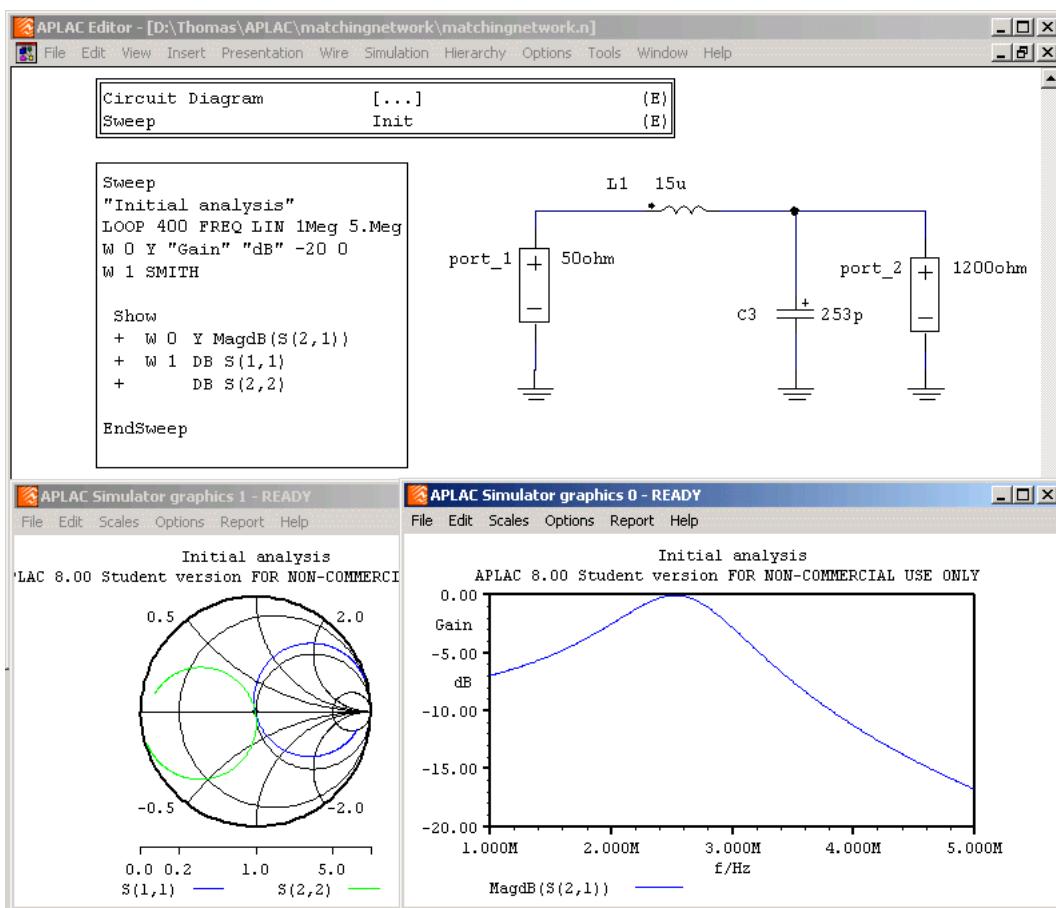


... and measure the input reflection coefficient:



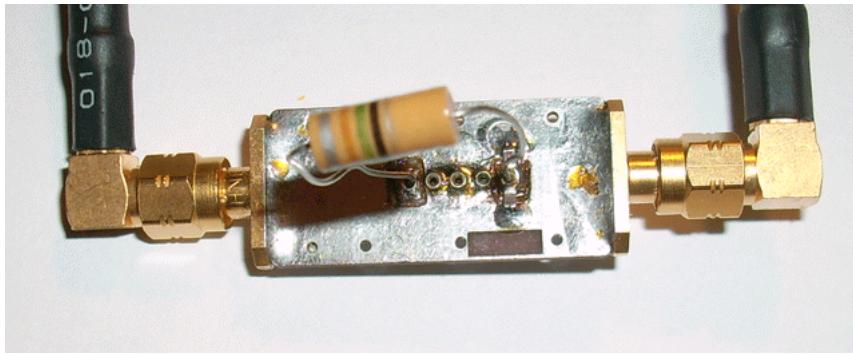
We observe that matching works. We achieve a return loss on input which is better than 20dB. Now the question arises, what is the insertion loss of our matching network.

We could do a simulation e.g. with APLAC:

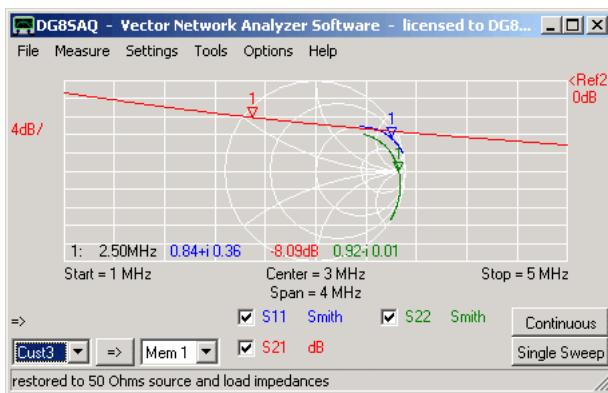


We see that S_{11} looks like the measurement and S_{21} becomes nearly 0dB at 2.5MHz because we have simulated with lossless ideal components.

4. In order to find the real insertion attenuation of the matching network, we must perform a full 2-port measurement of the matching network (S_{11} , S_{21} , S_{12} , S_{22}):

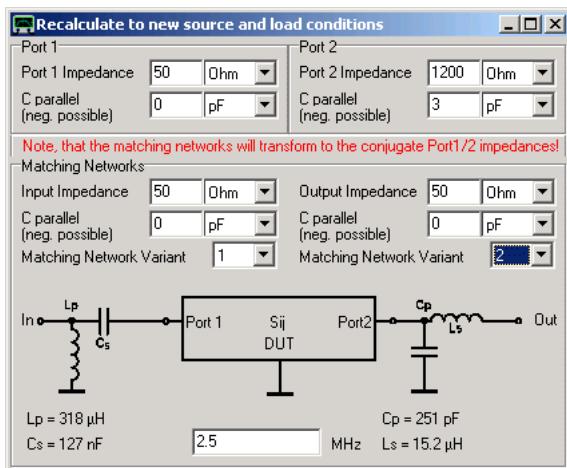


We will see an insertion loss of about 8dB and a severe mismatch on input and output:

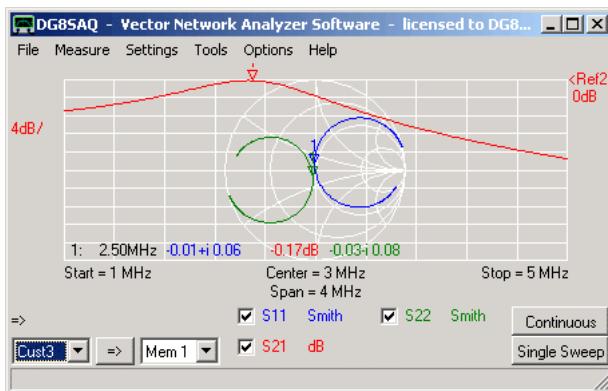


This is because the matching network is now terminated with 50 Ohms of the VNWA.

5. Using the matching tool we can recalculate the S-parameters to what they would be like if the load impedance were the one of our hardware load. We enter the source and load impedance and the matching frequency:

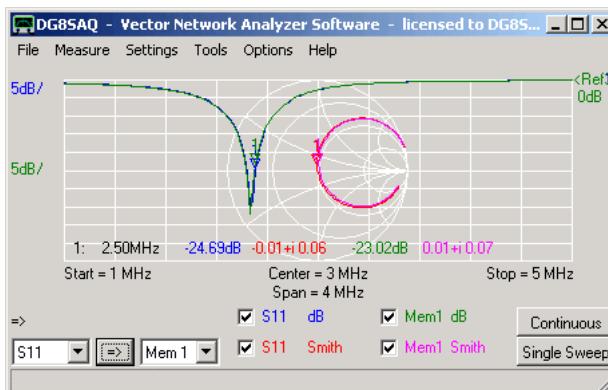


We observe how the S-parameters change under the new load condition:



Now we can see that our matching network introduces a transmission loss of 0.17dB, which is very good. Observe how closely this recalculated measurement resembles the APLAC simulation.

6. The renormalized S11 for the new load condition should be identical to the measured S11 when our hardware load is connected to the matching network output. This can be verified by a measurement:



S11 contains the renormalized 50Ohm measurement while Mem1 contains the input reflection coefficient of the matching network with the hardware load connected to the output. As expected the two are identical.

CRYSTAL ANALYZER: EXTRACTING RESONATOR MODEL PARAMETERS

General

Example 1: Reflection Measurement

Example 2: Transmission Measurement

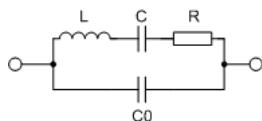
General Hints

Analyzing a batch of crystals

Example 3: Mixed measurements with virtual ground calibration

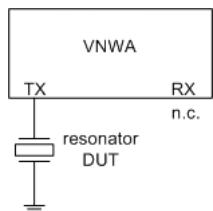
General

A crystal or any other electrical resonator can be described reasonably accurate by the following **equivalent circuit model** :

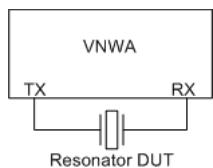


Knowledge of the model parameters is useful as input to circuit simulators for e.g. synthesizing crystal filters or modeling crystal oscillators. The **VNWA crystal analyzer tool** allows to **extract these model parameters from a measurement**.

A crystal or one port resonator can be measured in **reflection** (S11)



or in **transmission** (S21)



Often low impedance (e.g. 12.5 Ohms) test fixtures are available for crystal transmission measurements. Therefore the **fixture impedance must be taken into account** for transmission measurements. **You should NOT use a low impedance fixture for reflection measurements, unless you exactly know how to calibrate it properly.**

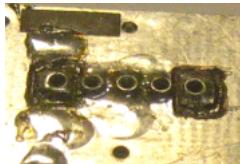
The following examples show several methods to measure crystal resonators and to extract model parameters. In all measurements 3000 data points and a sweep time of 2ms/data point have been used for measurement. This is not a recommendation but simply the first and only settings I have tried. Feel free to experiment. But keep in mind that a crystal is an extremely narrow band device and as such requires some time for settling. So, do not sweep too fast. Also, the accuracy of your calibration will improve with slower sweep rates.

Example 1: Reflection Measurement

1. Do a proper SOL calibration of your VNWA + test fixture:

You might want to use customized calibration standards that plug straight into your test fixture for this purpose, e.g.

OPEN:



SHORT:



LOAD:

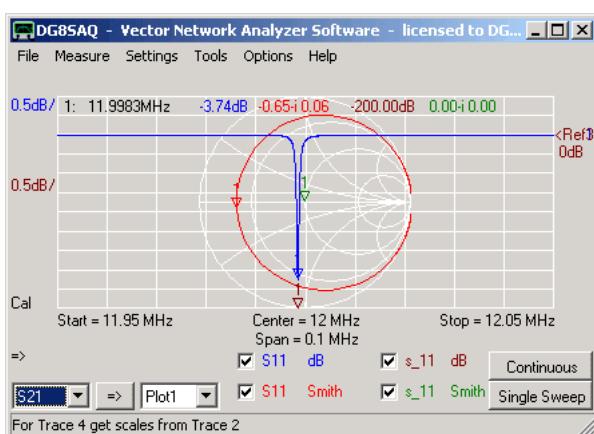


2. Perform a reflection (S11) measurement of your crystal with one crystal pin grounded.



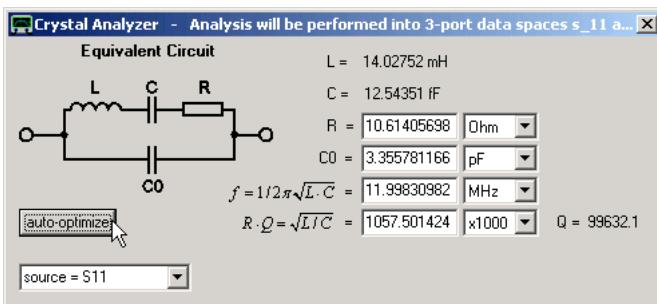
By doing so, you measure the reflection coefficient of your resonator as 1-port device. Make sure, the crystal is connected as close as possible to the calibration plane.

This is what the measurement result should look like for a 12 MHz crystal:

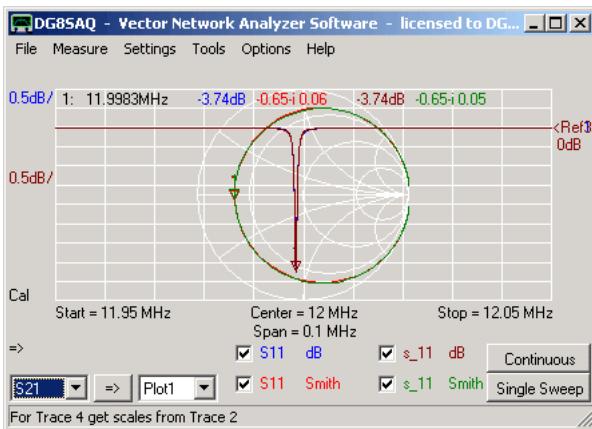


Make sure, to display 3-port memory space s_11 before moving to the next step. 3-port memory spaces s_11 and s_21 will be used to store and allow to display the model simulation in reflection and transmission.

3. Start the "Crystal Analyzer" tool via the main tools menu.



4. The above is what you see a short moment **after having pressed the auto optimize button**. Make sure that **source = S11** is selected before auto-optimizing. The displayed equivalent circuit model is fitted to the measurement data. The simulated data is stored in s_11 (and s_21) for comparison. Note that there is a perfect match between measurement and simulation:



►► Hint: After auto-optimization, the fitted model parameters are automatically stored in the Windows cut-and-paste buffer and can be pasted into any text document in the following form:

```
F = 11998309.8281313
Q = 99632.1600629463
L = 0.0140275240017728
C = 1.25435105103075E-14
R = 10.6140569892503
C0= 3.35578116630557E-12
F1= 12020712.3
F2= 12020712.9
F3= 11998309.2
F4= 12020632.9
F5= 12020772.2
F6= 11998312.5
```

Units are Hz, none, Hy, F, Ohms and F.

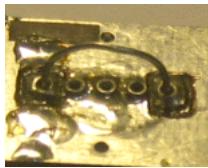
Note: Frequencies F1...F6 will be explained in the next section about measuring the crystal in transmission.

Example 2: Transmission Measurement

1. Do a Thru calibration of your VNWA + test fixture:

You might want to use a customized calibration standard that plugs straight into your test fixture for this purpose, e.g.

THRU:

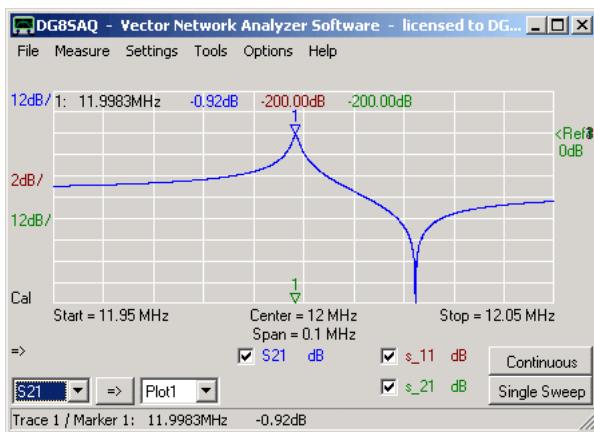


2. Perform a transmission (S_{21}) measurement of your crystal with one crystal terminal being input and the other being output.



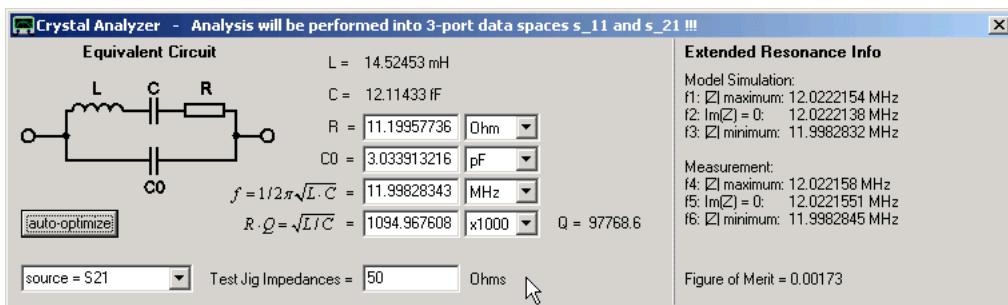
By doing so, you measure your resonator as 2-port device. Make sure, the crystal is connected as close as possible to the calibration plane.

This is what the measurement result should look like for a 12 MHz crystal:



In transmission one can clearly see the serial resonance (marker) as well as the parallel antiresonance. Make sure, to display 3-port memory space s_{21} before moving to the next step. 3-port memory spaces s_{11} and s_{21} will be used to store and allow to display the model simulation in reflection and transmission.

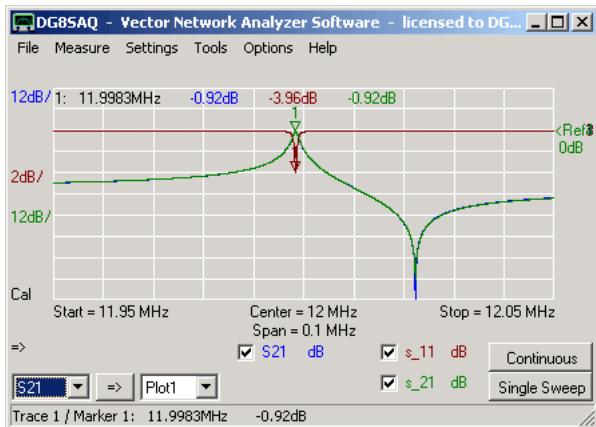
3. Start the "Crystal Analyzer" tool via the main tools menu and increase the width of the crystal analyzer window:



4. Make sure that source = S_{21} and the correct test fixture impedance (must be identical on input and output) is selected before auto-optimizing. The above is what you see a short moment after having pressed the auto optimize button. The displayed equivalent circuit model is fitted to the measurement data.

Note that on the right hand side of the crystal tool, various frequencies are extracted from the simulation as well as from the measurement data. f_1/f_4 and f_2/f_5 are two different definitions of the parallel antiresonance frequency. f_3/f_6 is a definition of the serial resonance frequency differing slightly from f on the left hand side. This data is also calculated for reflection measurement fits.

The simulated data is stored in s_21 (and s_11) for comparison. Note that there is a perfect match between measurement and simulation:



Note, that after auto-optimization, the fitted model parameters are automatically stored in the Windows cut-and-paste buffer and can be pasted into any text document like in the reflection example.

General Hints

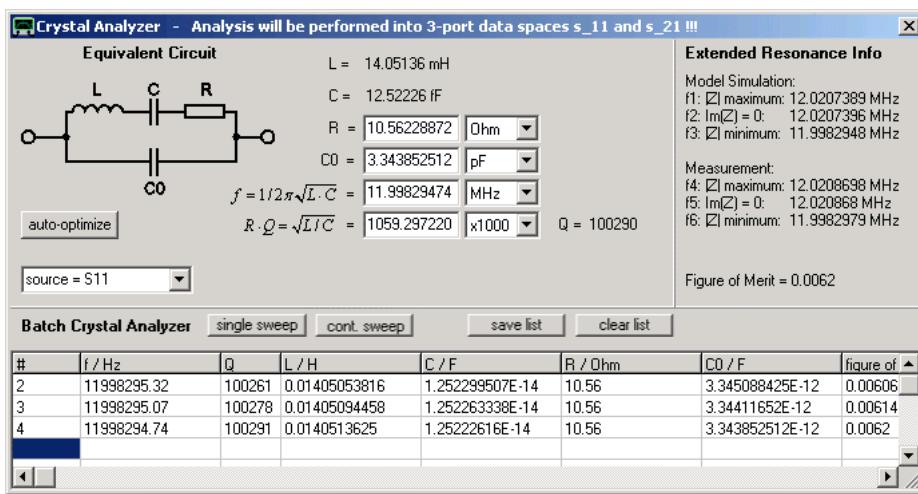
You can also **manually optimize** by changing the data in the edit fields by keyboard or by **mouse-wheel** while observing the changes of s_11 and s_21 on the VNWA main window..

Mouse-wheel increments can be set individually by **right-clicking onto the edit fields**.

The displayed **figure of merit** is a distance between simulation and measurement and should thus be as close to zero as possible.

Analyzing a batch of crystals

If you want to analyze a whole batch of more or less identical crystals, you can increase the size of the crystal analyzer window downwards in order to access the batch table:



On pressing the "single sweep" button, a measurement sweep will be started and at its end, the data will be analyzed and the values will be entered into the table at the cursor position. You can remeasure a line, by selecting it with the mouse prior to pressing measure. You can save the list into a csv-file, which can be read by Excel. You can also erase the list.

The line numbering is assigned automatically, but you can manually change it if desired. The above picture shows three consecutive measurements on the same crystal.

Hint: The fitted model parameters of the last sweep will also be automatically stored in the Windows cut-and-paste

buffer and can be pasted into any text document.

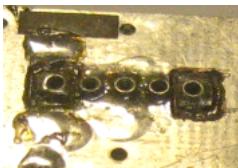
►►► **Hint:** The "cont. sweep" function can be used to continuously monitor a single crystal over time. The crystal will be swept continuously. After every completed sweep, a line will be added to the fit data table. Note, that the time when the sweep was done (in seconds after having started continuous sweeping mode) is logged in the last column of the table. This is useful to e.g. measure the crystal parameters vs. varying temperature.

Example 3: Mixed measurements with virtual ground calibration

Kurt Poulsen OZ7OU has come up with the idea of calibrating the VNWA+test fixture such that the hot RX pin serves as a virtual ground for the reflection measurement. In the following this type of calibration is referred to as virtual ground calibration. It is an SOLT calibration where the real ground is replaced by a virtual ground for reflection measurements only. The advantage of this type of calibration is that you can measure your crystal in transmission and in reflection without having to change connections in the test fixture. So, the crystal stays in the fixture in the same way (between hot TX pin and hot RX pin) no matter if you measure it in reflection or in transmission.

1. Do a virtual ground calibration of your VNWA + test fixture:

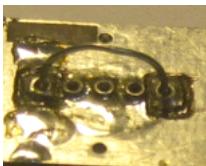
OPEN:



LOAD:



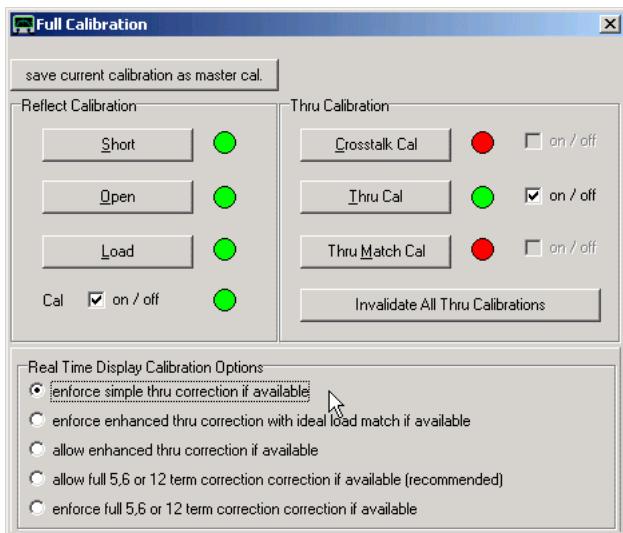
SHORT and THRU:



Don't perform a thru match calibration.

►►► **IMPORTANT:** Select "enforce simple thru correction..." as real time display calibration option (see mouse pointer below).

►►► **Remark:** The parallel capacitance C0 that will be calculated from the measurement, will contain also the capacitance of the test fixture from input to output. This test fixture capacitance can be effectively removed by performing an **isolation calibration** with the empty test fixture in place.

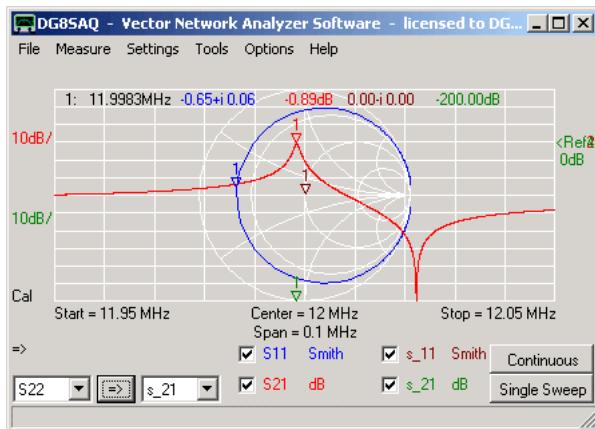


Any correction scheme other than the one shown above will produce wrong transmission measurement results! This is due to the fact that transmission and reflection do NOT share the same ground.

2. Perform a transmission (S21) and a reflection (S11) measurement of your crystal with one crystal terminal being input and the other being output.

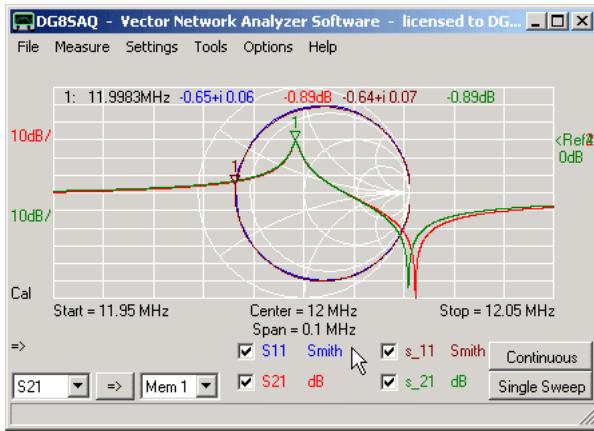


This is what the measurement result should look like for a 12 MHz crystal:



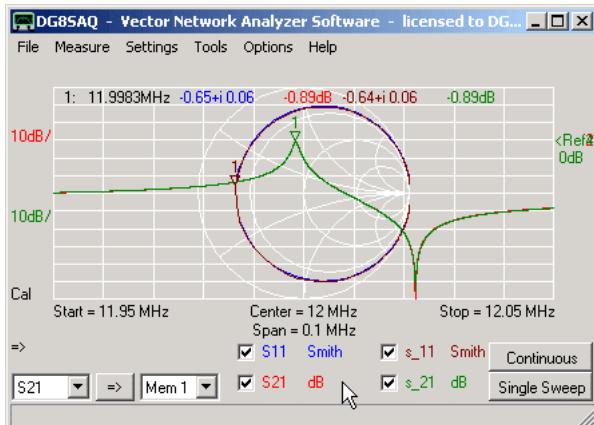
In transmission one can clearly see the serial resonance (marker) as well as the parallel antiresonance. Make sure, to display 3-port memory space s_{21} and s_{11} before moving to the next step. 3-port memory spaces s_{11} and s_{21} will be used to store and allow to display the model simulation in reflection and transmission.

3. Start the "Crystal Analyzer" tool via the main tools menu, select **source = S11** and press the **auto-optimize** button. When the tool is finished you will see this simulation result:



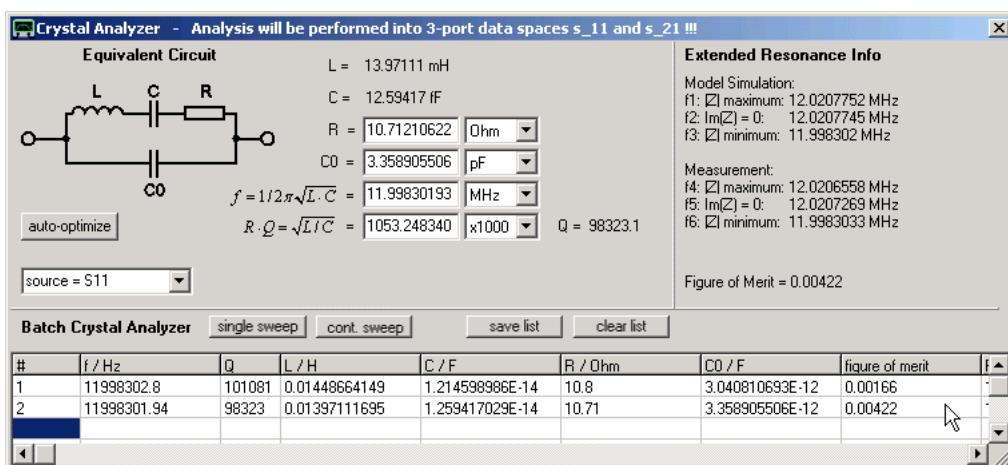
The model perfectly matches the reflection measurement S11. Model parameters can be read off the Crystal Tool window. In transmission (S21) there is a mismatch visible for the parallel antiresonance. Note, that the transmission simulation is mathematically derived from the reflection fit. The reason for this deviation in transmission will be discussed below.

4. Make sure that now **source = S21** and the correct **test fixture impedance** (must be identical on input and output) is selected before auto-optimizing. Now, the model simulation looks like this:



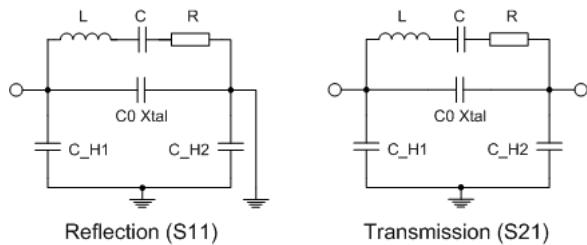
Now, the parallel antiresonance of the transmission (S21) measurement is perfectly modeled, while that of the reflection measurement is off (what you cannot really see in above Smith chart, though, since we missed to place a marker onto the parallel antiresonance).

Next, we compare the extracted model parameters of above two optimizations:



Above, line #1 shows the model parameters for the transmission (S21) fit, while line #2 shows the reflection (S11) fit parameters. The most apparent difference is that of the parallel capacitance C0. While in transmission C0=3.04pF, in

reflection $C_0=3.36\text{pF}$, which is 0.32pF higher. The increased parallel capacitance shifts the parallel antiresonance to lower frequencies. But what's the reason for this C_0 increase? It lies in the fact that the model for a real crystal is a bit more complex than assumed above:



As shown above, a real crystal consists of the crystal disk and a holder. The holder pins each have a capacitance to ground (C_H). While in transmission the series circuit of the holder capacitances add to the crystal parallel capacitance ($C_0\text{ Xtal}$), C_H2 is effectively shorted in a reflection measurement and C_H1 alone adds to $C_0\text{ Xtal}$ to yield an effective C_0 . Assuming that C_H1 and C_H2 are approximately identical, C_0 for a reflection measurement should be $0.5*C_H$ larger than for a transmission measurement. We can thus interpret our above experiment such that the holder capacitances are $C_H=2*0.32\text{pF}=0.64\text{pF}$ and $C_0\text{ Xtal}=3.04\text{pF}-0.32\text{pF}=2.72\text{pF}$.

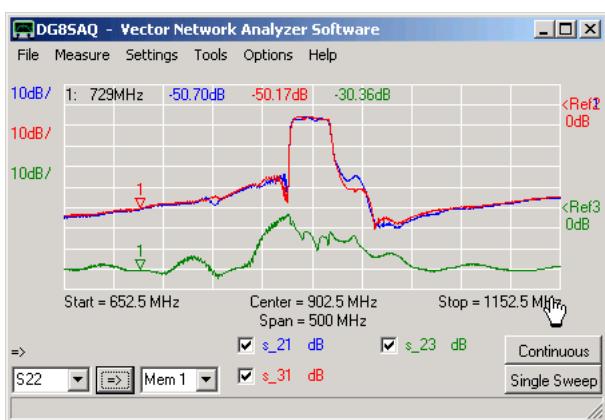
3-PORT ANALYZER: ANALYZING BALANCED COMPONENTS

Three port RF devices like baluns or SAW-filters with e.g. balanced outputs confront the user with a complex characterization task:

On the one hand, one would like to know e.g. the insertion attenuation from single ended input to differential outputs and the complex differential output impedance (**differential mode**). On the other hand, the **common mode attenuation** is of interest.

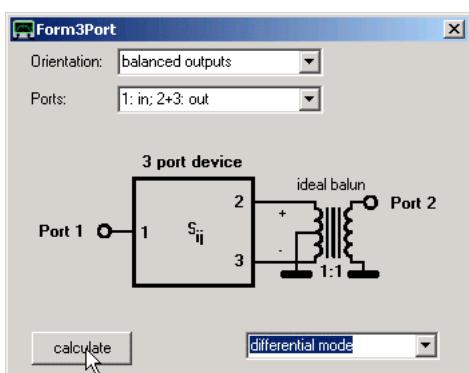
The 3-Port analyzer tool allows to perform both characterizations on a set of 3-port S-parameters. The following is a step by step example of performing a 3-port analysis:

1. Import or measure the 3-port S-Parameters of your DUT.:



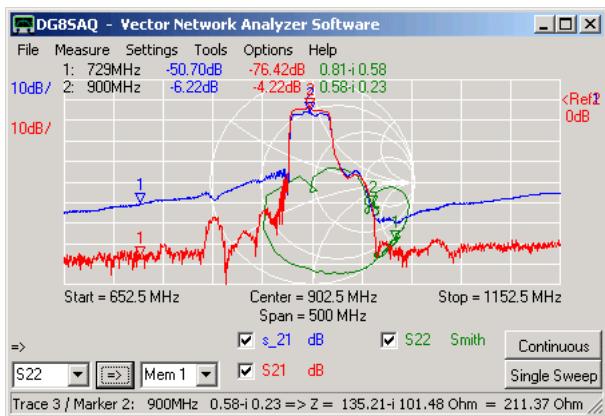
Note, that all signal paths from any DUT port to any other DUT port including all reflections must be characterized. The above imported 3-port S-parameters of a GSM RF filter are courtesy of EPCOS. Filter port 1 is the single ended input and ports 2 and 3 are the differential filter outputs. The above plot shows highly symmetric filter transmissions from port 1 to 2 and from port 1 to 3 (s_{12}, s_{13} , note the underscore!).

2. Start the 3-port analyzer



Select the desired mode of operation (1 = in, 2&3 = balanced out, **differential mode**) and press calculate. The result of this simulation will be stored as 2-port S-parameters into the 2-port memory spaces $S11, S21, S12$ and $S22$ and into the 2-port S-parameter buffer (useful for later matching analysis). This simulation is equivalent to combining the two output signals with an ideal 1:1 balun and thus turning the 3-port into a 2-port.

3. Observe the result in the main window.



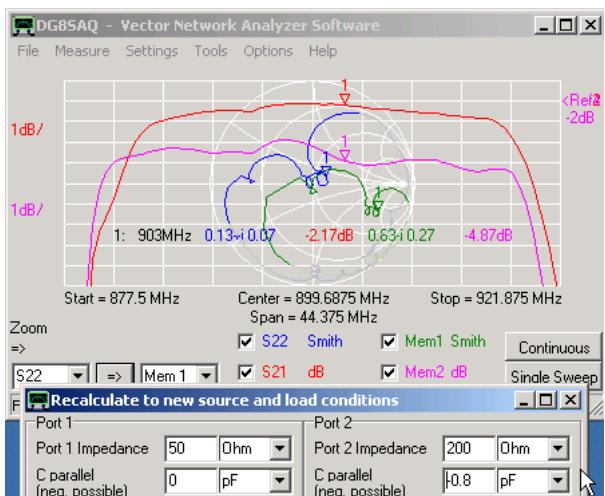
Note that the simulated 2-port insertion attenuation S21 is almost 3dB lower than the 3-port one s_21, because the signals of DUT ports 2 and 3 are being combined now.

Note, that the attenuation level improves dramatically, since the almost identical feedthrough signals 1-2 and 1-3 cancel each other at the balanced port.

Also note, that the differential output impedance (parallel equivalent circuit) of the filter is 200 Ohms (see bottom status line with marker info).

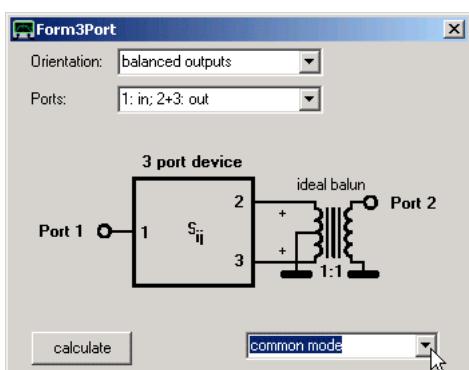
4. Perform a matching analysis

By using the matching tool, the simulated 2-port transfer characteristic can be recalculated to what it would look like if the filter was terminated with 200 Ohms instead of the 50 Ohms of the measurement system:

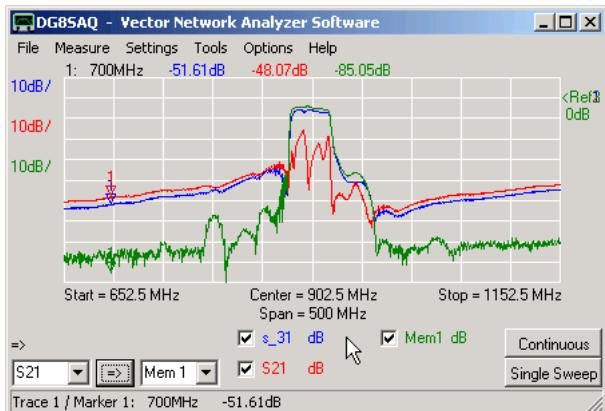


For comparison, the original 50 Ohm terminated data has been stored to Mem1 and Mem2 respectively. Note, how by proper termination the insertion attenuation can be further reduced.

5. Perform a common mode analysis:



Select the desired mode of operation (1 = in, 2&3 = balanced out, **common mode**) and press calculate. The result of this simulation will again be stored as 2-port S-parameters into the 2-port memory spaces S11,S21,S12 and S22 and into the 2-port S-parameter buffer. The common mode analysis is equivalent to connecting the normally differential outputs in parallel, thus turning the 3-port into a 2-port with the two output ports connected in parallel.



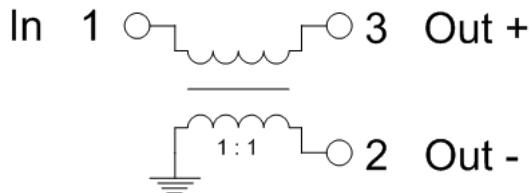
Note, that the electrical feedthrough of the common mode (S21) is 3dB higher than the s_31 signal of the original 3-port, since s_31 and s_21 are being added.

Also note, that there is signal cancellation in the filter passband now, as in the passband region s_21 and s_31 exhibit a 180° phase offset to each other.

For comparison, Mem1 shows the differential mode signal.

CHARACTERIZING A BALUN

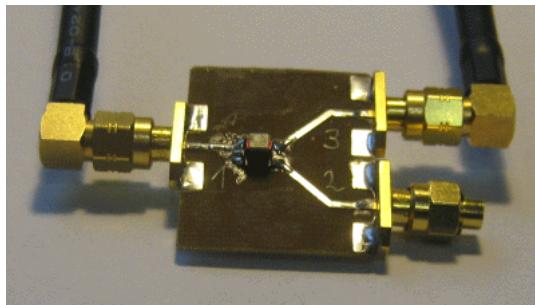
A BALUN is generally a passive device with 3 hot terminals and a ground terminal, which transforms an unbalanced signal relative to ground to a push-pull signal pair:



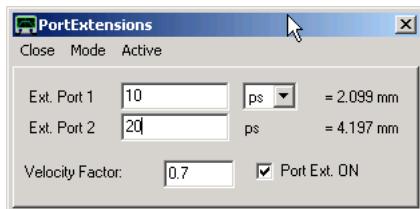
1:1 BALUN built from a 3dB coupler

Once a BALUN is built, the question arises how to characterize it, e.g. what is its impedance transformation ratio, insertion loss, balance and common mode suppression.

All these questions can be answered once the **3-port S-matrix** of the BALUN is known. This can be measured with the VNWA 3-port S-parameter acquisition menu. Before measuring the 3-port S-parameters of the BALUN, clearly mark the BALUN ports with **unique port numbers** (e.g. 1,2,3):

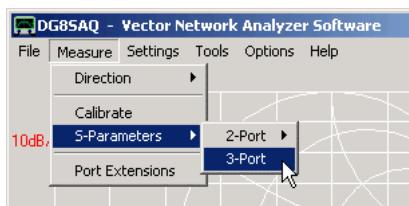


Next, after proper calibration of your VNWA (2-port SOLT calibration), you have to determine the electrical length of every line from the calibration plane to the BALUN terminal on the test board. To do so, switch on the port extensions:

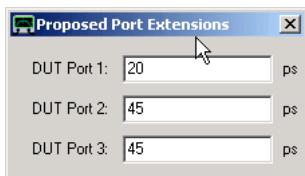


Next, measure the reflection coefficient of every port while generating a short circuit at the BALUN terminal and then adjust the port extension = electrical delay such, that the short shows up as a short in the Smith chart (dot on the left edge). Write down the values for all three ports as these will be needed for the 3-port measurement.

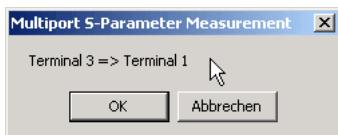
Next start a 3-port measurement:



A popup window will ask for the delays you have just determined. Enter those...



... and close the window. Now you will be asked to connect your 3-port DUT to the VNWA in various ways, e.g.

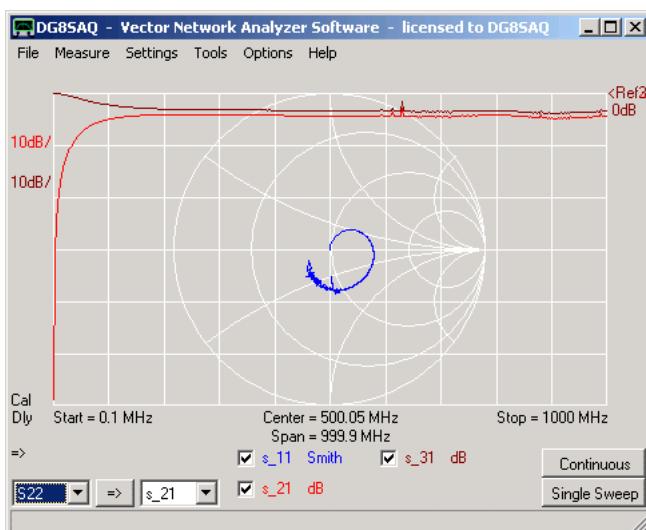


The above window asks to connect the VNWA TX port to DUT port 3 and the RX port to DUT port 1.

The unused DUT port (here port 2) must always be terminated with a 50 Ohms load.

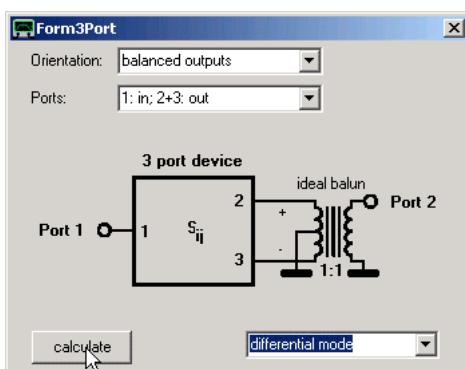
In the course of a 3-port measurement, you will have to measure from every port to every other port of your DUT which involves a lot of reconnecting. The reconnecting effort can be reduced by means of an S-parameter test set.

Once the measurement is done, the DUT 3-port S-matrix resides in the 3-port memory spaces s_{11} , s_{21}, \dots, s_{33} :

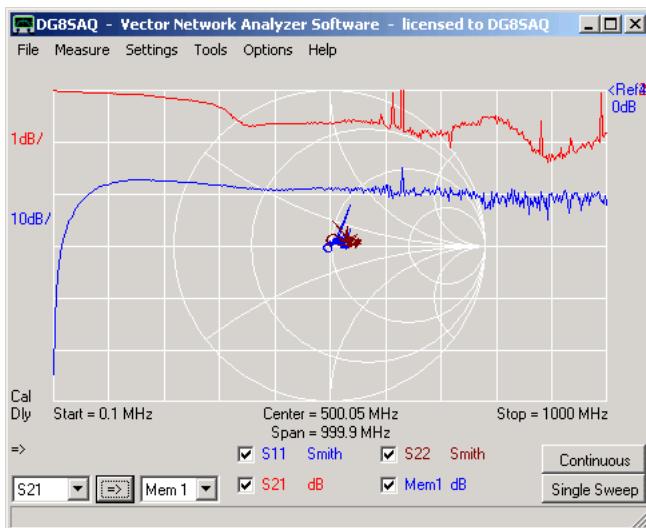


The S-matrix may be exported to an s3p file for later analysis.

In order to analyze the BALUN open the 3-port analyzer tool, select proper port topology (e.g. 1 in, 2+3 balanced out), select **differential mode** ...

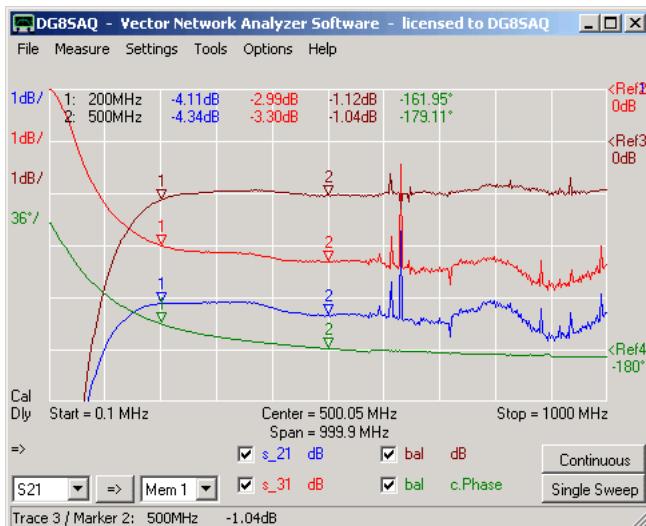


and press **calculate**. The tool will calculate the differential mode response and store in into memory spaces $S11$, $S21$, $S12$, $S22$:



It appears that the BALUN works with very low loss from 0.1MHz to 1GHz (see S21 trace) and there is an almost perfect match to 50 Ohms on both ports, the single ended input (S11 trace) and the balanced output (S22 trace). Mem1 in above screenshot contains the S21 result when choosing **common mode** instead of differential mode. This means our BALUN has a common mode suppression of around 20dB over the entire frequency span.

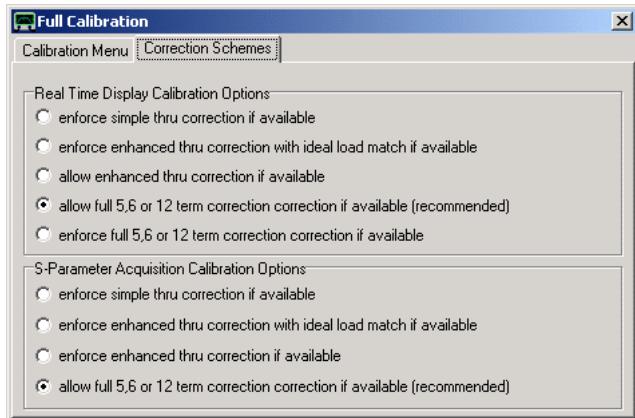
Now, does this mean our BALUN shows perfect BALUN action over the entire measured frequency span? No, we haven't looked at the balance yet, i.e. we haven't looked if the port 2 and port 3 signals actually exhibit identical amplitudes and a $+/-180^\circ$ phase shift. We can investigate this by comparing s_{21} and s_{31} either by inspection or by generating a custom trace with the function s_{21}/s_{31} as is shown below for the two custom traces named bal. For a perfect push-pull operation s_{21}/s_{31} should be -1, i.e. 0dB and -180° or $+180^\circ$ phase.



Here we see that even though, the BALUN exhibits a very low total loss down to 0.1MHz, it is only usable down to about 200MHz as below this frequency the amplitude and phase balance degrades rapidly.

ERROR CORRECTION MODELS

The VNWA software has implemented several error correction models, which can be selected in the **Correction Schemes** tab of the calibration window.



Real Time Display Calibration Options determine the error correction model to be used for **real time sweeps** continuously displayed in the main window.

S-Parameter Acquisition Calibration Options determine the error model used for the main menu command "measure" "2-port S-parameters".

►►► **Note:** For **one port** measurements these options are of **no importance**.

For **two port measurements**, the following **error correction models** are available:

- simple thru correction

The transmission data is simply corrected by division with the thru calibration measurement.

- enhanced thru correction with ideal load match

The transmission data is additionally approximately corrected for errors arising from the TX port not having 50 Ohms source impedance. Note, that a reflection calibration as well as the measurement of S11 is necessary to apply this correction to S21.

- enhanced thru correction

The transmission data is additionally approximately corrected for errors arising from the RX port not having 50 Ohms load impedance. Note, that a reflection calibration, a thru match calibration and the measurement of S11 are necessary to apply this correction to S21.

- 5,6 or 12 term correction

This is the best available error correction scheme. All transmission and reflection data are exactly corrected for imperfect TX source and RX load impedances. A full reflect calibration as well as a thru and a thru match calibration are required (5 term correction). If an isolation calibration is done, it will be extended to a 6 term correction. If using an S-parameter test set and both measurement directions are calibrated, it will be extended to a 12 term correction model.

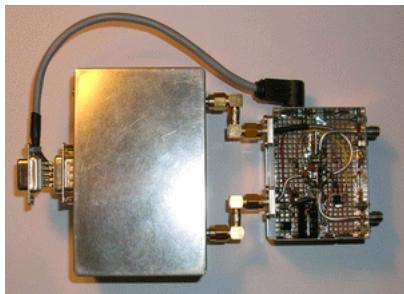
►►► **IMPORTANT:** In order to apply this error correction, all four S-parameters (S11,S21,S12,S22) must be measured.

- If "**allow full 5,6 or 12 term correction**" is selected, it is only applied if the user decides to display and thus measure all 4 S-parameters. Note, that VNWA usually measures only those S-parameters which are displayed in order to save measurement time.

- If "**enforce full 5,6 or 12 term correction**" is selected, VNWA will always measure all 4 S-parameters, even though they might not be displayed.

►►► **Note:** The term "**if available**" in above menu denotes the circumstance, that the error correction model might not be available if the user hasn't performed all necessary calibration measurements or if not all necessary S-parameters are measured. In such a case, the software will fall back onto the best *available* correction.

S-PARAMETER TEST SET



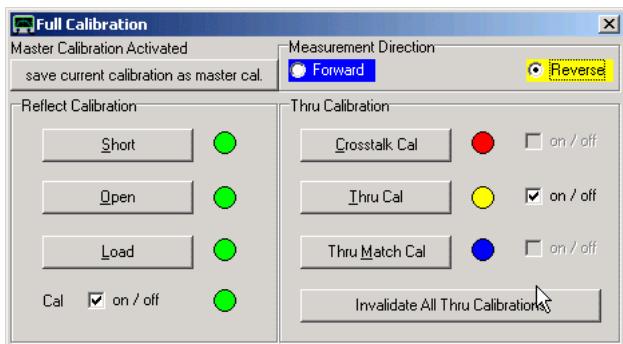
The VNWA software supports usage of an **S-parameter test set**. The test set contains several switches, which interchange output and input of the VNWA. This actually allows for automatic change of the signal flow direction, which is useful for measuring S-parameters of two port devices. In order to measure S12 and S22, the DUT doesn't need to be turned around manually any longer, but the test set simply exchanges input and output of the VNWA instead.

In LPT mode, the test set is controlled via the parallel printer port terminal **1/"Strobe"** of the Sub D-25 connector. In USB mode, the control signal for the S-parameter test set is available on the **USB_VNWA Interface** at pin 3 / J26 or alternatively pin 7 / J5 (Sub-D9) provided the appropriate resistors are in place.

Regardless of operation mode or controlled VNWA type, the test set may also be controlled by a dll plug-in provided by the user (VNWA36.2.zc or later). Details can be found on page "Controlling an S-parameter test set via dll plug-in".

Test set support is activated in the instrument settings tab of the "options"- "setup" main menu.

If activated, **two independent calibrations** for the two measurement directions will be required. This is indicated in the calibration menu window by the additional choice of "**Forward**" and "**Reverse**" measurement direction.



The two calibration directions are indicated by the colors blue (= forward) and yellow (= reverse). The calibration indicator lamps can show four colors:

- red:** no calibration done in any direction
- blue:** calibration done in forward direction only
- yellow:** calibration done in reverse direction only
- green:** calibration done in both directions

Using a test set will also allow convenient usage of a 12 term error correction model in realtime.

CONTROLLING AN S-PARAMETER TEST SET VIA DLL PLUG-IN

From software version VNWA36.2.zc on, there is implemented a customizable way to control an S-parameter test set via a dll plug-in provided by the user. Previously, the test set had to be connected to a specific LPT port pin or a specific pin of the VNWA USB interface. Now, the user can implement his own switch control, e.g. via a separate USB connection to the switch.

This is accomplished by copying a customized version of SwitchDLL.dll into the VNWA installation directory. The dll will automatically be detected upon program start, and, if detected, will be loaded and called every time the measurement direction changes.

SwitchDLL.dll Delphi example

Note, that the library is implemented with C-language calling convention.

```
library SwitchDLL;

uses
  Windows,
  SysUtils,
  Classes,
  Forms,
  ShellAPI;

$R *.res

procedure _Init; export; cdecl;      //...is being called when the dll is loaded upon program start
begin
  FreeConsole;
  AllocConsole;                  //open a console window for debugging
  writeln('SwitchDLL function "Init" called.');
end;

procedure _Close; export; cdecl;     //...is being called when the dll is unloaded upon program termination
begin
  FreeConsole;                  //close debug console
  //put your own cleanup code here
end;

procedure _Process(Data: Integer); export; cdecl;    //... is called every time the test set is operated
begin
  writeln('SwitchDLL function "Process" called.');
  writeln(Data);
  case Data of
    //replace the following by your own switching code
    0: ShellExecute(Application.Handle, 'open', PChar('testset.exe'), PChar('FWD'), nil, SW_NORMAL);
    //Data=0:Forward
    1: ShellExecute(Application.Handle, 'open', PChar('testset.exe'), PChar('REV'), nil, SW_NORMAL);
    //Data=1:Reverse
  end;
end;

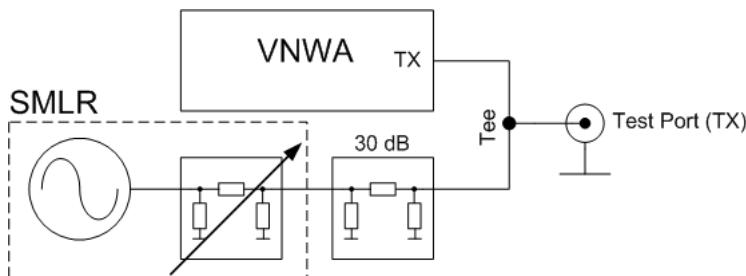
exports _Init;
exports _Close;
exports _Process;

begin
end.
```

IMMUNITY OF IMPEDANCE MEASUREMENTS TO INTERFERENCE

When measuring the impedance of an antenna, in many cases the antenna will pick up interfering signals from other signal sources like radio broadcasting stations, which can become quite strong.

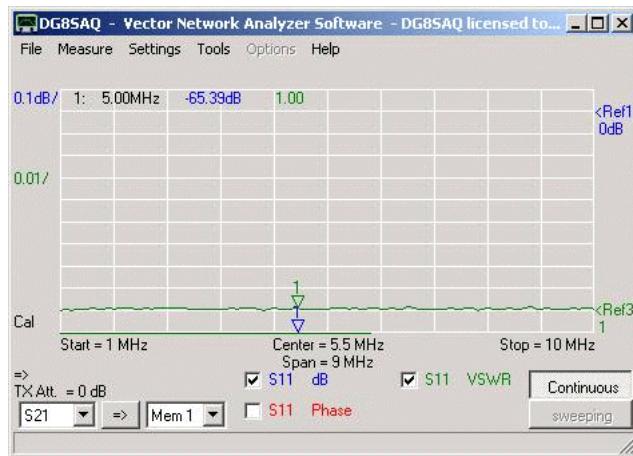
Now the question arises, how reliable a VNWA impedance measurement will be in the presence of such external interference signals. In order to mimic such a situation in a controlled lab experiment, I have come up with the following setup:



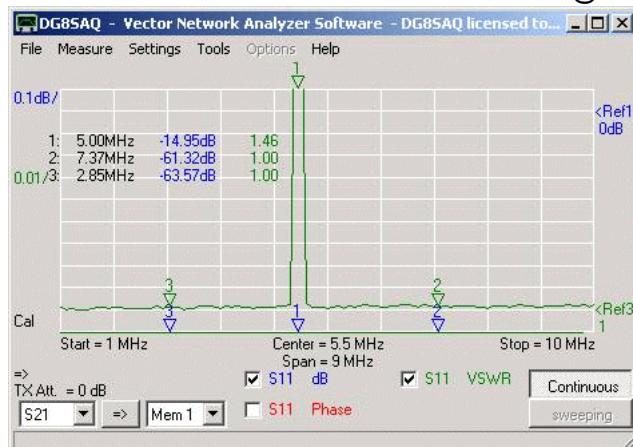
The interferer is simulated by a R&S SMLR signal generator capable of delivering up to 2W of RF power. The interferer signal is being further attenuated by a 30dB pad. VNWA and interferer are being combined in a coax Tee the open port of which serves as new TX test port with interference. This port can be calibrated the normal way. Now, a test impedance can be measured at various interferer levels to study the impact of the interferer on the measured impedance. A first experiment without the 30dB pad was unsuccessful, because instead of measuring the impact of different signal levels, the setup measured the changing port impedance of the SMLR at different attenuator settings.

Here are some test results with above setup:

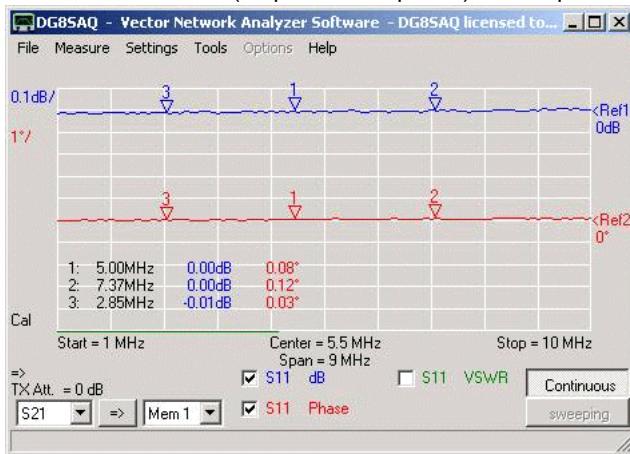
VSWR of the load cal standard without interferer:



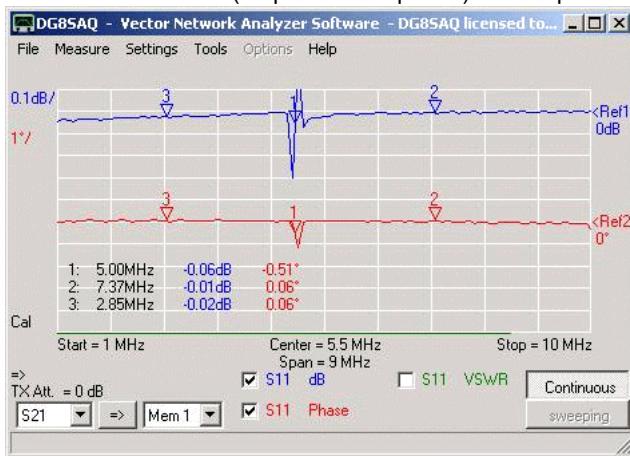
VSWR of the load standard at +3dBm interferer level @5MHz at the Test port.



Reflection coefficient (amplitude and phase) of the open cal standard without interferer.



Reflection coefficient (amplitude and phase) of the open cal standard at +3dBm interferer level @5MHz at the Test port.

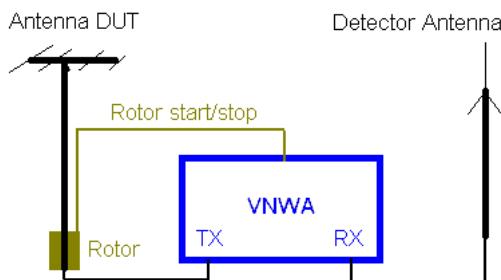


Summarizing, the effect of interference is relatively small. In order to test impact of interference at higher levels, a higher power signal source (e.g. 10W in order to measure at 10mW interferer level) would be required.

MEASURING ANTENNA RADIATION PATTERNS

This extension to the VNWA measurement capabilities evolved from experiments Erik ON8DC, who also tested the described features. It allows to measure antenna radiation patterns and display and analyze them with the VNWA software.

The following image shows the basic test setup:



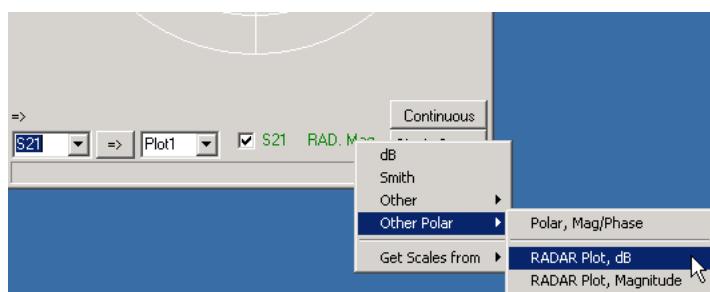
The VNWA TX output feeds the DUT antenna mounted on a rotor, while the RX input detects the radiation power in some distance. DUT antenna and detector antenna may be swapped.

The VNWA is set to a fixed frequency (span = zero). The VNWA sweep and the rotor engine are started simultaneously, such that the radiated power depending on turning angle is measured. The rotor start/stop signal is implemented in LPT mode, for VNWA2 from firmware version v4.12 on and for VNWA3. The rotor may also be started manually.

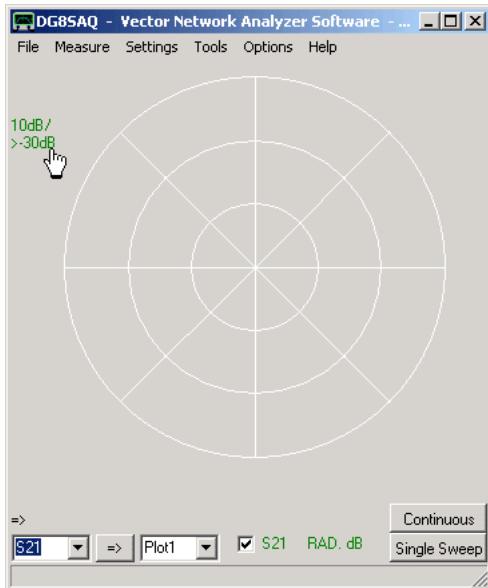
If the distance between DUT antenna and detector antenna becomes large (far field measurements), it is not feasible to feed the DUT antenna from the VNWA TX because of excessive cable length and/or insufficient TX power. In this case, a separate highly frequency stable cw-transmitter may be used to feed the DUT antenna. As the transmitter signal and the VNWA LO are not phase locked any longer, the measurements have to be performed in Spectrum Analyzer mode in this case.

The following shows how to set up the VNWA software for this measurement:

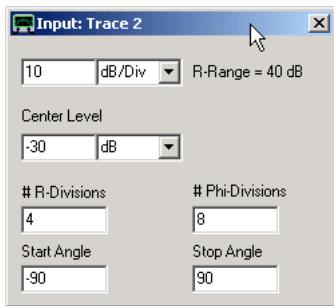
1. Set the proper measurement frequency (range) with span zero.
2. Select to display S21 in a polar RADAR diagram:



3. Set the display range to the desired values by doubleclicking the the label with the handpointer below. The displayed values indicate a center level of -30dB and 10dB per radial unit.

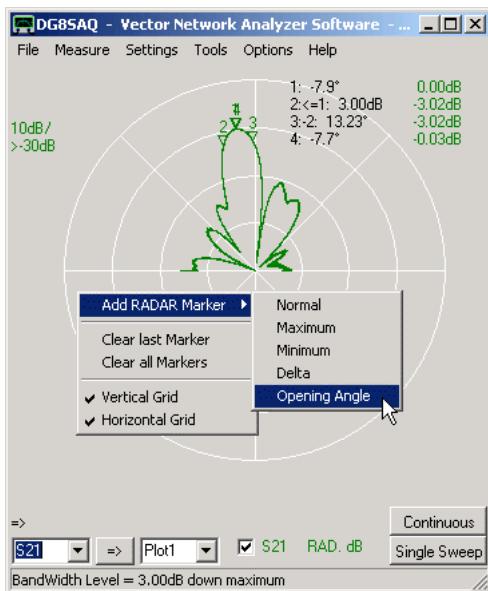


Upon doubleclicking the above mentioned label, the following parameter entry window pops up:



With the above settings the sweep data is interpreted as stretching from -90° to 90° . Note, that the angle settings can also be corrected after the measurement to account for angle shifts.

4. Perform your measurement by pressing **single sweep** and simultaneously starting the rotor engine. After the sweep you may add RADAR markers, e.g. to determine the antenna 3dB opening angle as seen below.



The above displayed data was provided by Erik ON8DC. It was measured on a 23cm band loop antenna at 1.25 GHz.

Here is Erik's description how he performed the measurement:

- DUT Ant needs a stable freq source.(It could be a generator, QRP TX) or if enough gain the TX output from the VNWA. Power is depending on distance and gain of antennas. (A far end test must be possible)

- At RX side use an antenna connected to RX VNWA. Again, depending on distance and power used at TX side, a gain antenna may be needed.

- Set VNWA to SA mode.

- Verify with a sweep (S21) if you can receive the DUT. Lower the span to lowest value 0Hz. The RX is a receiver with a narrow filter.

- Adjust the level offset in the sweep menu such that your RX receives 0dB at max position.

- Verify that in the min signal received you are above noise level of SA. (linearity verify with attenuators)

- Measure the time your antenna rotor needs for a full turn and adjust sweep-time until sweep time=rotor time.

- Select S21 and S21 RAD dB (other polar) as traces.

- Start manually or automatically synchronic the rotor and the sweep. You will see now the signal variation with changing angle at the screen.

For the antenna plot you can change the trace parameters start/stop angle divisions and center level.

- For automatic rotor start you can use the high signal on D7 Lpt (pin9) to drive a relay, it stays on until the sweep is finished. (LPT mode was used here)

VERY LOW FREQUENCY MEASUREMENTS

Improved Very Low Frequency Signal Processing Facility

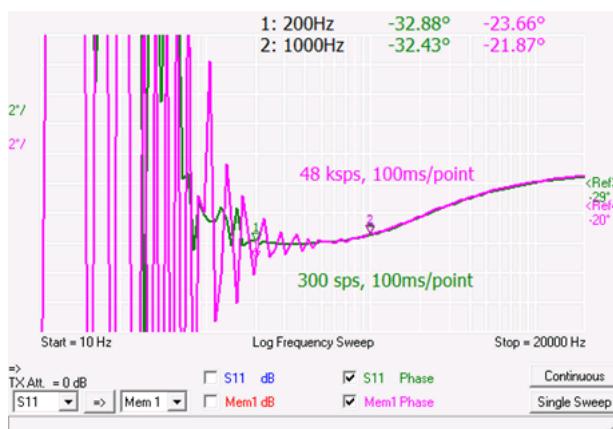
VNWA software version 36.8.1 and newer has implemented an improved signal processing algorithm to improve measurement results at very low frequencies.

As described below, for measurement frequencies below the sound codec cut-off frequency, the test signal contains a messy spectrum of desired and undesired signals. The new algorithm implements digital filtering to greatly improve the signal-to-noise ratio at such low frequencies.

Prior to this, it was recommended to use the codec anti-aliasing filter to filter out some of the unwanted signals by drastically reducing the sample rate. This is no longer necessary.

Now, the standard settings can be maintained down to the lowest measurement frequencies.

The following figure shows a comparison when measuring the uncalibrated phase response of the open TX port using the old method at reduced sample rate (pink) and the new digital filtering at full sample rate (green), both with otherwise identical settings, i.e. sweep time.

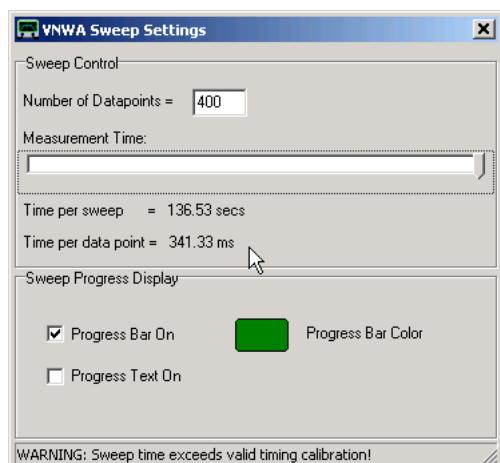


While the measurement with the old method starts getting noisy below 1kHz, the new method grants clean measurement data down to 200Hz!

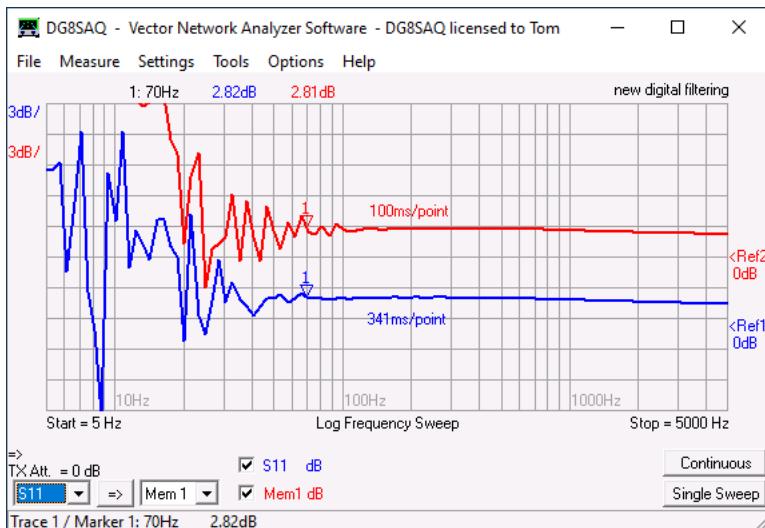
Now, there are two factors setting the lower frequency limit of the VNWA:

1. Measurement time per frequency point: For appropriate filtering at least two signal periods must be captured. Thus, at a sweep time of 100ms/point 20Hz is a hard limit.
2. Due to the ac-coupled signal chain the signal power drops quickly as the measurement frequency gets lower.

The new software addresses 1. by increasing the permissible measurement time to the maximum of about 340ms which the VNWA firmware allows:

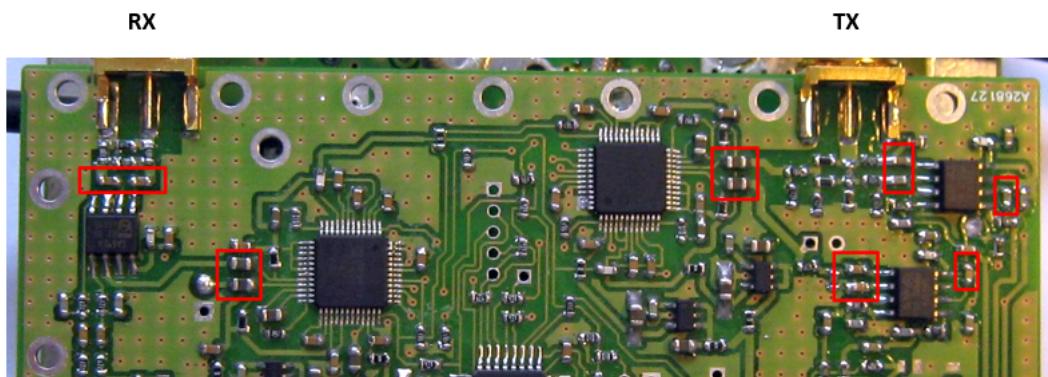


This lowers the theoretical signal processing limit to 6Hz. This cannot fully be exploited with a standard VNWA due to the strong drop in signal levels, though:

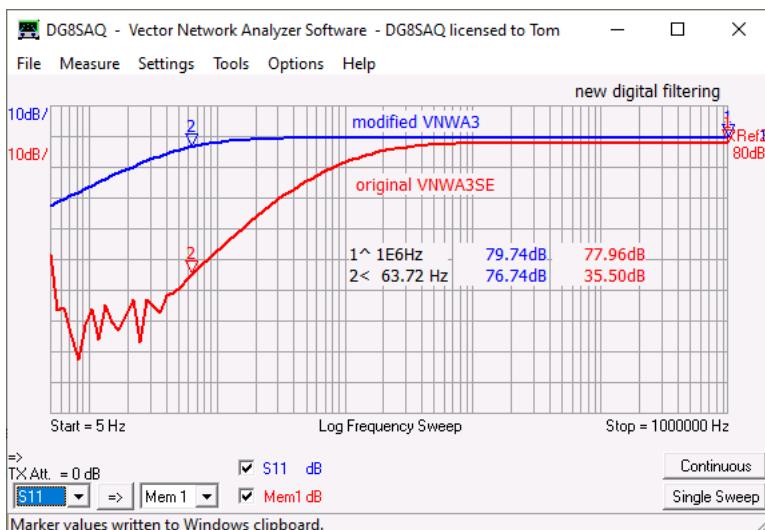


With a standard VNWA the new settings allow measurements down to about 70Hz as compared to about 1kHz with the older software.

Now, since the design of the VNWA, properties of passive components have improved considerably. While the VNWA uses 0805 coupling capacitors of 1uF and 0603 coupling capacitors of 220nF, today 0603 capacitors are available with up to 47uF. So, I have replaced all relevant coupling capacitors on a VNWA3 (12 caps, marked with red boxes below) by these new 47uF ones:

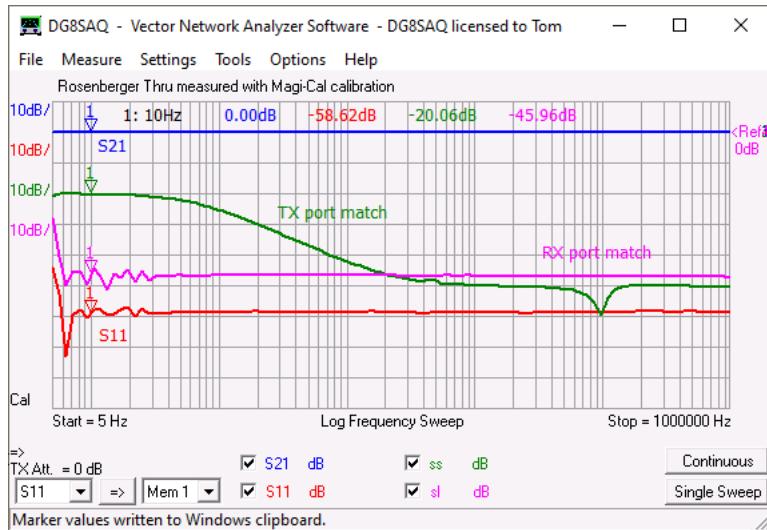


This considerably improves the low frequency signal level, which VNWA can directly measure when normalization is de-activated:



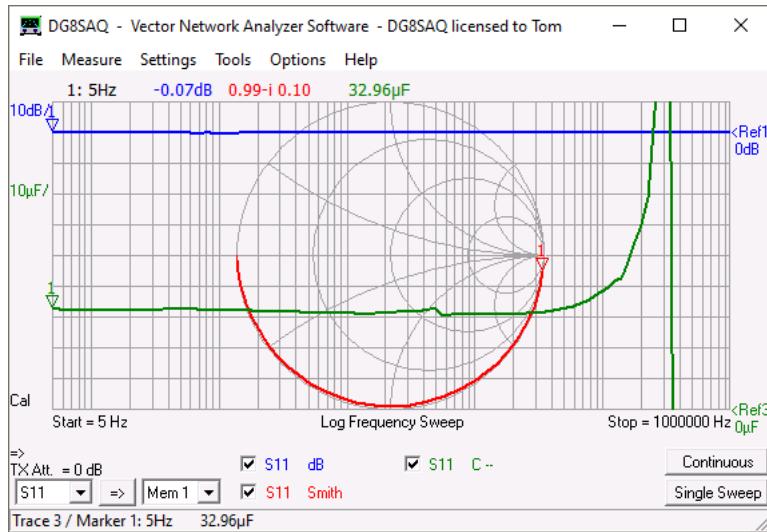
The modification decreases the -3dB signal drop frequency from about 2kHz to 60Hz!

Now, the modified VNWA is calibrated with the Magi-Cal automatic calibrator to determine test port impedances and the quality of the Magi-Cal calibration at very low frequencies. The latter can be determined by performing a full two port measurement of a quality SMA thru barrel, which should yield 0dB insertion loss and a reflection loss near the noise floor.

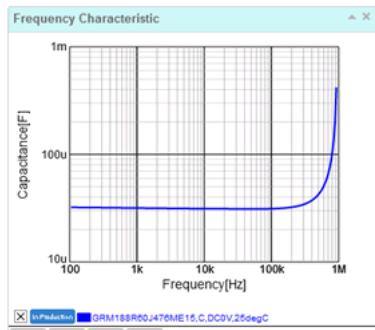


Indeed, S21 shows at 0dB and S11 around -60dB, which proves that the Magi-Cal does an excellent job down to 5Hz. The RX port match is below -45dB down to very low frequencies. This is not surprising as the RX port is dc coupled. The TX port match increases from below -45dB at high frequencies to -20dBm at the lowest frequencies, which is still an excellent match.

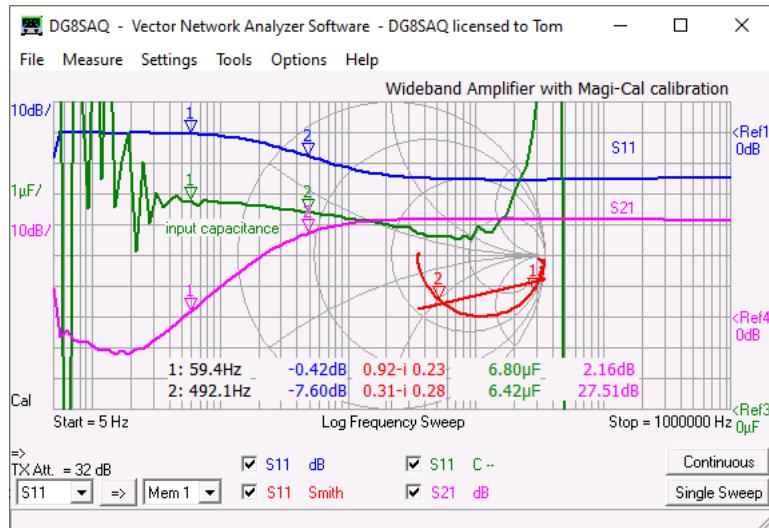
Now, the modified VNWA calibrated with the Magi-Cal automatic calibrator is used to measure one of the 0603 47uF capacitors used in the modification:



The capacitor could be measured down to 5Hz! It shows a rather constant capacitance of 33uF up to about 30kHz. This is not a measurement error but fact. Data on the Murata web page also shows this 47uF capacitor having 33uF, which is actually outside the specified +/-20% tolerance window.



Also active devices like amplifiers can be measured to very low frequencies. **Here is an example measurement of a wideband amplifier with about 30dB gain:**



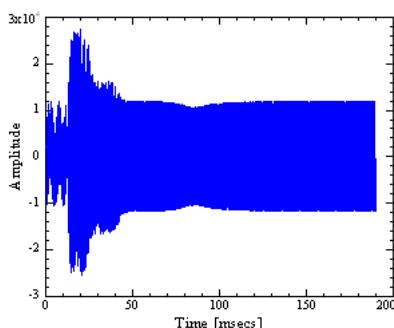
Here, the **output power has been reduced by 32dB** in order to not overload the amplifier or the VNWA RX port. Still, S11 can be measured reasonably well at this very low signal level. From S11 one can assume that the amplifier's dc blocking capacitor in series with the 50 Ohms input is 6,8uF, see green trace.

Increased Signal Amplitudes at Low Frequencies

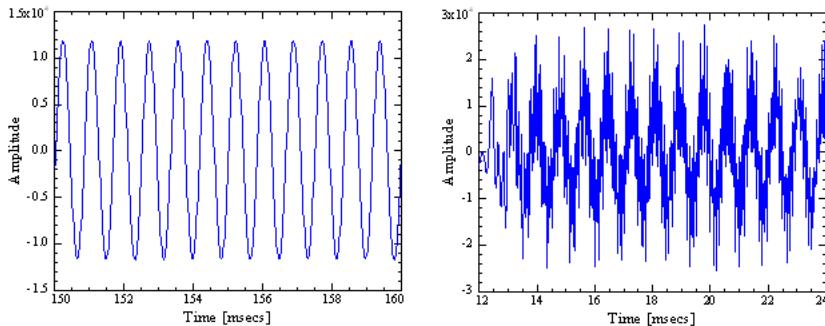
Quite against intuition, overall signal amplitudes may rise when the VNWA is used at audio frequencies **below 20 kHz**.

The reason is that the mixers used show some "RF" feed through from the RF input to the IF output and also from the LO input to the IF output. Thus, when measuring in the audio range the RF feedthrough signal will be superimposed on the IF signal leading to **signal distortion** and possible **ADC overloads** as it falls inside the sound card bandwidth.

The following image shows the raw audio stream captured with a 16 bit audio card for a sweep from 1 kHz to 100 kHz. Note, that for a 16 bit card the maximum amplitude range is -32767...32768.



For frequencies outside the sound card bandwidth (>20 kHz, beyond 50 msecs in the plot) the amplitude of the sound stream is about 50% of the allowed maximum amplitude. This will also be seen with the Test Audio function, as it performs the test in the MHz range. At low frequencies the amplitude approximately doubles with the risk of overloading the sound ADC. The mechanism leading to this amplitude increase can be seen when zooming into a high frequency and a low frequency region for comparison:



The left image shows a clean IF signal corresponding to a 100 kHz input frequency, while the right image shows an apparently "noisy" IF signal corresponding to an input frequency in the 1kHz range. Here, the superimposed "noise" is actually a superposition of RF, LO and all kinds of mixing products.

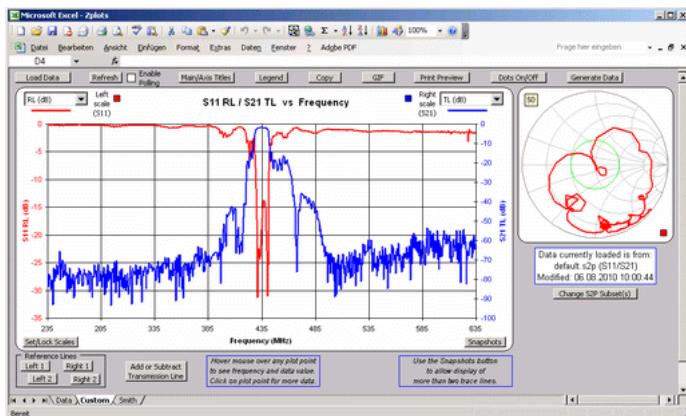
If the instrument signals an overload condition, the simple cure is to **manually decrease the codec capturing levels** by some 10%.

Summary:

- With the new software release the standard VNWA can work down to **60 Hz**.
- Replacing 12 capacitors enables the thus modified VNWA to operate down to **5Hz**, thus enabling the VNWA to also cover the full audio frequency range.
- No special settings are required for very low frequency coverage other than allowing sufficient measuring time.
- The Magi-Cal automatic calibrator reliably works down to 5Hz and probably below, as it is dc coupled.

INTERFACING ZPLOTS

Dan, AC6LA, has written **Zplots** (<http://ac6la.com/zplots.html>), a wonderful Excel application, that uses the power of the Excel charting engine to neatly plot and analyze S-parameters like the ones generated during a VNWA measurement.



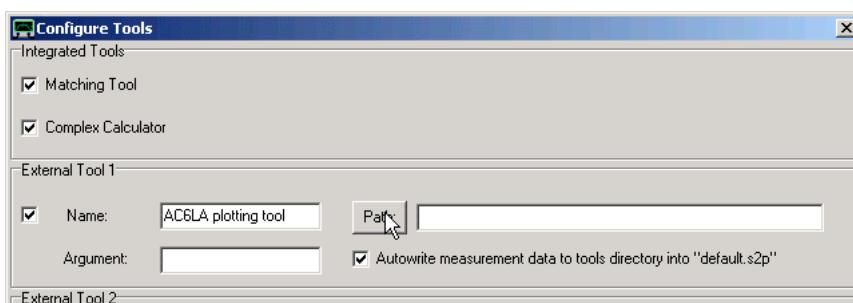
Many thanks to Dan for having modified ZPlots such that it can be easily integrated into the VNWA software.

The following interactions between VNWA and Zplots are available:

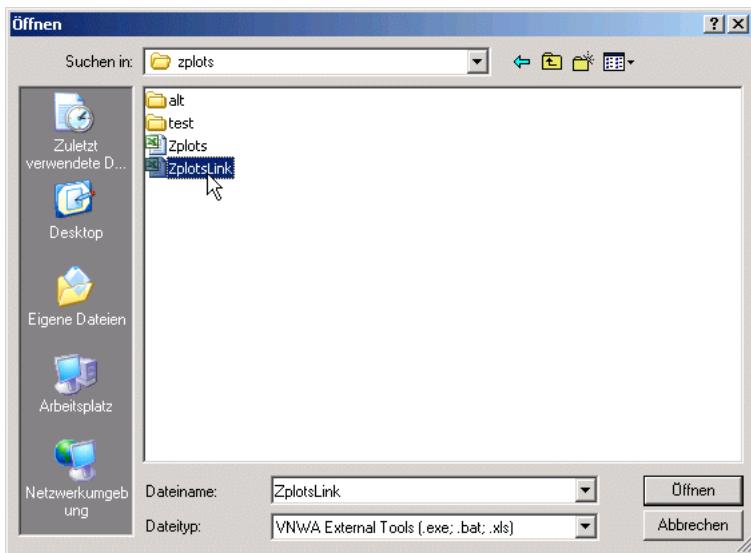
- Zplots can be started or reactivated as external tool from within VNWA.
- Upon (re-)activation of Zplots through the tools menu, VNWA can automatically transfer measurement data to Zplots.
- Zplots can automatically poll for new measurement data, that VNWA can continuously provide after each completed sweep.

Configuring VNWA for interfacing Zplots:

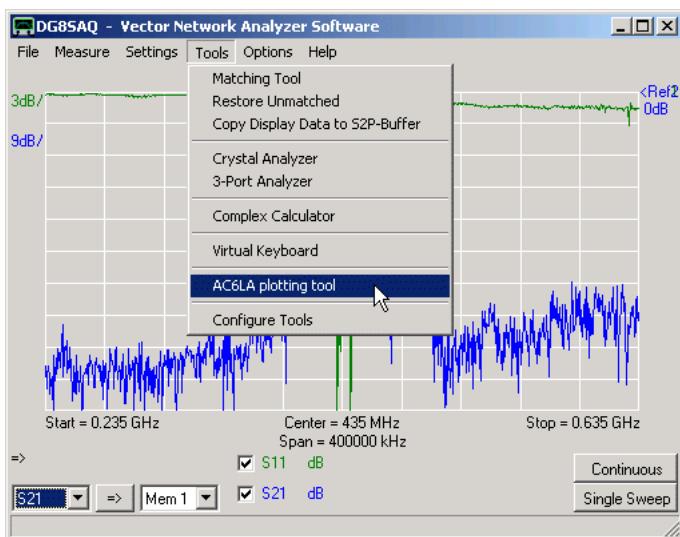
1. Copy the latest version of Zplots to any directory of your choice. The Zplots package consists of two files, namely the application **Zplots.xls** and the interface **ZplotsLink.xls**. Both files need to reside in the same directory.
2. In VNWA open the "Tools"->"Configure Tools" menu, activate an external tool, give it a descriptive name e.g. "AC6LA plotting tool" and select "autowrite measurement data..." in order to give Zplots access to the VNWA measurement data. The "Argument" field should remain blank.



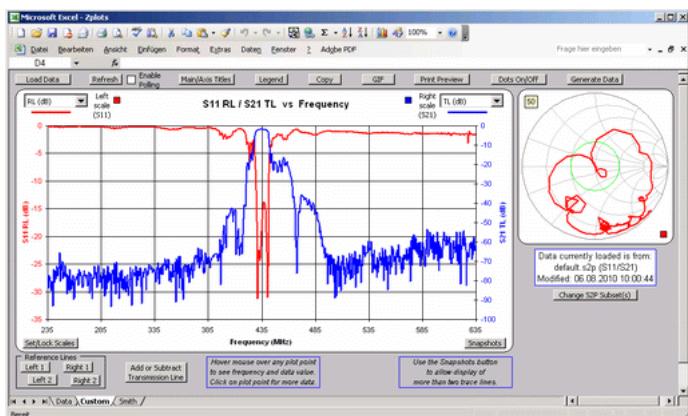
3. Next browse for the Zplots application by pressing the "Path" button.
Note: You must select the interface file **ZplotsLink.xls** and open it.



4. Close the "Configure Tools" menu. Observe, that your new Zplots tool is now visible in the VNWA "Tools" menu as "AC6LA plotting tool".



Selecting the just generated new tool menu "AC6LA plotting tool" will open Zplots and transfer the current measurement data to Zplots. If Zplots has previously been started already, only new data will be transferred:



For more information on Zplots, please consult the author's documentation, e.g. on <http://ac6la.com/zplots.html>.

REMOTE-CONTROLLING THE VNWA

For automated measurements the VNWA software can be remote-controlled via **TCP/IP**, **Windows messages** or via a **script file**. Under normal conditions the remote control interface in VNWA is disabled. In order to enable it, the VNWA software must be started with certain runtime arguments:

| Starting VNWA for acting as a TCP/IP web server:

Syntax:

VNWA.exe -tcpip [port number] [-silent] [-debug]

[...] is an optional argument, multiple arguments must be separated with blank characters.

-tcpip [port number] activates the VNWA tcpip web servers listening to the optional port number and two subsequent ports. If no port number is specified, the default port number 55555 is used.

-silent makes the VNWA main Window invisible. Note, that this will also hide the VNWA application from the task bar and the task manager application tab.

-debug opens a debug Window (also in silent mode), where all incoming remote commands are being listed.

-csweep opens VNWA immediately sweeping in continuous mode. **Note: If -csweep is the first command in the argument list, all subsequent commands will be ignored.** This option can also be used without the -tcpip option.

Example:

VNWA.exe -tcpip 54321 -debug

VNWA will be started as TCP/IP server listening to ports 54321, 54322 and 54323 with the main window and debug window visible.

For details on the remote control interface see page Controlling the VNWA via TCP/IP.

| Starting VNWA for accepting Windows messaging commands:

Syntax:

VNWA.exe -remote -callback ownhandle ownmessage [-silent] [-debug]

[...] is an optional argument, multiple arguments must be separated with blank characters.

-remote activates the VNWA remote control interface

-callback allows to specify a handle and a message number which VNWA will use to return its own handle and remote control message number and notify end of processing of a remote command via a Windows message.

-silent makes the VNWA main Window invisible. Note, that this will also hide the VNWA application from the task bar and the task manager application tab.

-debug opens a debug Window (also in silent mode), where all incoming remote commands are being listed.

-csweep opens VNWA immediately sweeping in continuous mode. **Note: If -csweep is the first command in the argument list, all subsequent commands will be ignored.** This option can also be used without the -remote option.

Example:

VNWA.exe -remote -callback 2622836 1024 -debug

VNWA will be started in remote control mode with the main window and debug window visible. The ownhandle and ownmessage parameters must be generated at runtime by the calling program and might change from program run to program run.

Upon successful launch VNWA will send a Windows message back to the handle of the calling program passed to VNWA (ownhandle). This message contains the handle of the VNWA application, the message number VNWA listens to, and, in case of errors, an optional error code.

For details on the remote control interface see page Controlling the VNWA by Windows Messages.

| Starting VNWA for processing a script file:

Syntax:

VNWA.exe filename.scr [-silent] [-debug]

[...] is an optional argument, multiple arguments must be separated with blank characters.

filename.scr is the name of the script file to be processed. Note, that the file ending MUST be **.scr**.
-silent makes the VNWA main Window invisible. Note, that this will also hide the VNWA application from the task bar and the task manager application tab.
-debug opens a debug (also in silent mode) Window, where all processed commands and error messages are being listed.

Example:

VNWA.exe test.scr -debug opens the VNWA main window and the debug window and executes the script file test.scr.

For details on the script file syntax see page Controlling the VNWA by a Script File.

CONTROLLING THE VNWA VIA TCP/IP

Terms:

The **SERVER** is the software directly connected to the VNWA hardware via USB. It is running on the same machine that the VNWA hardware is physically connected to. The server runs on Windows or inside a WINE environment on Linux.

The **CLIENT** is the software that is talking to the server and thus remote-controlling the VNWA. As communication is achieved via TCP/IP, the server may reside on a different machine. Communication should even be possible through wireless LAN or through the internet. The client can run on any platform.

The VNWA TCP/IP Interface Architecture:

Note, that the VNWA TCP/IP interface actually consists of three independent servers with distinct port numbers. The base port number PORT is either the default port 55555 or the user defined port number specified in the VNWA runtime argument.

Server 1 at PORT:

This is the control server which receives remote control commands and returns error messages or status information. Only one client is allowed to connect to this server at any given moment.

Commands are plain text ASCII text messages. Subsequent commands are separated by the NULL character (0x00). Command syntax is explained below.

Server2 at PORT+1:

This is the data port which streams measurement data in real time to a data client as it is measured. Any number of data clients are allowed to connect to this server simultaneously.

Port datagrams are plain ASCII text messages separated by the NULL character (0x00). For the structure of datagrams see below.

Server3 at PORT+2:

This is the file exchange server which allows to upload files to or download files from the server machine. This is useful to download measured files or to upload calibration files or instrument states.

The syntax will be detailed below.

Proofs of concept of this interface have been realized as virtual VNWA devices and are described on pages VNWA Data Client (using only server 2) and VNWA RC Client (using all servers).

Launching the TCP Servers

There are several ways to start the TCP servers.

1. Upon program start using run-time arguments:

Launch VNWA using the runtime argument -tcpip, e.g.

`VNWA.exe -tcpip [base port number] [-debug]`

This will launch all three servers. If the optional base port number (e.g. 54321) is omitted, Server 1 will use port 55555, Server 2 will use port 55556 and Server 3 will use port 55557.

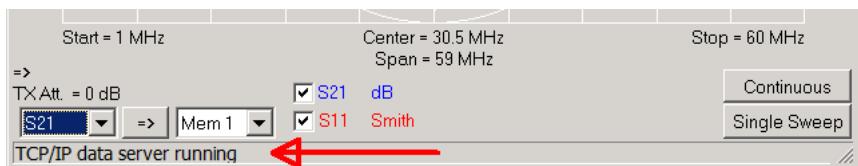
When the optional -debug option is used, an additional debug window will open together with the VNWA application, that allows to monitor server traffic and switch servers off or on manually.

2. During run-time using keyboard short-cuts:

Operating the following keyboard short-cuts on the VNWA main window will switch the indicated servers on or off in a toggle style:

`ctrl+alt+s` *TCP/IP data server 2 on/off*
`ctrl+alt+t:` *TCP/IP RC servers 1+2+3 on/off*

The execution will be prompted on the bottom VNWA status line:

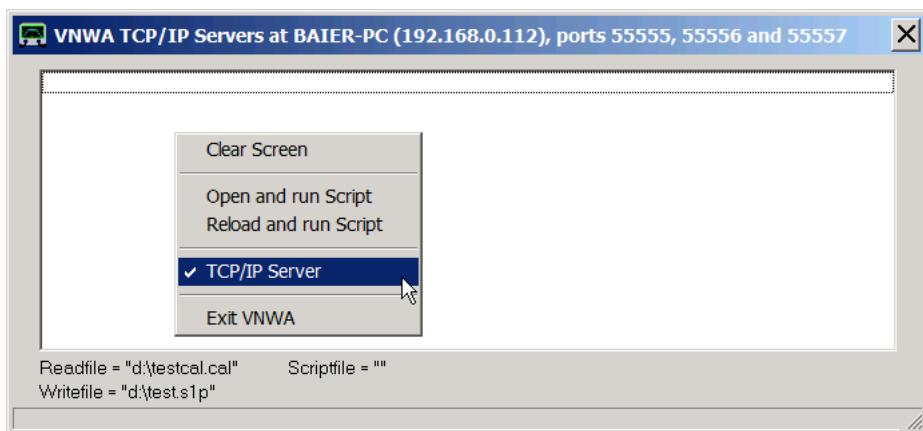


3. During run-time using the debug window:

The remote control debug window may also be launched manually via keyboard short-cut:

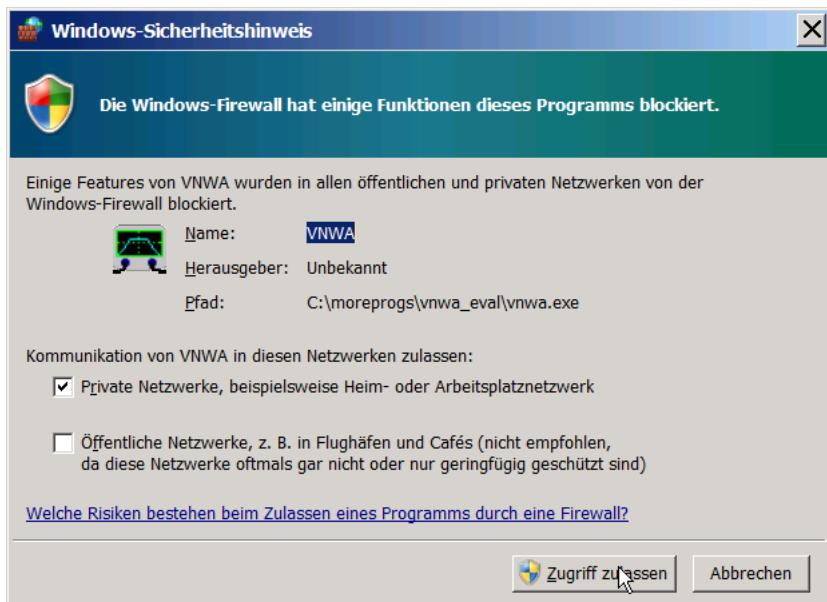
ctrl+alt+r open RC debug window

There, the servers can be switched on or off by right-clicking the white list area and checking or un-checking the TCP/IP server by clicking on the corresponding menu item:



Note, that in above screenshot, the top blue title bar indicates the server IP address and port numbers, which may be useful information when connecting clients, particularly from different machines on the network.

Upon launching a server for the first time, the Windows firewall might ask for permission, which you must grant, otherwise connectivity will be blocked by the firewall:



Implemented VNWA network commands:

autosetupaudio <r> performs an autosetup of the VNWA2/3/3E/3SE audio devices. This will invalidate the sample rate calibration which is normally no problem.
Parameter r determines if a subsequent sample rate calibration is performed (r=1) or not (r=0).

cadir <direction> set calibration direction (if S-parameter test set available). Direction = forward / reverse.

calsweep <calibration> do a calibration sweep. Calibration = Open, Short, Load, Crosstalk, Thru, ThruMatch

exitVNWA terminates the VNWA software

frame <n> <sweeptype> sets the sweep frame data. n=number of data points, sweeptype=lin or sweeptype=log sets a linear or logarithmic frequency grid.

hardware? Queries the hardware connected to the server.

installation_folder? Queries the remote installation folder path

vnwa_serial? Queries the VNWA serial, will return "none" if not licensed or not connected.

loadmastercal <cal-filename> load a calibration file as master-calibration

loadcal <cal-filename> load a calibration file

savecal <cal-filename> [comment] save a calibration file, optional comment string may contain blank characters

loadcalkit <calkit-filename> load a calibration kit file

timeperpoint <t> sets the measurement time per data point. The number t denotes the time per data point in milliseconds (VNWA only)

range <start> <stop> sets the measurement frequency range. start=start frequency [Hz], stop=stop frequency [Hz]

readsnp filename <n> <d> reads Touchstone data (n=1:s1p, n=2:s2p, n=3:s3p) to data space d (0=S21, 1=S11,..., 7=Mem4)

runscript <filename> run a script file as described on page Controlling the VNWA by a Script File

setgrid <sweeptype> *f1 g1 n1* [*f2 g2 n2*] ... sets the frequency grid. sweeptype=LIN, LOG, LIST. fi,gi=start/stop frequency ni=#points of segment i. For LIN/LOG there is only one frequency segment.

setTXpower <p> sets the TX DDS amplitude to p = 0...16383

sg <state> sets signal generator mode: state = on/off/disable

sg_lo <state> sets signal generator local oscillator: state = on/off

sg_tr <state> sets signal generator, thru/reflect switch: state = thru/reflect

sg_freq <frequency> sets signal generator: frequency in MHz

sg_mod <type> sets signal generator: modulation type = none, am, fm, pulse

sg_mod_freq <frequency> sets signal generator: modulation frequency in Hz

sg_mod_am_depth <depth> sets signal generator: am depth in %

sg_mod_fm_dev <deviation> sets signal generator: fm deviation in kHz

sg_atten <attenuation> sets signal generator: level attenuation in dB, allowed range -50dB...0dB

stop Immediately stop a running sweep

sweep [s11] [s21] [s12] [s22] starts a sweep for the specified data fields

writes1p <filename> <trace> writes measurement data of trace to an s1p Touchstone-file. trace can be any of s11, s21, s12, s22

writes2p <filename> writes the full measurement data (s11, s21, s12 and s22) to an s2p Touchstone-file

write-cust-expr <expr.> <filename> writes the result of the mathematical expression to an s1p Touchstone-file. Custom trace Cust6 is used and thus overwritten.

magical <standard> switch in Magi-Cal standard. Standard = open, short, load, thru

download-magi-cal-models download the calibration standard models from the connected magi-cal

WARNING: Blank characters in file names must be masked as %20 since blanks in command strings serve as argument separators!

Example sequence and server responses:

client sends (explanation)
server answers

hardware? (check if and what hardware is online)
online 7 VNWA3E (VNWA3E online, capabilities 7, see below)
setgrid lin 1e6 2e6 3 (set linear frequency grid, 1...2MHz, 3 frequency points)
Error Code: 0 (no errors)
timeperpoint 4.5 (set time per frequency point to 4.5ms)
Error Code: 0 (no errors)
sweep S21 S11 (do a single S21 and S11 sweep. Note that cal/sweep behaves likewise)
Error Code: 0 (no errors)
.... (no communication during sweep)
sweep_complete (sweep is over, data available)

- ▶▶▶ **Note:**All commands are terminated by the NULL character (0x00).
- ▶▶▶ **Note:**Commands are not case sensitive.
- ▶▶▶ **Note:**Error code 0 means OK, all other error codes indicate that remote command has failed..
- ▶▶▶ **Note:**VNWA capabilities consist of 4 bits:
 - Bit 1: can measure S11
 - Bit 2: can measure S21
 - Bit 3: can measure S11 and S21 simultaneously
 - Bit 4: has automatic S-parameter testset
- ▶▶▶ **Note:**Client must read the server response BEFORE accessing another server.

The VNWA TCP/IP Data Structure (Server 2)

Typical data stream of a linear sweep with 3 points from 1MHz to 2MHz:

```
sweep_start      3 1.000000E+006 2.000000E+006 1
data    0 1.000000E+006 9.999131E-001 -2.236229E-003 7.087993E-006 9.944799E-006 0.000000E+000
0.000000E+000 0.000000E+000 0.000000E+000
data    1 1.500000E+006 9.999723E-001 -3.267469E-003 7.562893E-005 -9.409467E-005 0.000000E+000
0.000000E+000 0.000000E+000 0.000000E+000
data    2 2.000000E+006 9.999005E-001 -4.436659E-003 5.288430E-005 -3.454693E-005 0.000000E+000
0.000000E+000 0.000000E+000 0.000000E+000
sweep_complete 0
```

- ▶▶▶ **Note:**All datagram lines are terminated by the NULL character (0x00).

Explanation:

sweep_start indicates beginning of sweep data, it also indicates the frequency grid. The numbers following **sweep_start** denote:

3 = total number of points

1e6 = start frequency

2e6 = stop frequency

1 = sweep type LIN (2=LOG, 3=LIST).

For the first sweep in a sweep sequence (S11, S21, S22, S12), the list of sub-sweeps is added to this number:

0x10 added => S11 will be measured

0x20 added => S21 will be measured

0x40 added => S22 will be measured

0x80 added => S12 will be measured

Depending on the capability of the VNWA one or two of the above properties will be measured per sub-sweep.

Note, that above example does not show a first sweep in a sweep sequence, thus the list of sub-sweeps is missing.

In case of a listed sweep, the total number of linear frequency segments follows, then the the linear frequency segments follow in the following order: segment start, segment stop, #segment points.

data indicates a frequency point data line. Meaning of following numbers:

n = point number (here 0, 1, 2)

f = point frequency (here 1MHz, 1.5MHz, 2MHz)

Re(S11), Im(S11), Re(S21), Im(S21), Re(S12), Im(S12), Re(S22), Im(S22)

sweep_complete <n> Indicates that the sub-sweep has completed and no more data will be transferred for the running sub-sweep. The number n indicates what sub-sweeps are still to be expected. n=0 indicates that the last sub-sweep of a sweep sequence has ended.

If the user stops a sweep before it is completed, the datagram **sweep_stopped** will acknowledge this. No more data will be transferred after this for the current sweep.

- ▶▶▶ **Note:** The data server is unable to detect and notify a disconnect event from the client.

The VNWA TCP/IP File Transfer Syntax (Server 3)

1. Obtaining remote directory list:

Command sent to server:

dir [filepath]+mask +NULL-Character

- ▶▶▶ **Note:** If no filepath is specified, then the working directory of the server software is assumed.

Example:

`dir c:\VNWA*.jpg`

A file mask is mandatory. The string must be terminated by 0x00.

[Answer from server:](#)

A 128 byte header followed by the list of file names (NOT folders) in the selected directory. Note, that each file name is terminated by a NULL character (0x00).

Header:

`dir <n>`, where n denotes the ASCII representation of the length of the file name list in bytes including the terminating NULL-characters.

The header string is padded with NULL-characters to 128byte size.

2. Downloading a file from the remote server:

[Command sent to server:](#)

`download [filepath]+name +NULL-Character`

Example:

`download c:\VNWA\test.jpg`

The string must be terminated by 0x00.

►►► **Note:** If no filepath is specified, then the working directory of the server software is assumed.

[Answer from server:](#)

A 128 byte header followed by the binary byte stream of the file.

Header:

`file <n>`, where n denotes the ASCII representation of the length of the file in bytes EXCLUDING the 128byte header size.

The header string is padded with NULL-characters to 128byte size.

3. Uploading a file to the remote server:

[Command sent to server:](#)

A 128 byte header followed by the byte stream of the file to be uploaded.

Header:

`upload <n> [filepath]+name +NULL-Characters`

n denotes the ASCII representation of the file size in bytes to be sent EXCLUDING the 128 byte header size.

The header must be padded at its end with NULL characters to 128 bytes size.

Example:

`upload 51349 c:\VNWA\test.jpg`

The string must be padded with 0x00 at its end to 128 bytes size..

►►► **Note:** If no filepath is specified, then the working directory of the server software is assumed.

[Answer from server:](#)

NONE

CONTROLLING THE VNWA BY WINDOWS MESSAGES

| The Windows Message Interface:

Basic Concept of Windows Messages

The Windows command "PostMessage" allows to send a short message to any specific visible or invisible Window:

```
BOOL WINAPI PostMessage(
    __in_opt HWND hWnd,
    __in     UINT Msg,
    __in     WPARAM wParam,
    __in     LPARAM lParam
);
```

Parameter description:

- **hWnd** is the window handle (or address) so the specific window can be reached.
- **Msg** is a unique message number upon which VNWA is prepared react.
- **wParam** and **lParam** are two 4 byte integer constants, that can be sent to VNWA:
wParam is interpreted as VNWA command.
lParam is interpreted as an optional parameter for the VNWA command.

Implemented VNWA remote commands:

wParam:	Command:	lParam:
0	terminate VNWA	n.a.
1	sweep	bits:select 0:S21, 1:S11, 2:S12, 3:S22
2	load cal [rfile]	n.a.
3	load mastercal [rfile]	n.a.
4	write s2p [wfile]	n.a.
5	write s1p [wfile]	value:selects saved record: 0:S21, 1:S11, 2:S12, 3:S22
6	change rfile string	0: clear string 1...255: add char(lParam) to string
7	change wfile string	0: clear string 1...255: add char(lParam) to string
8	set start frequency	start frequency [Hz]
9	set stop frequency	stop frequency [Hz]
10	set number of points	number of points
11	set time per data point	time [microseconds]
12	set sweep mode	0: Lin, 1: Log
13	read and execute [rfile] script file (*)	
14	set RF DDS	frequency [Hz]
15	set LO DDS	frequency [Hz]
16	set VNWA	frequency [Hz] RF and LO DDS are being set simultaneously with IF
offset		
17	set TX power	Amplitude = 0...16383
18	continuous sweep	bits:select Bit 0:S21, Bit 1:S11, Bit 2:S12, Bit 3:S22
19	stop continuous sweep	0: stop after sweep completion, 1: stop immediately
20	set calibration direction	0: forward, 1: reverse (if S-parameter test set available, otherwise ignored)
21	calibration sweep	0: Short, 1: Open, 2: Load, 3: Crosstalk, 4: Thru, 5: Thru Match.
Add 16 to display request to connect cal standard.		
22	read snp-file	lower 4 bits:0x00...0x07 select S21, S11,...,Mem4; higher 4 bits: 1=s1p, 2=s2p, 3=s3p
23	read spec-file	(spec tester window will open)
24	spec sweep	bits:select 0:S21, 1:S11, 2:S12, 3:S22 (spec tester window will open, timer will be deactivated)
25	spec result	(return error code= 0: pass, 1: fail, 2: other error)
26	load instrument state [rfile]	0: ignore hardware settings, 1: also load hardware settings
27	save instrument state [rfile]	0: ignore memory spaces, 1: save memory spaces
28	save cal [wfile]	n.a.
29	load cal kit [rfile]	n.a.
255	echo (wParam & 0xFFFF) 0x10000 and lParam by PostMessage back to sender (for debugging while software development)	

(*) For details on the script file syntax see page Controlling the VNWA by a Script File.

wfile is a write file name including path held inside the VNWA application while it is running. This is the file, where measured S-parameters can be stored to. The filename can be modified via remote command 4.

rfile is a read file name including path held inside the VNWA application while it is running. This is the file, where a calibration or mastercalibration is read from. The filename can be modified via remote command 5.

Return Message = Acknowledgement:

When VNWA is being launched for remote operation, it will send a Windows message back to the handle of the calling program passed to VNWA (ownhandle). This message contains the handle of the VNWA application and the message number VNWA listens to.

Also, every remote command sent to VNWA is acknowledged by a return message containing an error code.

WParam:

- If higher 16 bits of WParam are zero, then the message contains the VNWA [remote handle = LParam](#). VNWA listens to [message code = WParam](#) in this case.
- If higher 16 bits of WParam are NOT zero, then the return message contains an error code. The [error code = higher 16 bit of WParam](#).

Error codes:

- 1: No error
- 2: Script file error, non existent file error, non existent directory error
- 3: File access error (e.g. write-file blocked)

Software Implementation

The following Delphi/Pascal example code shows how to remotely start the VNWA software and remotely control it:

```
unit URemote;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ShellAPI, Menus;

type
  TForm1 = class(TForm)
    BStartVNWA: TButton;
    ListBox: TListBox;
    EPath: TEdit;
    ERuntimeArgument: TEdit;
    BSendMessage: TButton;
    ECommand: TEdit;
    EParameter: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    PopupMenu1: TPopupMenu;
    ClearScreen1: TMenuItem;
    Edit5: TEdit;
    BSendWName: TButton;
    Label3: TLabel;
    Label4: TLabel;
    Edit6: TEdit;
    BSendRName: TButton;
    procedure BStartNWAClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure BSendMessageClick(Sender: TObject);
    procedure ClearScreen1Click(Sender: TObject);
    procedure BSendWNameClick(Sender: TObject);
    procedure BSendRNameClick(Sender: TObject);
  private
    Private declarations
  public
    Public declarations
    remotehandle: hwnd;
```

```

WM_REMOTE: WORD;
procedure Receiver(var Msg: TMessage); message WM_USER;
end;

var
  Form1: TForm1;

implementation

$R *.dfm

procedure TForm1.BStartVNWAClick(Sender: TObject);
var path, argument: string;
begin
PostMessage(remotehandle, WM_REMOTE, 0, 0); //try to terminate VNWA if it is still running
sleep(500); //wait for VNWA to terminate
path:=EPath.Text;
argument:=ERuntimeArgument.Text;
ListBox.Clear;
ListBox.AddItem('ShellExecute return code = '+
inttostr(ShellExecute(Handle,'open', PChar( Path),PChar( Argument), nil, SW_SHOWNORMAL)),nil );
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
EPath.Text:='D:\VNWA\VNWA.exe';
ERuntimeArgument.Text:='-remote -silent -debug -callback '+inttostr(self.handle)+' '+inttostr(WM_USER);
end;

procedure TForm1.Receiver(var Msg: TMessage);
var cmd: integer;
begin
cmd:=Msg.WParam shr 16;
ListBox.AddItem('Message received LParam='+inttostr(Msg.LParam)+' WParam='+inttostr(Msg.WParam),nil);
case cmd of
  0: begin
    WM_REMOTE:=Msg.WParam;
    remotehandle:=Msg.LParam;
    ListBox.AddItem('remote handle received',nil);
  end;
  1: begin
    ListBox.AddItem('remote command executed successfully',nil);
  end;
  else ListBox.AddItem('remote command executed with ERROR! Error code = '+
  inttostr(cmd-1),nil);
  end;
end;
end;

procedure TForm1.BSendMessageClick(Sender: TObject);
begin
PostMessage(remotehandle, WM_REMOTE, strtoint(ECommand.Text), strtoint(EParameter.Text));
end;

procedure TForm1.ClearScreen1Click(Sender: TObject);
begin
ListBox.Clear;
end;

procedure TForm1.BSendWNameClick(Sender: TObject);
var s: string;
  i: integer;
begin
s:=Edit5.Text;
PostMessage(remotehandle, WM_REMOTE, 7, 0);
for i:=1 to length(s) do
  PostMessage(remotehandle, WM_REMOTE, 7, byte(s[i]));
end;

procedure TForm1.BSendRNameClick(Sender: TObject);

```

```
var s: string;
    i: integer;
begin
s:=Edit6.Text;
PostMessage(remotehandle, WM_REMOTE, 6, 0);
for i:=1 to length(s) do
  PostMessage(remotehandle, WM_REMOTE, 6, byte(s[i]));
end;

end.
```

CONTROLLING THE VNWA BY A SCRIPT FILE

| The VNWA Script File Syntax:

Implemented VNWA script file commands:

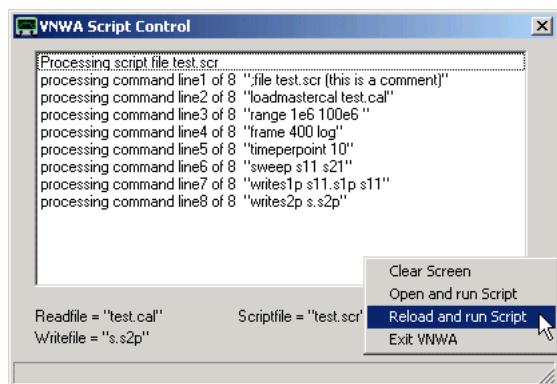
; comment	a line starting with ";" is interpreted as comment.
cadir <direction>	set calibration direction (if S-parameter test set available). Direction = forward / reverse.
calsweep <calibration> [nv]	do a calibration sweep. Calibration = Open, Short, Load, Crosstalk, Thru, ThruMatch. nv=non verbose
loadmastercal cal-filename	load a calibration file as master-calibration
loadcal cal-filename	load a calibration file
savecal cal-filename	save a calibration file
loadcalkit calkit-filename	load a calibration kit file
timeperpoint <t>	sets the measurement time per data point. The number t denotes the time per data point in milliseconds
range <start> <stop>	sets the measurement frequency range. start=start frequency [Hz], stop=stop frequency [Hz]
frame <n> <sweeptype>	sets the sweep frame data. n=number of data points, sweeptype=lin or sweeptype=log sets a linear or logarithmic frequency grid.
setTXpower <p>	sets the TX DDS amplitude to p = 0...16383
sweep [s11] [s21] [s12] [s22]	starts a sweep for the specified data fields
writes1p filename <trace>	writes measurement data of trace to an s1p file. trace can be any of s11, s21, s12, s22
writes2p filename	writes the full measurement data (s11, s21, s12 and s22) to an s2p file
readsnp filename <n> <d>	reads Touchstone data (n=1:s1p, n=2:s2p, n=3:s3p) to data space d (0=S21, 1=S11,..., 7=Mem4)
exitVNWA	terminates the VNWA software

►►► **Note:** VNWA is automatically terminated if an error occurs and the debug-option is not activated.

| Script File Example:

```
:file test.scr (this is a comment)
loadmastercal test.cal
range 1e6 100e6
frame 400 log
timeperpoint 10
sweep s11 s21
writes1p s11.s1p s11
writes2p s.s2p
```

Starting VNWA with the command **VNWA.exe test.scr -debug** will open the VNWA window and the debug or **script control window**:

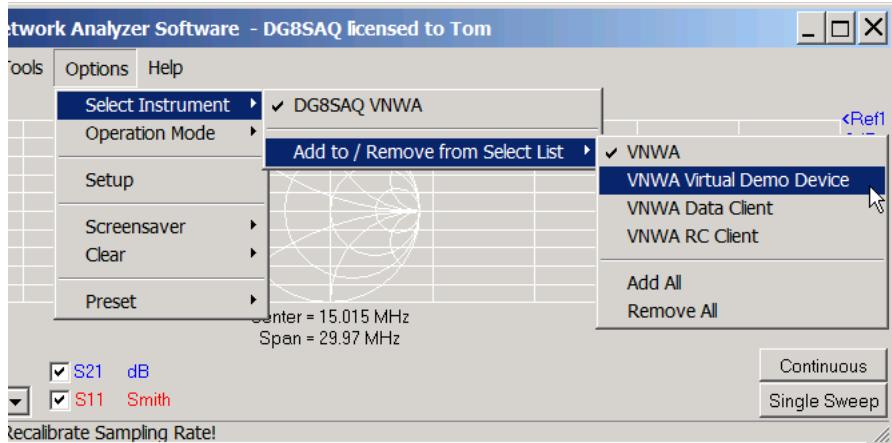


The script control window shows the progress of script execution and occurring errors.

Right-clicking onto the script control window will open the **script control drop-down menu** seen above. It allows to:

- Clear Screen = clear the white message box
- Open and run a script file
- Reload (the previously opened) script file and run it (useful if the file has changed)
- Exit the VNWA application.

VIRTUAL VNWA INSTRUMENTS



The VNWA software includes three virtual VNWA instruments, that do not correspond to actual hardware, but either simulate hardware or connect to other VNWA software instances via LAN/TCP:

VNWA Virtual Demo Device
VNWA Data Client
VNWA RC Client

These will be detailed in the following pages.

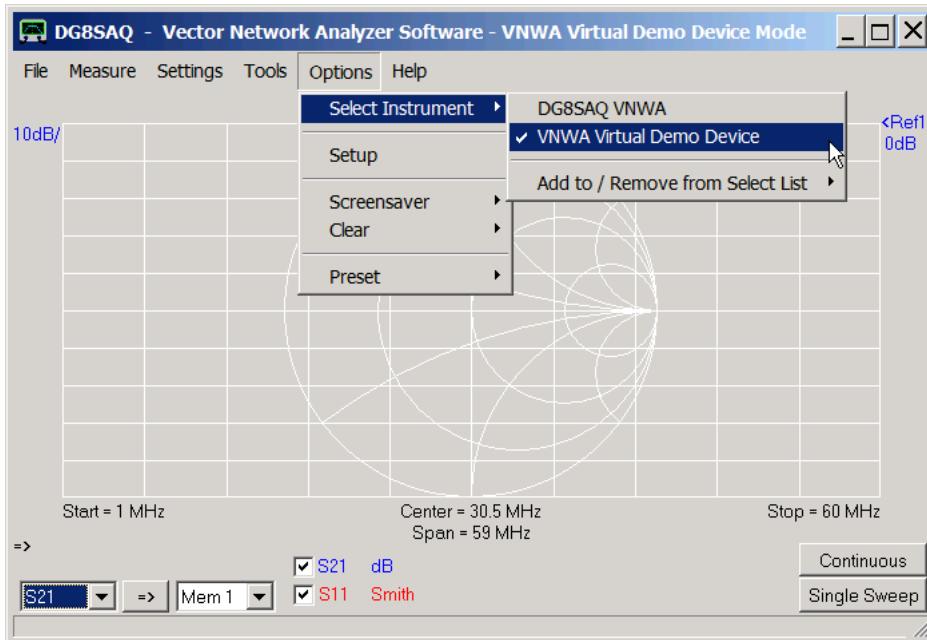
VNWA VIRTUAL DEMO DEVICE

The virtual demo device acts like a hardware VNA controlled by the VNWA software, while it is pure software by itself. It is particularly handy for users, who do not yet own a DG8SAQ VNWA and who want to explore handling of the VNWA software.

It is also a good learning tool as it can simulate measurement results of various test objects, so the user can explore before measuring actual hardware what to expect.

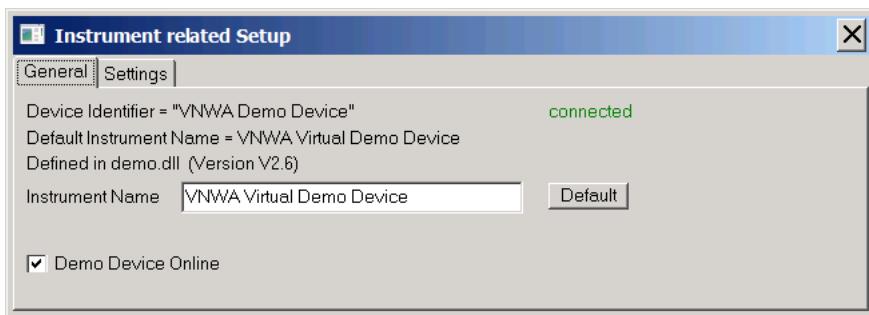
General:

After adding the VNWA Virtual Demo Device to the Select Instruments list, it can be selected there:



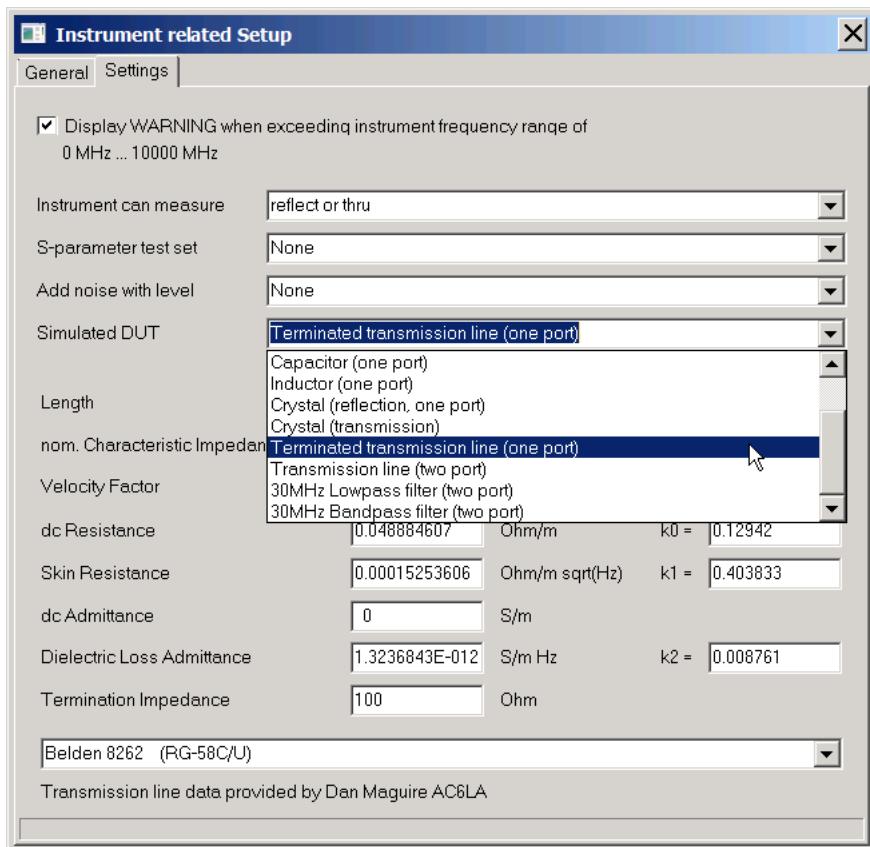
After activation, the blue title bar shows *VNWA Virtual Demo Device* as shown above.

Now, the virtual VNA can be handled pretty much like a hardware VNWA. The first thing is to do a setup via main menu "Options-Setup":



Here, we can change the instrument name and we can switch the virtual instrument on or off. Of course, we will not be able to sweep if the instrument is switched off like if it was a real instrument.

And clearly, we cannot connect real test objects to a virtual VNA. Therefore, we need to select virtual test objects in Setup on the Settings tab:



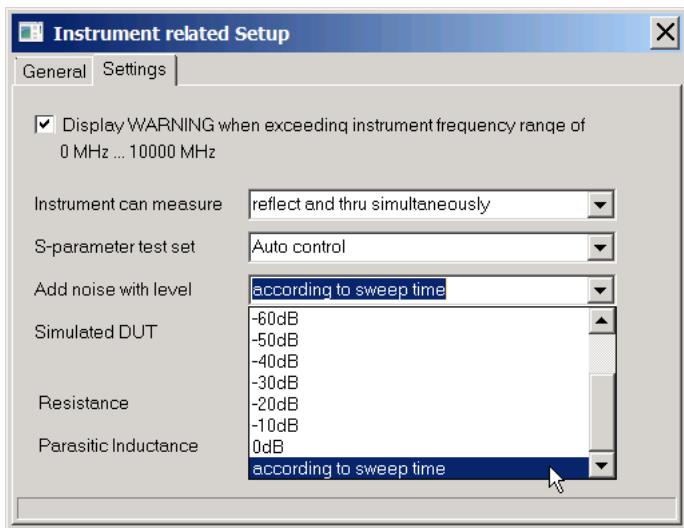
Available virtual test objects:

[Resistor](#)
[Resistor in transmission \(two port\)](#)
[Resistor shunt \(two port\)](#)
[Capacitor](#)
[Inductor](#)
[Crystal](#)
[Crystal in transmission](#)
[Arbitrarily terminated transmission line](#)
[Transmission line two port](#)
[30MHz Lowpass filter](#)
[30MHz Bandpass filter](#)

Moreover, we can specify if the instrument can measure *reflect and thru* in a single sweep (like the VNWA3E) or *reflect or thru* requiring two sweeps to measure reflect and thru (like the VNWA3).

Also, the presence of an S-parameter testset can be simulated.

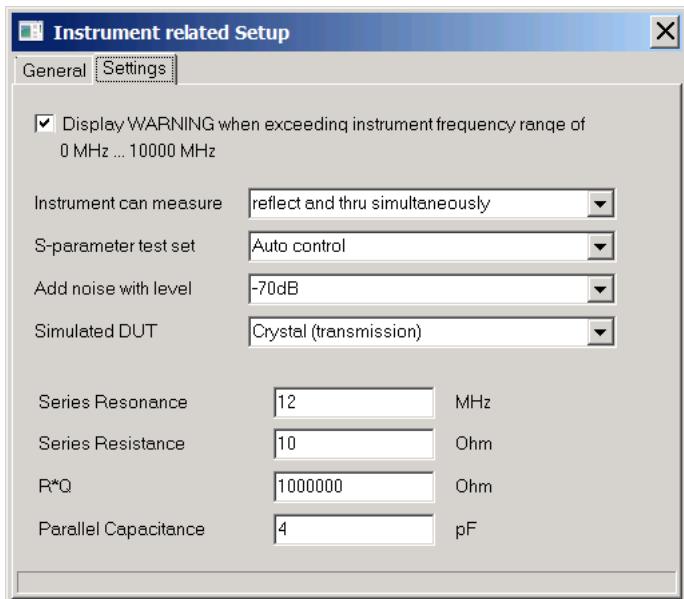
Noise may be added to the simulated measurement to make the measurement look more real and to study the impact of noise on the various measurements:



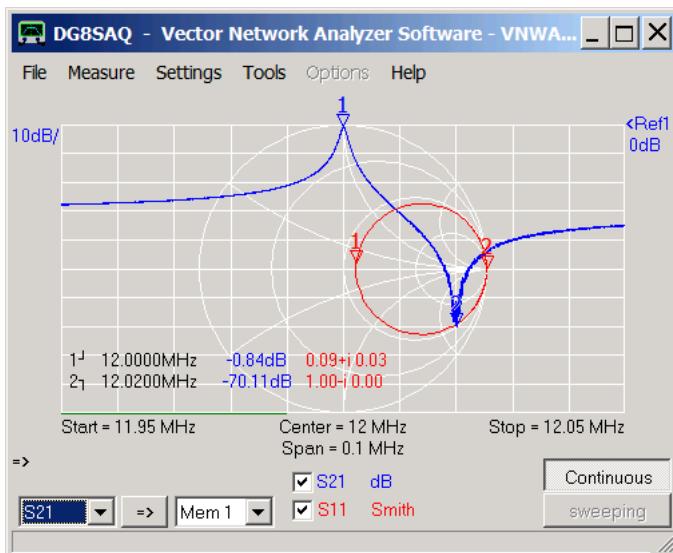
If a noise level "according to sweep time" is selected, then sweep time dependent noise is added like what happens for a real VNA. Slower sweep speed corresponds with a narrower receiver bandwidth and thus with reduced noise power. In this mode, the noise levels are similar to those of the VNWA3 in the VHF range.

Example 1: Crystal Resonator in Transmission

We select the default crystal model in transmission, i.e. the crystal is connected between virtual TX port and RX port.



We measure the virtual crystal like it would be a real one:



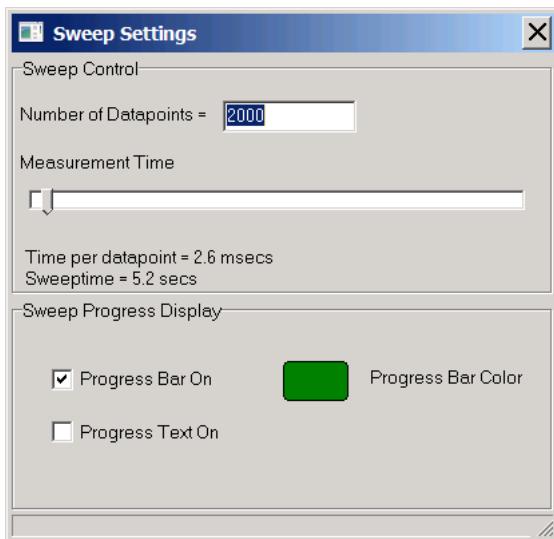
Important: As our virtual VNA is ideal by coding, there is no need to calibrate it prior to measurement. We wouldn't be able to connect calibration standards, anyway.

In our simulated measurement above we see the series resonance (maximum marker 1) and the parallel resonance (minimum marker 2) in the blue transmission trace. Near the parallel resonance we also see the impact of simulated noise, which changes from sweep to sweep.

Note, that the red reflection trace does not show the reflection coefficient of the crystal alone, but it shows the reflection coefficient of the crystal connected in series with the RX port impedance of 50 Ohms as would be the case for a real measurement of this kind.

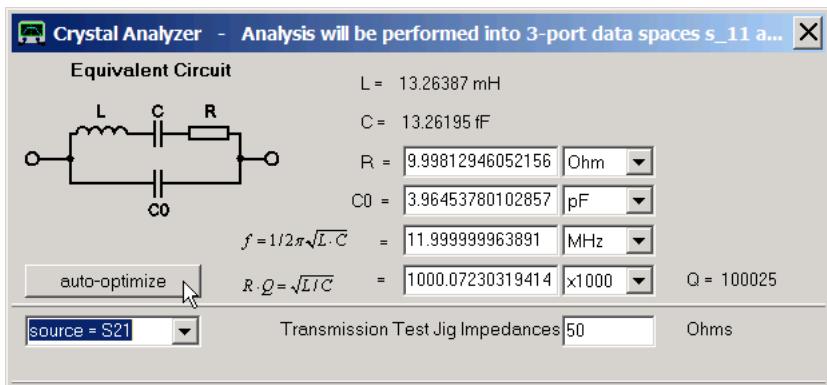
Like with a real hardware VNWA we can do single sweeps or sweep in continuous mode as above and we do see the green sweep progress bar on the bottom.

Also, we can specify the sweep speed via main menu *Settings-Sweep*:



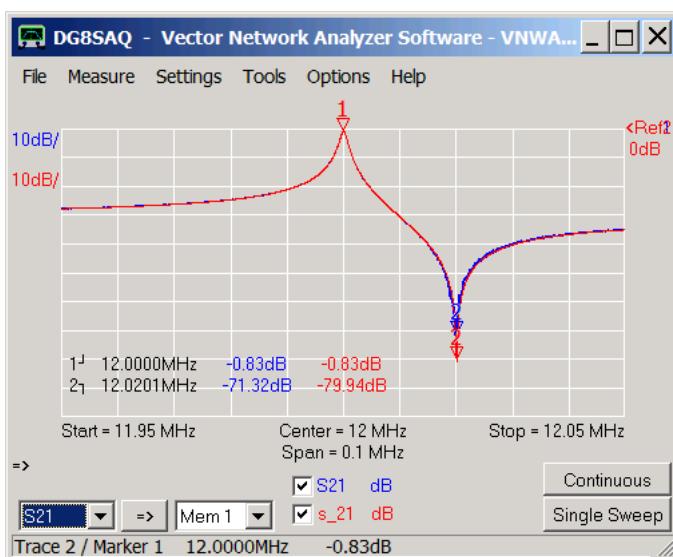
And we have all tools of the VNWA software available for analyzing the measurement result like we would for a real measurement.

Next, we use the crystal tool (main menu Tools-Crystal Analyzer) to extract the motional parameters of our simulated crystal:



And indeed, above we extract the parameters that we have originally set as test object parameters. Note, that with a transmission measurement we need to analyze S21.

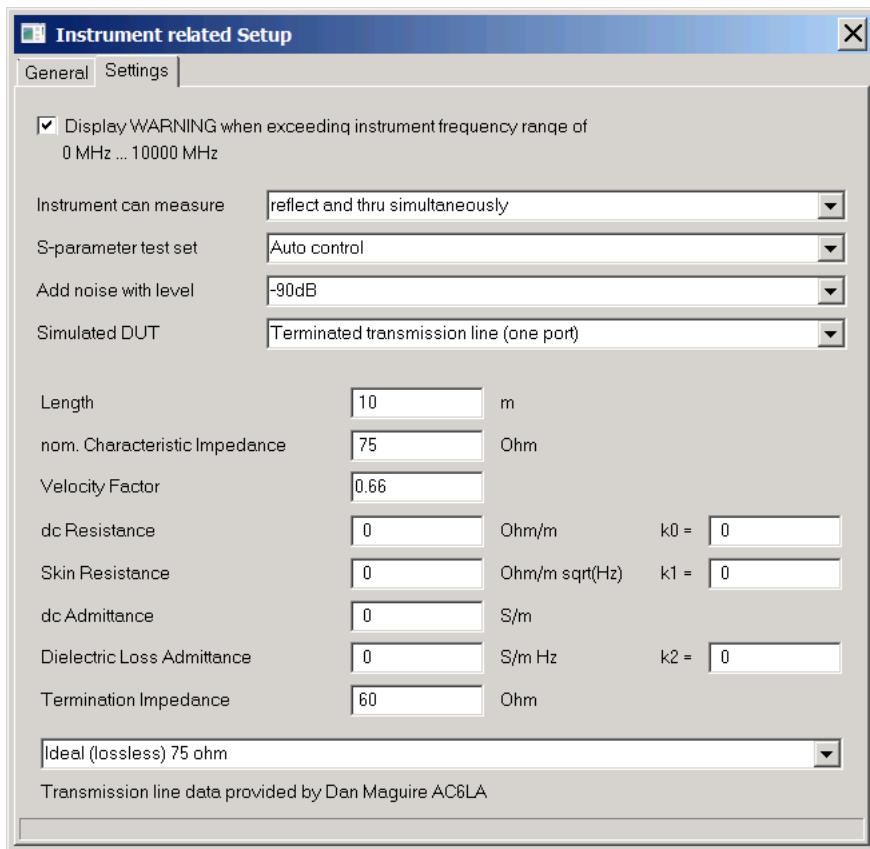
We may inspect the match between simulated measurement and fit model by looking at 3-port memory space s_21 which is mis-used to hold the model fit:



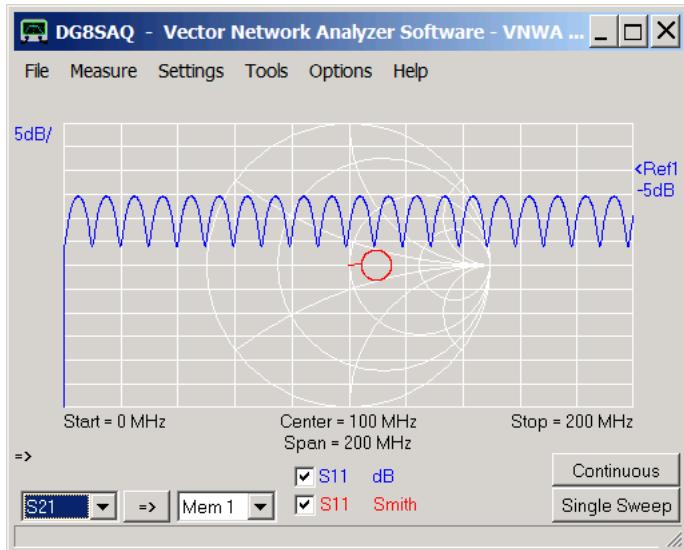
The match is perfect.

Example 2: Terminated Transmission Line

As a second example, let's connect a poorly terminated 75 Ohms transmission line to the virtual VNA ...



and measure its reflectivity by pressing the sweep button on the VNWA main window:

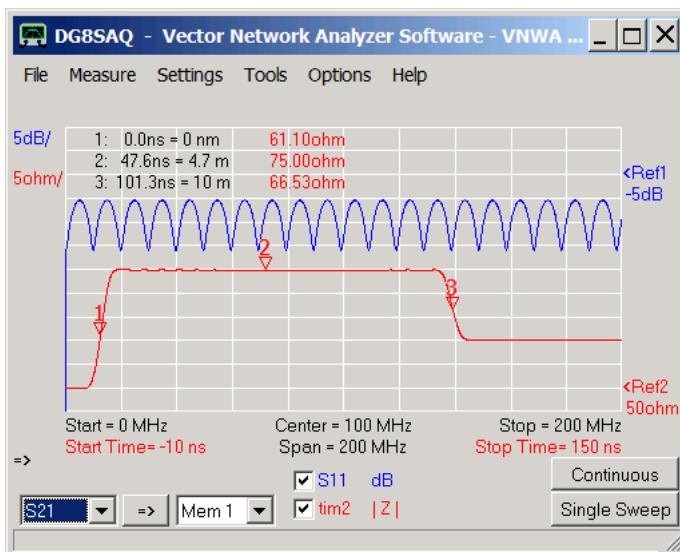
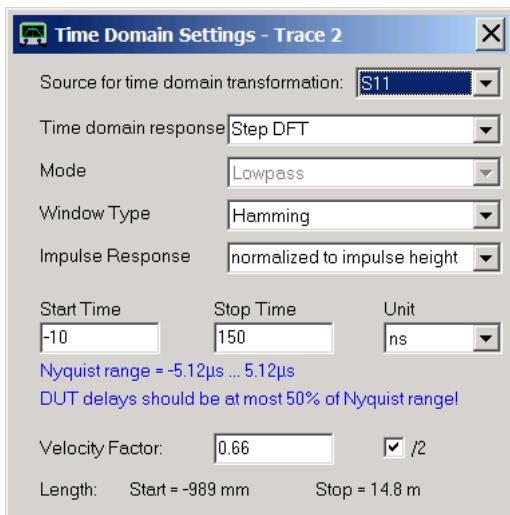


In our simulated measurement above we see the effect of mismatch.

Note, that like a real VNA, the virtual VNA cannot measure down to 0Hz, thus the erroneous down spike.

But the following time domain analysis requires the 0Hz frequency grid point. The fact that it does not contain useful data doesn't matter for this application.

Let's do a time domain analysis of our measurement result from above in terms of step response and impedance:



From our time analysis (red time trace above) we can read that we have a 50 Ohm VNA (negative times) with 10m (marker 3) of 75 Ohms (marker 2) transmission line connected, that is terminated with 60 Ohms (beyond marker 3). Note, that the numbers may not come out exactly as a transmission line may have loss and dispersion and we sweep over a limited frequency range.

With this tool, you can easily study the impact of various settings and test object parameters on the simulated measurement result and thus learn for real measurements.

VNWA DATA CLIENT

The data client device turns the VNWA software into a remote data viewer connected to the main VNWA software instance (VNWA server) which is controlling a VNWA hardware unit (or a virtual VNA).

It serves as an additional graphics window. More than one data client may be run at the same time thus multiplying the number of available graphics windows.

As communication is achieved via the VNWA TCP data interface, the data client need not run on the same computer as the VNWA server. Only a LAN connection to the computer which runs the VNWA server is required. Thus, the data client can also be used to remotely monitor measurements e.g. performed in an automated environment. Data clients may be launched on the fly without interfering with impacting the server.

Each data client offers a full additional set of independent memory spaces, custom traces and data analysis tools.

A data client on the same computer may be easily launched in a preconfigured way via the Tools-Data Client menu. If launched this way, a setup is neither necessary nor possible. Also, calibrating the client is neither necessary nor possible, as it only displays incoming measurement data. Moreover, the main VNWA instance will automatically be switched into server mode and the TCP connection will be established automatically.

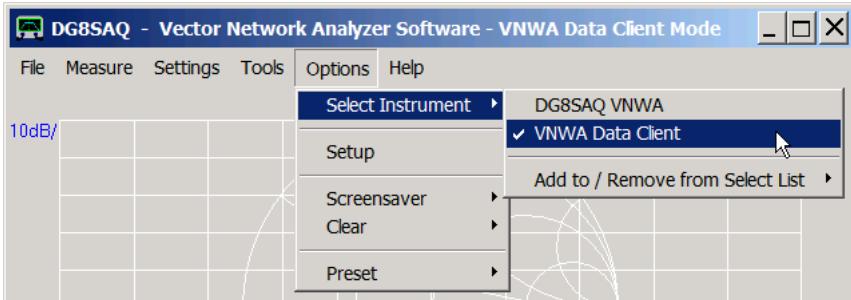
When a data client is launched manually, a corresponding server must be started as well and both must be connected manually. These procedures are described in the following.

General:

1. Launching Client

A second instance of the VNWA software different from the one controlling the VNWA hardware must be started from a directory *different* from the main installation folder. Either create a separate VNWA installation in a different folder or manually copy the installation folder to a separate folder. Hardware setup may be omitted there.

In this second VNWA software instance, which we will call client, we can select the virtual data client device via the *Options-Select Instruments* menu after we have added it to the select-instruments-list via the same menu:



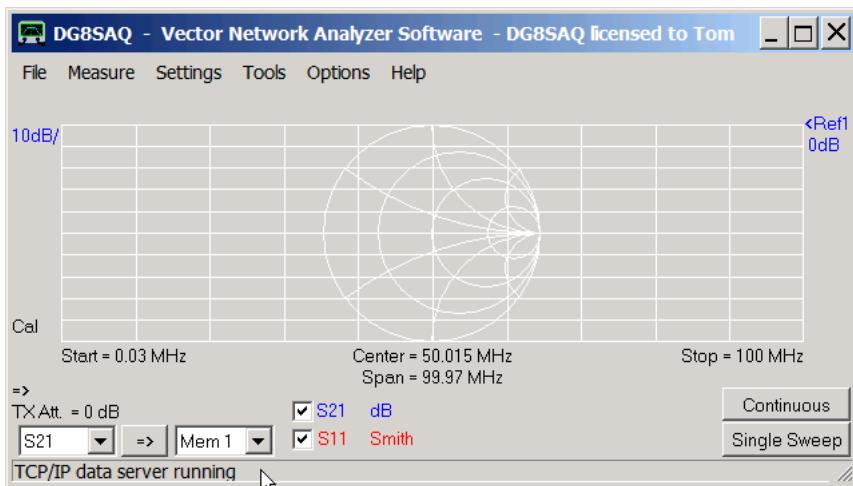
2. Launching Server:

Next we start the VNWA software from the main installation folder and launch the corresponding data server that the data client is supposed to listen to. Make sure, the client instance is NOT in VNWA mode to avoid simultaneous hardware access of two instances.

The possible ways to start a data server are described on page Controlling the VNWA via TCP/IP.

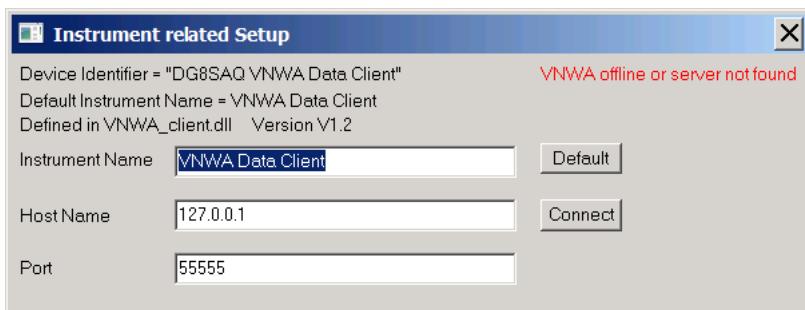
The simplest way is to use the keyboard combination `ctrl+alt+s` in the VNWA main instance which is controlling the VNWA hardware in order to switch the data server on.

Success will be prompted in the bottom status line:

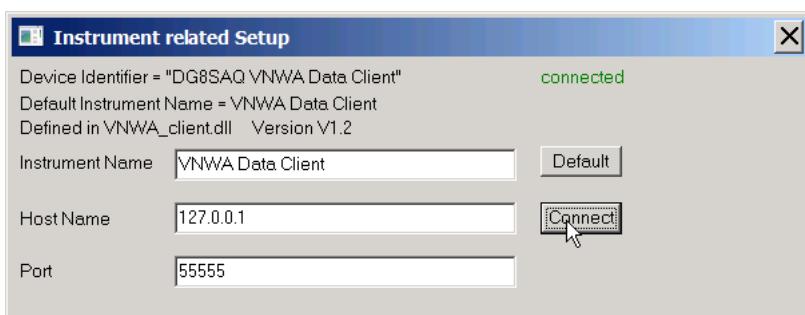


3. Connecting the Client to the Server:

In the VNWA client instance open main menu *Options-Setup*. The client will initially be disconnected as we have started the server after the client:



After entering the host name or host IP address of the server computer and the base port number (default it 55555) we can connect the client by pressing the *connect-button*:



Now, the client is ready to receive measurement data from the server.

►►► **Note:** The default IP address 127.0.0.1 corresponds to the local computer running the client. This is address valid for any computer. If the client is to be connected to a remote computer, use the proper IP address or computer name of the remote computer.

►►► **Note:** Note, that the client will memorize these settings and, when the server will be started before the client in future, the client will automatically connect to the server upon client start.

Example 1: Capturing a single sweep sequence

When we press the *Single Sweep* button of the client, the client will wait for a measurement sequence from the server. As soon as we start a sweep on the server, the measurement data will also show in real time on the client:

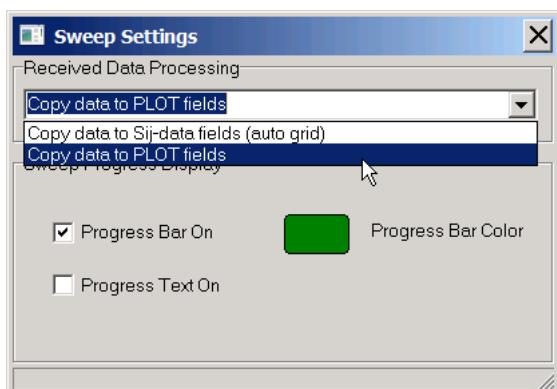


So, with the data client we have an extra display window where we can use marker settings and display settings different from the server window. In standard transfer mode, the frequency axes of the two must be identical, though (apart from box zooms).

The second example shows how to overcome this restriction.

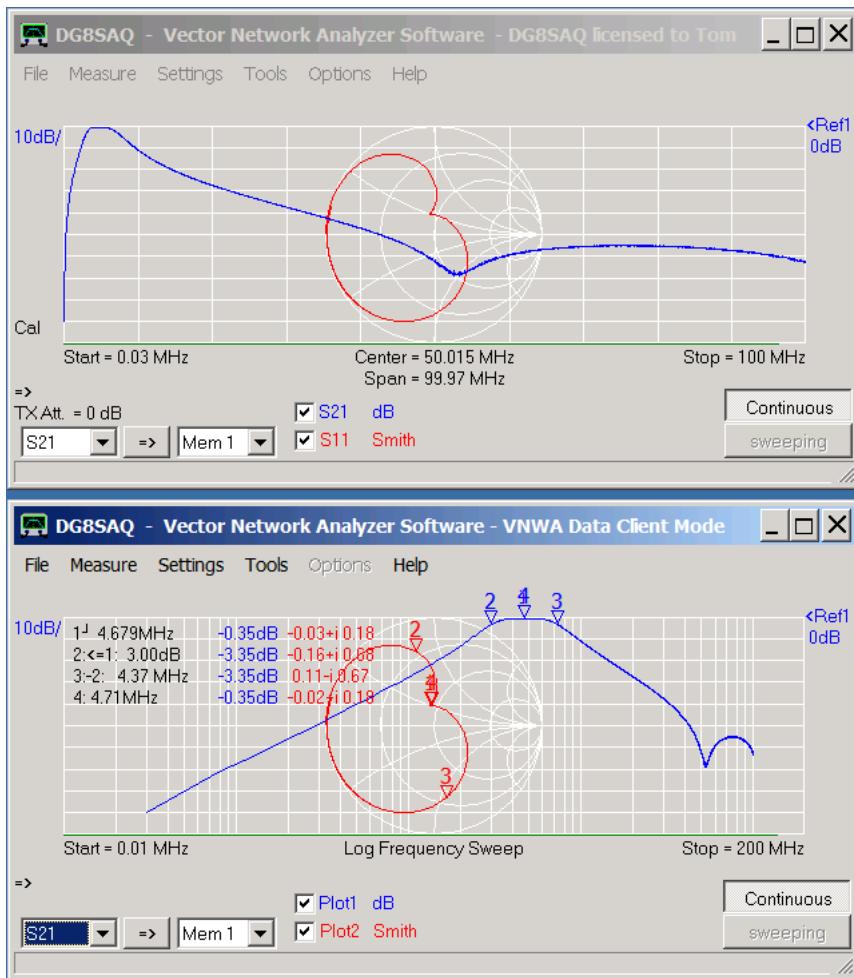
Example 2: Displaying measurement data with different frequency axis

We may configure the data client to receive data into the extended functionality *Plot* memory spaces instead of the default *Sij* measurement memory spaces via the client *Settings-Sweep* menu:



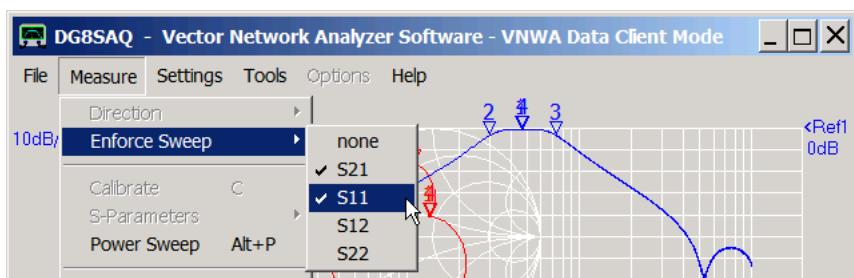
While the client will automatically adapt to the server frequency grid when receiving data into the default *Sij* memory spaces, we are free to specify any server frequency grid when using Plot data spaces.

Here, the server sweeps a linear frequency grid, but the client displays the data on a logarithmic frequency axis:



Note, that the user may select any start and stop frequency in this case.

►► **Important:** You must use the *Enforce Sweep* feature on the client to make the client sweep while neither showing an S11 nor S21 data field:



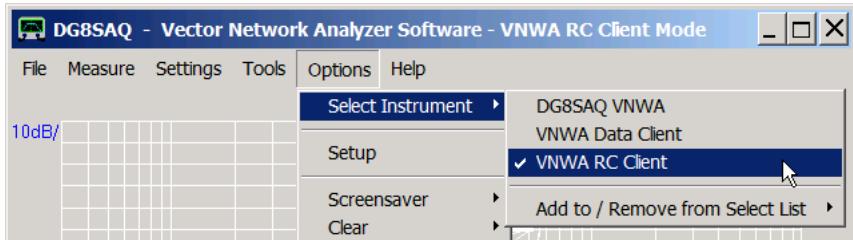
If not used, the error message "nothing to sweep" will occur as VNWA normally only sweeps what is being displayed unless explicitly enforced.

VNWA RC CLIENT

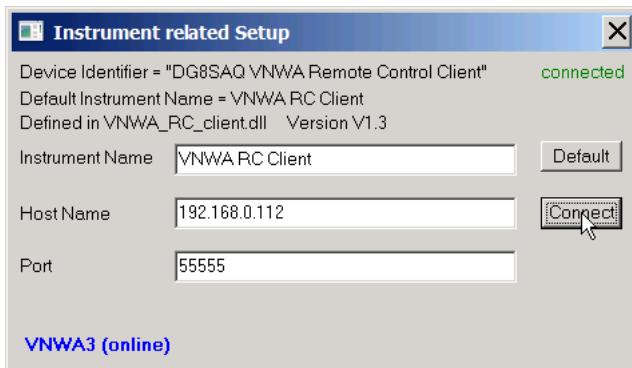
The VNWA RC Client was created as proof-of-concept for the VNWA TCP interface.

The RC Client allows to remote-control a VNWA installation running on any remote computer connected to the network. Like the VNWA data client it will receive and display measurement data in real time.

The virtual VNWA RC Client instrument can be selected via the *Options-Select Instruments* menu after having added it to the select-instruments-list via the same menu:



After having started a VNWA server instance e.g. on a remote computer as described on page Controlling the VNWA via TCP/IP, connect the RC client via its *Options-Setup* menu ...

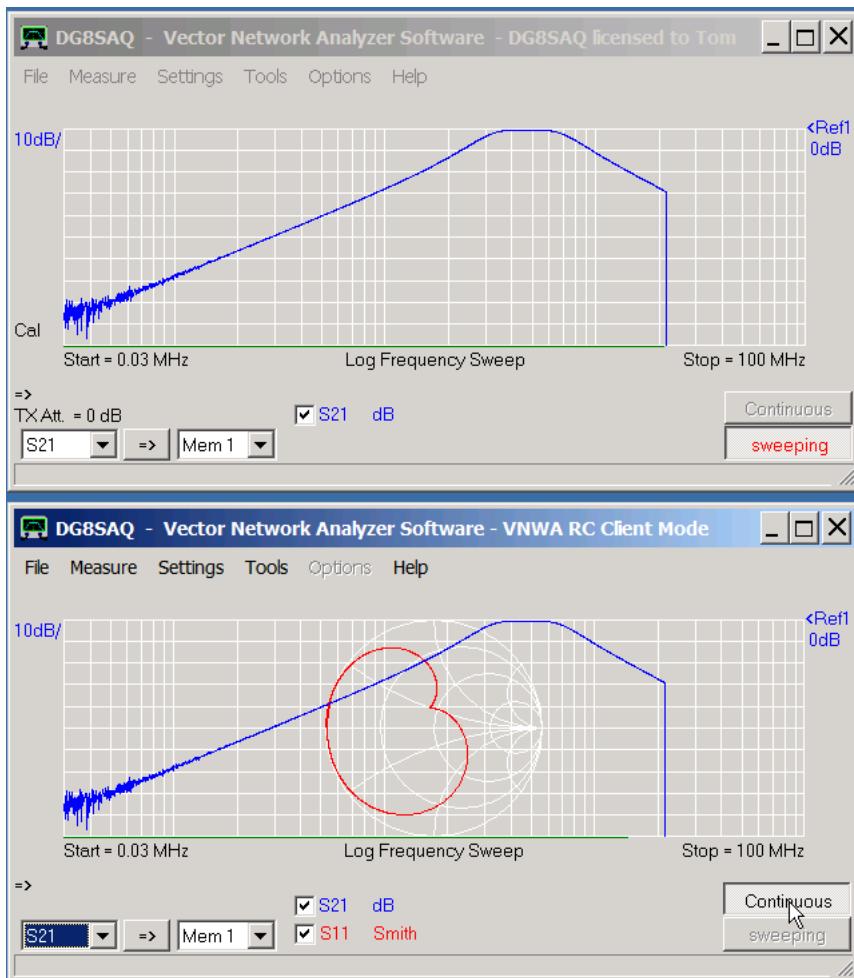


... by entering the remote IP address or host name, entering the appropriate base port number of the remote server and pressing the connect button.

The green *connected* label top right indicates that the RC client is connected to the remote server.

The blue *VNWA3 (online)* label bottom left indicates that the remote server is connected to a VNWA3 and that the VNWA hardware is online.

Now, the RC client is ready to control the remote server. Pressing the RC client *Single Sweep* or *Continuous* button will make the server sweep and send measurement data back to the RC client:



Note, that the server display changes while sweeping. It will only show the trace of the running sweep. The instrument calibration can either be performed on the remote server or on the client. But normally calibrating on the server makes more sense as one needs access to the VNWA instrument connected to the server in order to connect the calibration standards.

Also note, that changing the frequency grid on the client will automatically change the frequency grid on the server, too. This may invalidate the server calibration and may require a fresh calibration on the server.

BINARY S-PARAMETER FILE FORMAT

In order to make data logging more efficient and to cut down on disk usage, a new efficient binary storage format for S-parameters has been introduced with software version VNWA35.8.v. This format can be used alternatively to the standard Touchstone format.

Binary file format description, format version 2:

All numbers are stored according to Intel format with lowest significant byte first.
All floating point numbers are 4 byte, i.e. single precision if not otherwise stated.

Byte 0..2: File identifier "VNA"

Byte 3: Format version (currently 2)

Byte 4: Stored record(s): 1=S21; 2=S11; 3=S12; 4=S22; 5..8=Mem1...4; 200=all Sij

Byte 5: Sweep type: 1=Linear; 2=Logarithmic; 3=Listed Sweep

Byte 6..7 Number of frequency points (two byte unsigned integer)

Byte 8..11 4 byte floating point: VNWA temperature during measurement in degrees Celsius (VNWA3 only)

Byte 12..15 reserved

Byte 16..23 Double precision floating point number = date and time when data has been stored, measured in days that have passed since 12/30/1899 (=TDateTime format), same time as used in logged file names.

Byte 24...

If Sweep type Linear or Logarithmic:

Start frequency [Hz] (8 bytes floating point for format version 2 or 4byte floating point for format version 1, see byte 3)

Stop frequency [Hz] (8 bytes floating point for format version 2 or 4byte floating point for format version 1, see byte 3)

If Listed Sweep: (subject to change)

List of all frequency points [Hz] (#frequency points * 8 bytes floating point for format version 2 or 4byte floating point for format version 1, see byte 3)

Data:

Data points in format real/imag: (4 bytes/4bytes)*#frequency points*#records, all single precision floating point.

USER DATA POSTPROCESSING

From VNWA version 35.9.7 onwards, the VNWA software offers a **user programming interface** which allows the user to **automatically process measured sweep data** after completion of each full sweep cycle. The interface is accomplished by a user definable Windows dynamic link library file (dll). VNWA creates a local copy of the just measured sweep data, launches a lower priority parallel thread, inside which it passes the sweep data to a function inside a user definable dynamic link library. While the user function processes the sweep data inside a thread, VNWA may already measure the next sweep in parallel.

►►► **Important:** In continuous sweep mode, care must be taken that the external data processing time inside the user dll does **NOT exceed the sweep time**. Otherwise, unfinished data processing threads will accumulate and the system will eventually crash when it runs out of memory. Also, make sure to design your dll **thread-safe**.

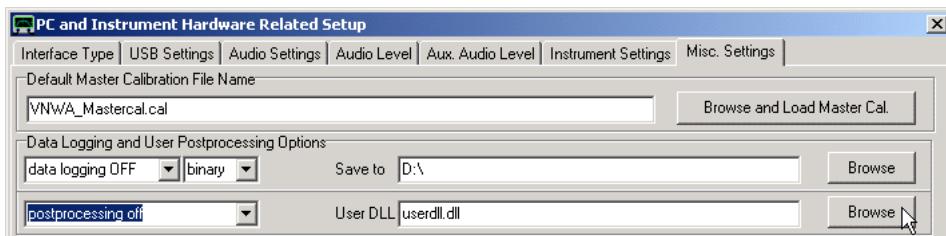
Setup and Activation

The user postprocessing feature is activated in the "Misc. Settings" tab of the VNWA Setup.

First, a **user dll must be defined**. This can conveniently be done by pressing the **browse-button** (see mouse pointer in below screenshot).

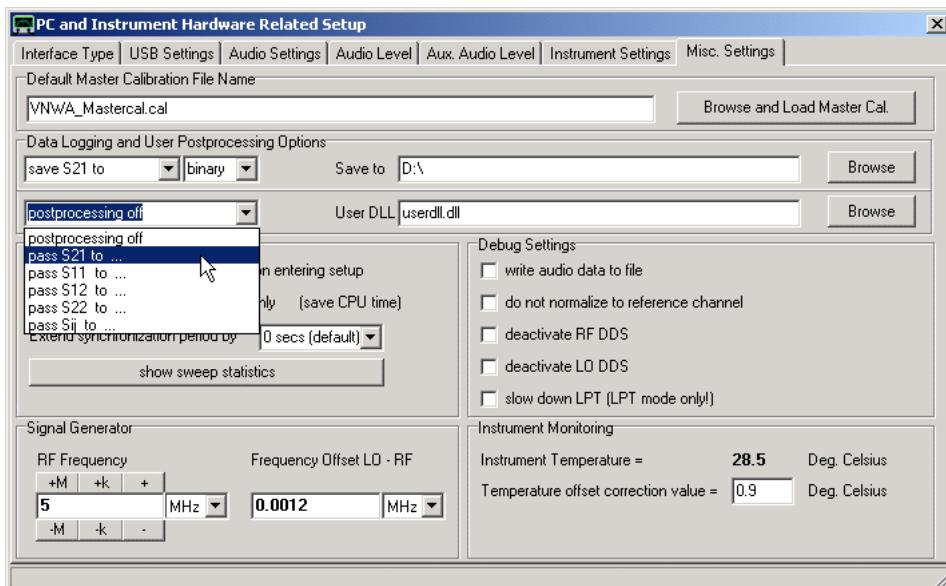
After selecting a dll-file (*.dll), the file will be copied to the VNWA program directory and the file name will be displayed in the edit field next to the browse-button.

Alternatively, the dll may be copied manually to the VNWA program directory and its name may be entered manually into the edit field.



►►► **Note:** Said edit field and browse-button will be **disabled**, as soon as postprocessing is being switched on.

Postprocessing is activated by selecting the data which is to be passed to the user dll:



Above selection will load the user dll into VNWA. If loading is not successful (wrong or corrupted dll), then postprocessing will immediately be switched off and the dll will be unloaded again.

Select another dll in this case.

►►► **Note:** If settings are valid, the user dll will be reloaded automatically at every VNWA program start.

Main Window Feature

Activated postprocessing is being indicated in the VNWA main window:



As accumulation of unfinished postprocessing threads is dangerous, the number of currently running postprocessing threads is being displayed (see red label, zero in above screenshot).

Interface Definition

The user dll must provide the following function interfaces shown in Pascal language:

procedure Init; export; cdecl;

Init is called automatically, right after VNWA has loaded the user dll. Allocate your resources here.

procedure Close; export; cdecl;

Close is called just before VNWA unloads the user dll. Make sure to release all allocated resources (files, memory) here.

procedure Process(FData: TDoubleArray; SType: Integer; SData: TComplexArray;
Len: integer; Rec: integer; tm: TDateTime; temp: Double);
export; cdecl;

Process is called after every completed sweep cycle. Here you do your data processing.

Parameters of function Process:

FData: Array of frequency data (=pointer to double precision floating point values)

Depending on the VNWA sweep mode, the length and contents of this array varies:

Linear or Log Sweep: FData is of length 2 and contains the sweep start and stop frequency

Listed Sweep: The length of FData is equivalent to the number of frequency points. FData contains the frequencies for every single frequency point.

SType: Integer value indicating the VNWA sweep mode.

1: Linear sweep

2: Log Sweep

3: Listed Sweep

SData: Array of sweep data (=pointer to double precision floating point values)

Starting at the first frequency point the array contains the sweep data in the following order: real1,imag1,real2,imag2...

Depending on postprocessing mode the length of SData varies:

- if only one record (e.g. S11) is being passed, then SData contains 2 x # data points real numbers.

- if all records (Sij) are being passed, then SData contains 4 x 2 x # data points real numbers.

Then the order of records is S21, S11, S12, S22.

Len: Integer value indicating the number of frequency points.

Rec: Integer value indicating which data is being passed:

1: S21

2: S11

3: S12

4: S22

5: all (S21,S11,S12 and S22)

tm: double precision floating point value indicating the system time at which the sweep was terminated, measured in days that have passed since 12/30/1899 (=TDateTime format)

temp: double precision floating point value indicating the VNWA temperature at which the sweep was performed in degrees Celsius.

►►► **Note:** The interface has been defined using the **C-language calling convention** in order to simplify interfacing to

dlls written in C language. Use calling convention **cdecl** when coding in Pascal. Also, the functions must be declared with a leading underscore in Pascal-code for C-language compatibility, e.g. the function Init will show as _Init in Pascal-code.

Programming Examples

A Pascal/Delphi code example can be found [here](#).

A C++ code example can be found [here](#).

PASCAL/DELPHI USERDLL EXAMPLE

The following Pascal code defines a simple example dll with name userdll.dll:

```
library UserDLL;

uses
  Windows,
  SysUtils,
  Classes;

type
  TDoubleArray= array of double;
  TComplex = Record
    R : double;
    I : double;
  end;
  TComplexArray= array of TComplex;

$R *.res

procedure _Init; export; cdecl;
begin
  FreeConsole;
  AllocConsole;
  writeln('UserDLL function "Init" called.');
end;

procedure _Close; export; cdecl;
begin
  FreeConsole;
end;

procedure _Process(FData: TDoubleArray; SType: Integer; SData: TComplexArray;
  Len: integer; Rec: integer; tm: TDateTime; temp: Double);
  export; cdecl;
begin
  writeln('UserDLL function "Process" called.');
  writeln(' Start Freq = ',FData[0]);
  writeln(' Stop Freq = ',FData[1]);
  writeln(' Temp = ',temp);
end;

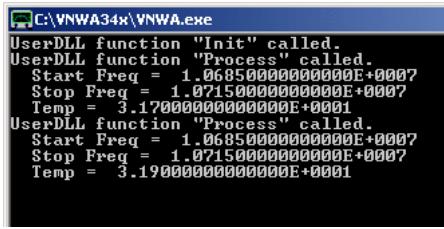
exports _Init;
exports _Close;
exports _Process;

begin
end.
```

►►► **Note:** The leading underscores in above procedure declarations (e.g. `_Init` instead of `Init`) are required for C-language compatibility.

The procedure **Init** initializes a console window and writes a short message to it.
The procedure **Process** writes some of the passed data to the console.
The procedure **Close** releases the console again.

After loading the user dll into VNWA, the console window opens as defined in function init:



```
C:\VNWA34x\VNWA.exe
UserDLL function "Init" called.
UserDLL function "Process" called.
Start Freq = 1.0685000000000E+0007
Stop Freq = 1.0715000000000E+0007
Temp = 3.1700000000000E+0001
UserDLL function "Process" called.
Start Freq = 1.0685000000000E+0007
Stop Freq = 1.0715000000000E+0007
Temp = 3.1900000000000E+0001
```

Init only writes the first line. After each sweep an additional message from the function Process is being written to the console as seen above.

BORLAND C++ BUILDER USERDLL EXAMPLE

The following Borland C-code defines a simple example dll with name userdll.dll:

```
// Example UserDLL for VNWA by DG8SAQ and M0PUB
// Written with Borland C++ 6.0 Personal
// March 8th, 2011

#include <windows.h>
#include <stdio.h>

HANDLE ho; //console handle

// User Function 'Init'
// will be called when UserDLL is being loaded
extern "C" __declspec(dllexport)
void Init (void)
{
    char *string = "UserDLL function Init called.";
    DWORD d;
    // release current console if any
    FreeConsole();
    // create new console
    AllocConsole();
    // get console output handle
    ho = GetStdHandle(STD_OUTPUT_HANDLE);
    // output string
    WriteConsole(ho, string, lstrlen(string),&d, NULL);
}

// User Function 'Close'
// will be called when UserDLL is being unloaded
extern "C" __declspec(dllexport)
void Close (void)
{
    // close console output handle
    CloseHandle(ho);
    // release console
    FreeConsole();
}

// User Function 'Process'
// will be called after every complete sweep cycle, if postprocessing is activated
//
// Pascal declaration for comparison
// procedure _Process(FData: TDoubleArray; SType: Integer; SData: TComplexArray;
//                   Len: integer; Rec: integer; tm: TDateTime; temp: Double);
//                   export; cdecl;
// Note that the C-code doesn't require the underscore in front of the function name
extern "C" __declspec(dllexport)
void Process (double *FData, int SType, double *SData, int Len,
              int what, double tm, double temp)
{
    char string[50];
    char sfloat[25];
    DWORD d;

    strcpy(string, "UserDLL function Process called.");
    WriteConsole(ho, string, lstrlen(string),&d, NULL);
    strcpy(string, " Start Frequency = ");
    sprintf (sfloat, "%.*f", 10, FData[0]);
    strcat(string, sfloat);
    strcat(string, "");
    WriteConsole(ho, string, lstrlen(string),&d, NULL);
}
```

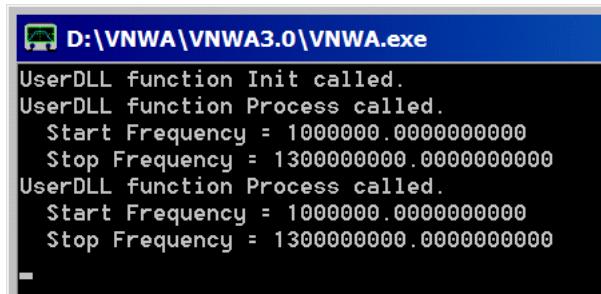
```
strcpy(string, " Stop Frequency = ");
sprintf(&sfloat, "%.*f", 10, FData[1]);
strcat(string, sfloat);
strcat(string, "");
WriteConsole(ho, string, lstrlen(string),&d, NULL);
}
```

The procedure **Init** initializes a console window and writes a short message to it.

The procedure **Process** writes some of the passed data to the console.

The procedure **Close** releases the console again.

After loading the user dll into VNWA, the console window opens as defined in function init:



```
UserDLL function Init called.
UserDLL function Process called.
Start Frequency = 10000000.0000000000
Stop Frequency = 1300000000.0000000000
UserDLL function Process called.
Start Frequency = 10000000.0000000000
Stop Frequency = 1300000000.0000000000
```

Init only writes the first line. After each sweep an additional message from the function Process is being written to the console as seen above.

MS VISUAL C++ USERDLL EXAMPLE

The following Microsoft Visual C++-code defines a simple example dll with name userdll.dll:

```
// Example 'User Processing' DLL for use with VNWA.EXE by M0PUB

#define WIN32_LEAN_AND_MEAN      // Exclude rarely-used definitions from Windows headers

// Windows Header Files:
#include <windows.h>
#include <math.h>
#include <cstdio>
#include <assert.h>

// DLL entry point - called when DLL is loaded and unloaded. Also when the system starts or terminates a process or
// thread,
// it calls DllMain() for each loaded DLL using the first thread of the process.
// *** Warning *** Be careful what you do in this routine. Generally better to perform
// initialisation elsewhere.

BOOL APIENTRY DllMain( HANDLE hModule,
                       DWORD  ul_reason_for_call,
                       LPVOID lpReserved
                      )
{
    return TRUE;
}

HANDLE ho; //console handle

// User Function 'Init'
// Will be called when this DLL is loaded by VNWA.EXE, and is a better place to perform initialisation
// than DllMain()
// Note: Visual C++ Express 2008 does not automatically prefix 'extern "C" __declspec' functions with an
// underscore, so we explicitly add one to the function name for this build environment

extern "C" __declspec(dllexport)
void _Init (void)
{
    char *string = "UserDLL function Init called.";
    DWORD d;

    // release current console if any
    FreeConsole();

    // create new console
    AllocConsole();

    // get console output handle
    ho = GetStdHandle(STD_OUTPUT_HANDLE);

    // output string
    WriteConsole(ho, string, lstrlen(string),&d, NULL);
}

// User Function 'Close'
// Will be called when UserDLL is being unloaded by VNWA.EXE
// Note: Visual C++ Express 2008 does not automatically prefix 'extern "C" __declspec' functions with an
// underscore, so we explicitly add one to the function name for this build environment

extern "C" __declspec(dllexport)
void _Close (void)
{
```

```

// close console output handle
CloseHandle(ho);

// release console
FreeConsole();
}

// User Function 'Process'
// will be called after every complete sweep cycle, if postprocessing is activated
//
// Pascal declaration:
// procedure(FData: Pointer;SType: Integer;SData: Pointer; Len: integer;
//           what: integer; tm: TDateTime; temp: Double); cdecl;
//
// Note: Visual C++ Express 2008 does not automatically prefix 'extern "C" __declspec' functions with an
// underscore, so we explicitly add one to the function name for this build environment

extern "C" __declspec(dllexport)
void _Process (double *FData, int SType, double *SData, int Len,
               int what, double tm, double temp)
{
    char string[50];
    char sfloat[25];
    DWORD d;

    strcpy(string, "UserDLL function Process called.");
    WriteConsole(ho, string, lstrlen(string),&d, NULL);

    strcpy(string, "Start Frequency = ");
    sprintf (sfloat, "%.*f", 10, FData[0]);
    strcat(string, sfloat);
    strcat(string, "");
    WriteConsole(ho, string, lstrlen(string),&d, NULL);

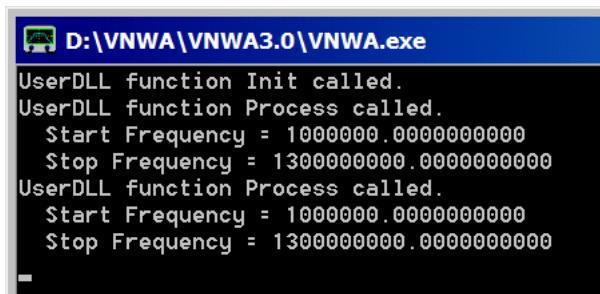
    strcpy(string, "Stop Frequency = ");
    sprintf (sfloat, "%.*f", 10, FData[1]);
    strcat(string, sfloat);
    strcat(string, "");
    WriteConsole(ho, string, lstrlen(string),&d, NULL);
}

```

►►► **Note:** In contrast to Borland C++, underscores in front of the function names are required here!

The procedure **Init** initializes a console window and writes a short message to it.
The procedure **Process** writes some of the passed data to the console.
The procedure **Close** releases the console again.

After loading the user dll into VNWA, the console window opens as defined in function init:



Init only writes the first line. After each sweep an additional message from the function Process is being written to the console as seen above.

OPTIMIZING USB TRAFFIC FOR MODULATOR USE

| **Tutorial about how to use USBView.exe and Connection Optimization for VNWA.**
by Kurt Poulsen OZ7OU

It has been discovered that certain VNWA functions (particularly the signal generator AM and FM modulator) can be interfered, pending which USB port on the computer the VNWA is connected to, when other USB devices, like an external Keyboard or e.g. an USB Mouse also is connected to another USB Port than the VNWA. It is in particular important for VNWA3, which has added functions compared to VNWA2, but VNWA2 might also be interfered.

To find out, when such interference happen, the free USBView.exe is a very helpful application.

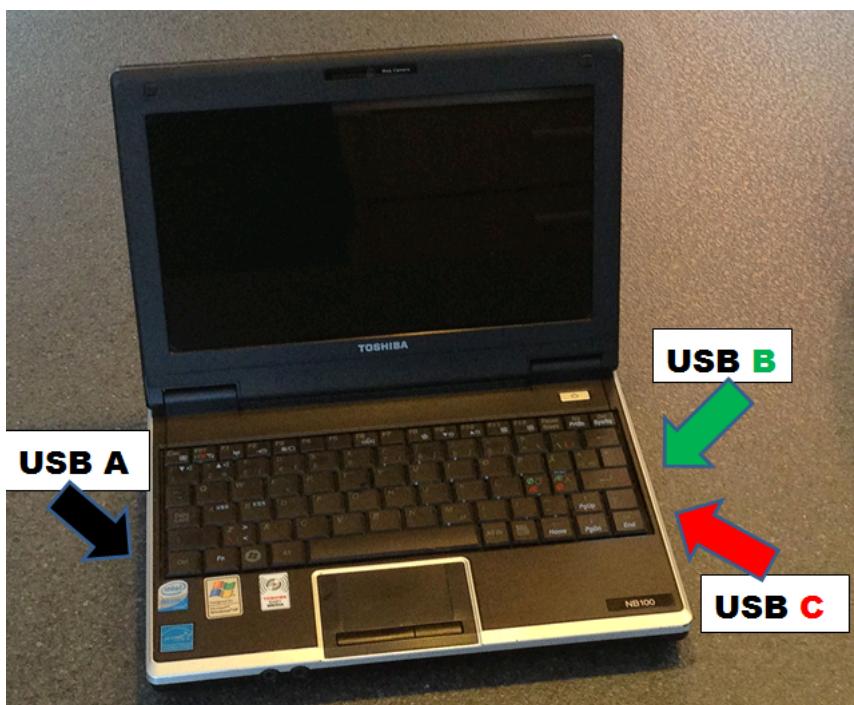
Search with Google for Microsoft USBView.exe or use the link provided here.

Use <http://www.ftdichip.com/Support/Utilities.htm> and then look for "Microsoft USBView - USB Connection Viewer" to be download as a zip file.

When using USBView, remember to enable in "Options" the "Auto Refresh", otherwise you have to press F5 to update the reading of the USB System.

Investigation:

A small 10 inch laptop has been used for the demonstration, provided with three USB ports, One on the left side called USB A, and two on the right side called USB B and USB C.



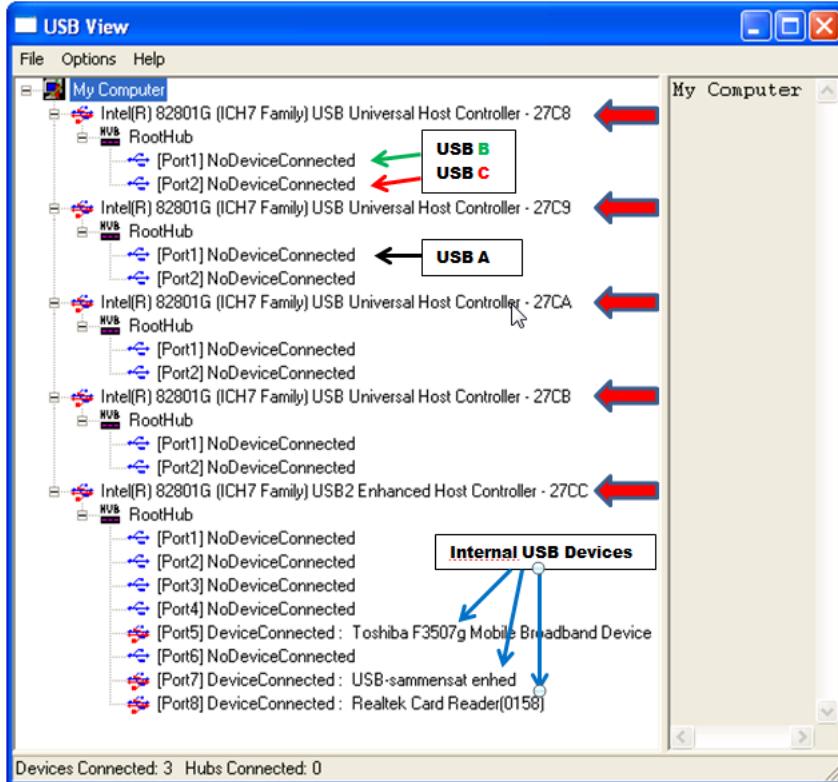
Preface:

I will be shown that connecting e.g. the VNWA3 and an external USB device, like an external keyboard, to USB B and C, the modulation for the Signal Generator is modulated by the scanning of the keyboard, such that the modulation has a "stuttering" sound. Whereas, if the VNWA3 is connected to USB A, this "interference" is not present. The conclusion drawn is that if the VNWA3 (and probably also VNWA2) when sharing a USB Universal Host Controller (a RootHub,) with another USB device, performance is likely to be influenced, pending the intensity of the "foreign device" data/traffic. To play it the safe way, test with USBView.exe how the VNWA can be solely serviced by a USB Universal Host Controller not shared. For a PC with three USB ports it is possible to find which the USB Port which never will share, but for a PC with e.g 4 USB Ports there is most likely two USB Universal Host Controller, each with two ports, which is servicing the 4 USB Ports, and thus two combinations will cause trouble.

However it shall be demonstrated in the following, how an External HUB can solve the problem on old PC's not having more than 2 USB ports and those USB ports serviced by the same USB Universal Host Controller.

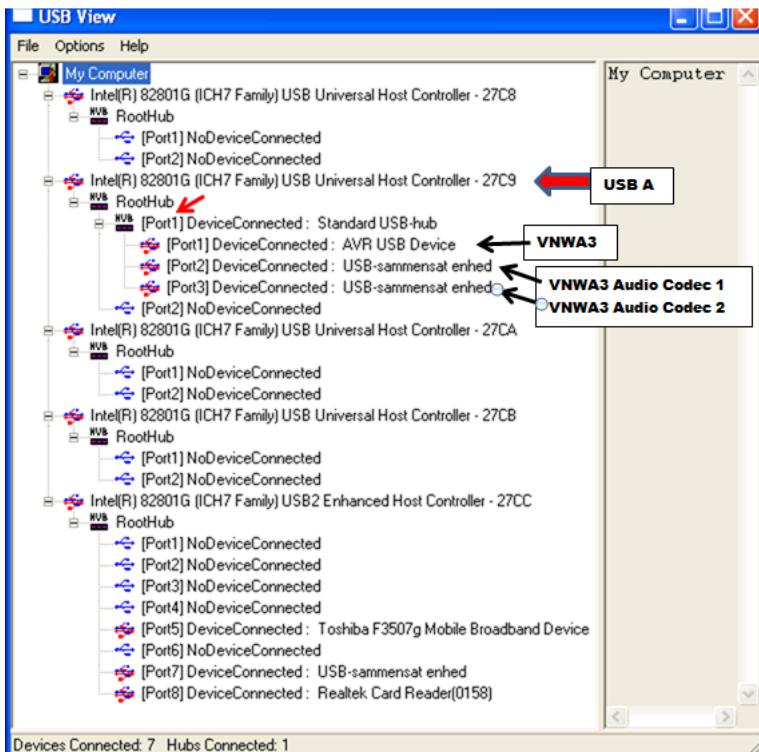
Demonstration of above statements:

Running USBView without any devices connected to USB A, B or C is shown below.



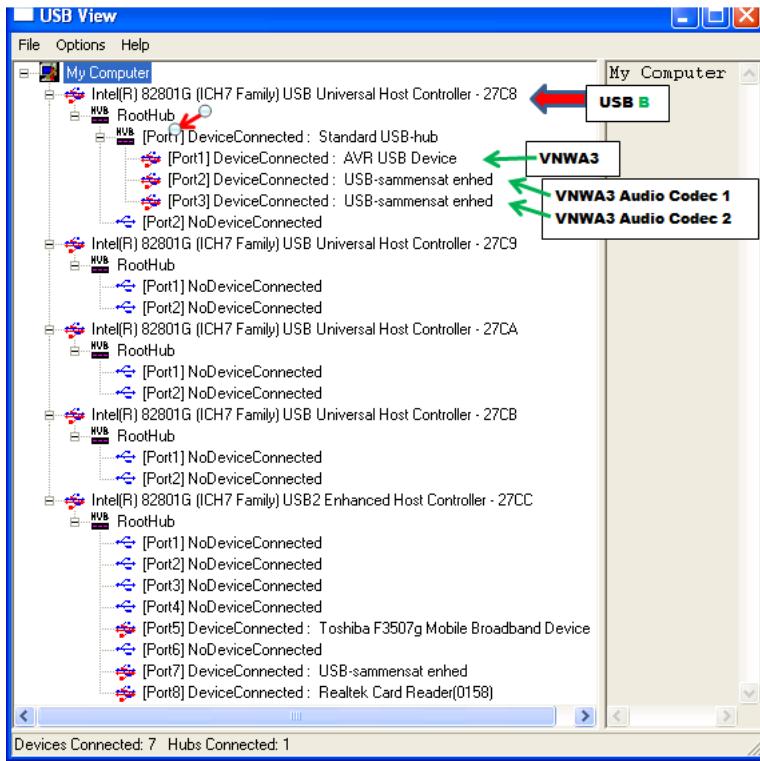
It will in the following be shown that USB A is serviced by USB Host Controller 27C9 port1, and that USB B and C both are serviced by USB Host Controller 27C8, the USB B by port1 and USB C by port2. In the bottom status line is indicated 3 Devices Connected, and the blue arrows indicates these devices are some Laptop build in USB devices, thus 0 Hubs connected, as no external devices connected. Note that the USB2 Enhanced Host Controller 27CC has 8 ports. All other has two ports.

Below is shown the situation where VNWA3 is connected to USB A serviced by 27C9 USB Universal Host Controller, occupying port1, 2 and 3 of the associated RootHub on port1 where the VNWA USB-hub now is connected.



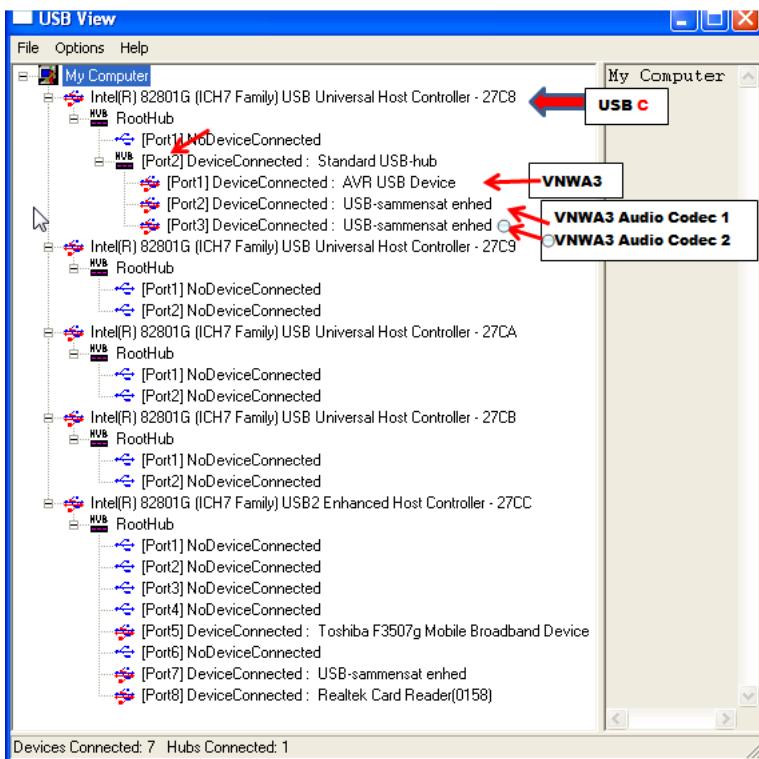
As said the 27C9 USB Universal Host Controller is servicing the three VNWA USB elements, the AVR USB Device and the two USB Codecs. If you are using the VNWA2 or your VNWA3 is not having the option board then there will be only one Audio Codec indicated connected.

Below is shown the situation where VNWA3 is connected to USB B serviced by 27C8 USB Universal Host Controller, occupying port1, 2 and 3 of the associated RootHub on port1 where the VNWA USB-hub now is connected.



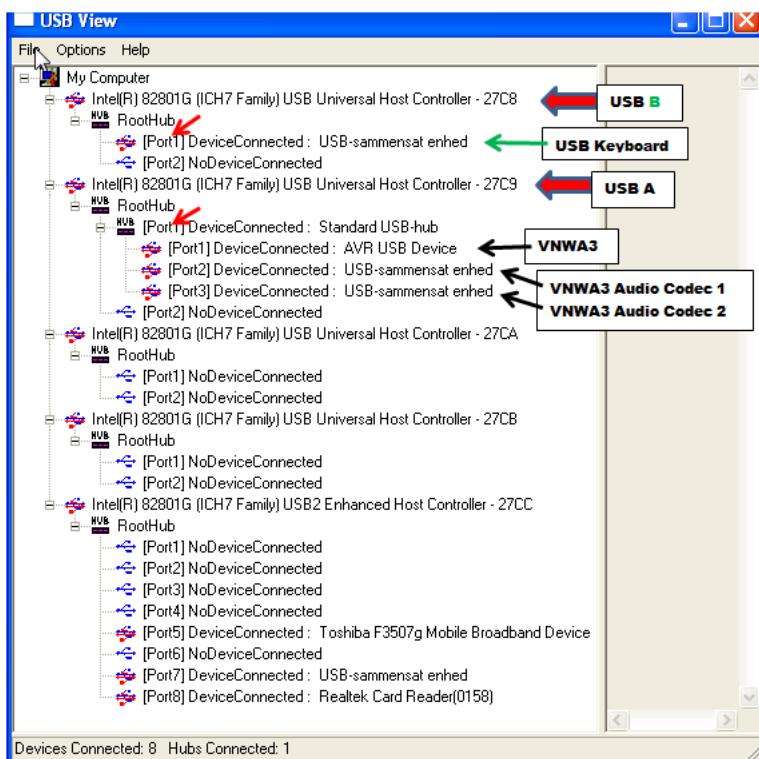
Likewise the 27C8 USB Universal Host Controller is now servicing the three VNWA USB elements, the AVR USB Device and the two USB Codecs. If you are using the VNWA2 or your VNWA3 is not having the option board then there will be only one Audio Codec indicated connected.

Below is shown the situation where VNWA3 is connected to USB C also serviced by 27C8 USB Universal Host Controller, occupying port1, 2 and 3 of the associated RootHub on port2 where the VNWA USB-hub now is connected.



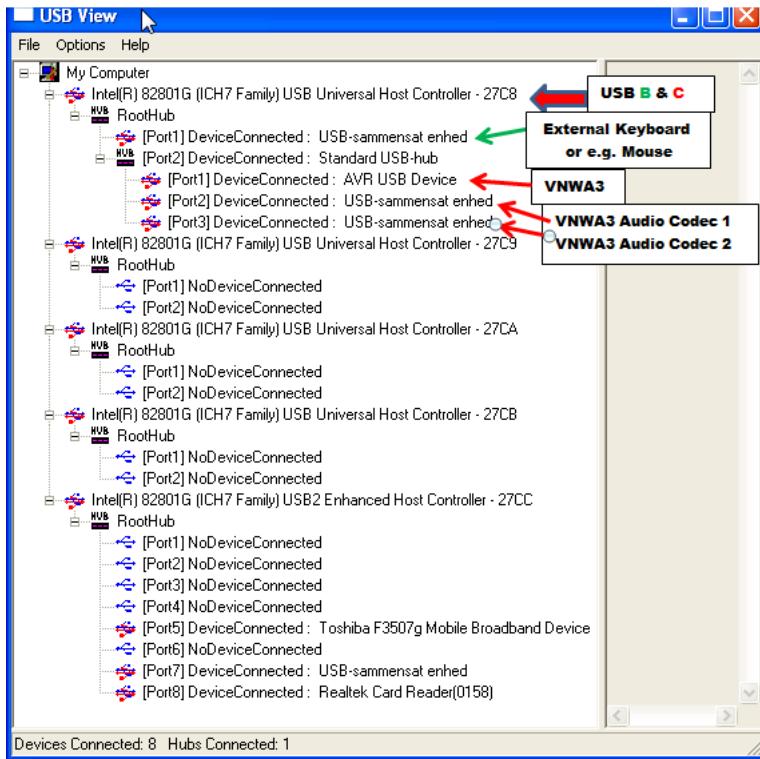
Likewise the 27C8 USB Universal Host Controller is now servicing the three VNWA USB elements, now via port2, the AVR USB Device and the two USB Codecs. If you are using the VNWA2 or your VNWA3 is not having the option board then there will be only one Audio Codec indicated connected.

Below is shown the situation where VNWA3 again is connected to USB A serviced by 27C9 USB Universal Host Controller, occupying port1, 2 and 3 of the associated RootHub on port1, where the VNWA USB-hub now again is connected. In addition an External USB keyboard is connected to USB B serviced by 27C8 USB Universal Host Controller.



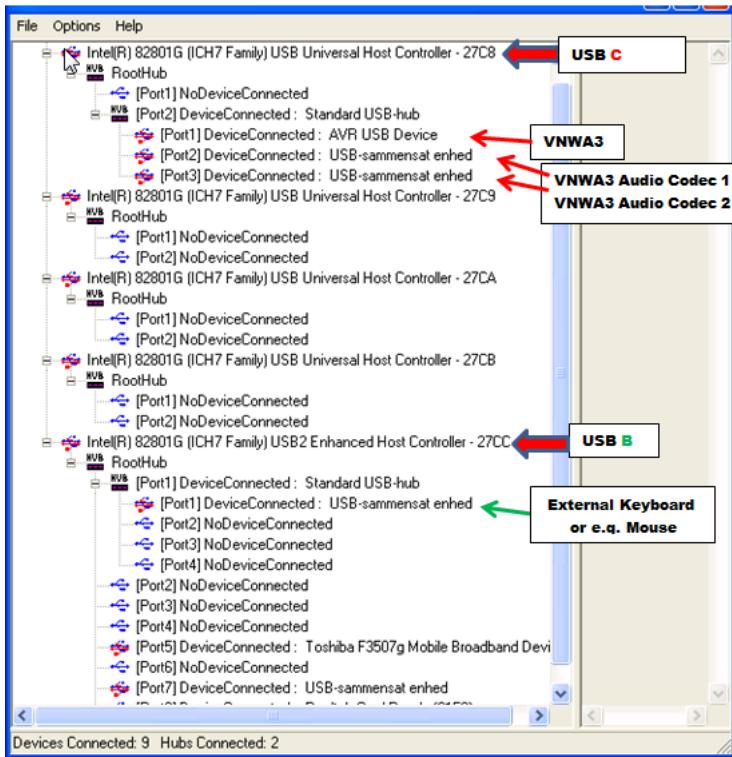
This situation is not causing any conflict between VNWA and other external USB devices, as VNWA is solely service by 27C9 USB Universal Host Controller.

Below is shown the situation where VNWA3 is connected to USB C as described and serviced by 27C8 USB Universal Host Controller port2, occupying port1, 2 and 3 of the associated RootHub on port2, where the VNWA USB-hub now is connected. In addition the external USB Keyboard is connected to USB B and thus sharing USB Universal Host Controller 27C8 with VNWA. This situation is causing problems with advanced features of the VNWA3, e.g. the Signal generator modulation will have "stuttering" sound. It might also be the cause for other observation like fragmented traces although not proven yet.



However this situation can be resolved if the USB keyboard is connected via an external USB Hub connected to USB B. See below...

Below is shown the situation where VNWA3 is connected to USB C and as described serviced by 27C8 USB Universal Host Controller port2, and occupying port1, 2 and 3 of the associated RootHub on port2 where the VNWA USB-hub now is connected. In addition an external USB Hub is connected to USB B, and the External USB Keyboard connected to the External Hub's port 1:



As seen the USB B is service by the 27CC USB2 Enhanced Host Controller and no longer sharing a RootHub with the VNWA3 and the Signal Generator Modulation clean as "a whistle".

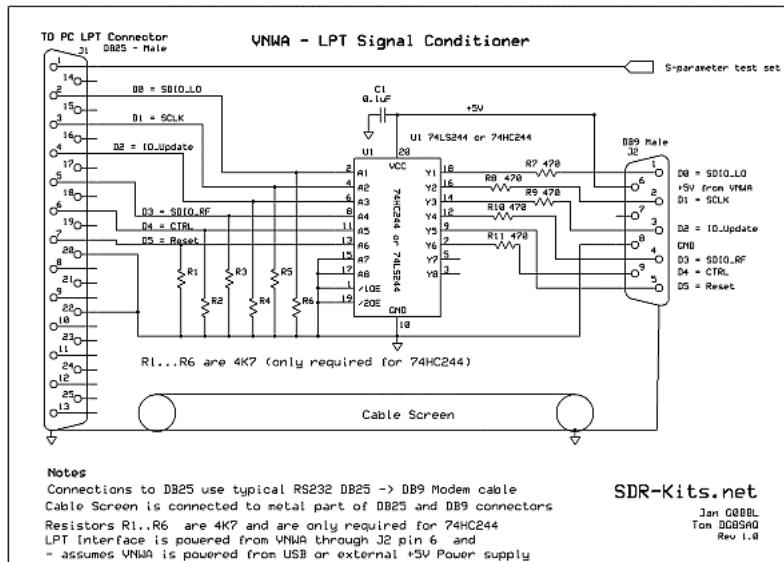
Conclusion:

It is essential for "playing the safe way" that VNWA is not sharing the USB Universal Host Controller Host, from which it is serviced, by any other External USB device. The USBView.exe is the tool to use for analyzing how YOUR PC is configured internally.

Kurt Poulsen, OZ7OU, 10/02/2013

LPT SIGNAL CONDITIONER

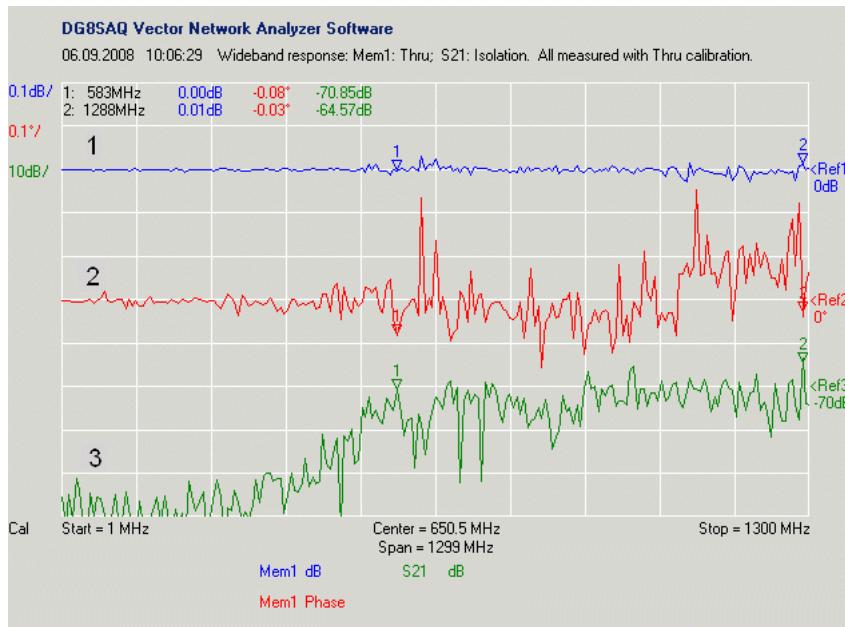
►►► **Note:** I have observed extremely slow signal rise times on some new PCI LPT interface cards. This is a problem when controlling the VNWA. Should you encounter unreliable communication between PC and VNWA in LPT mode, you might need the following signal conditioner, which sharpens the digital pulses.



DYNAMIC RANGE CONSIDERATIONS

A question often posed, is what is the achievable **dynamic range and accuracy** of the VNWA on transmission measurements.

The following picture taken from one of my QEX publications may serve as a reference:



The blue and red traces show the transmission characteristics of the Thru Calibration Standard. The green trace shows the measured noise floor with open RX input.

All measurements were done after a thru calibration (no crosstalk calibration was done!). The highest possible sweep time of 100ms/frequency point was used. The VNWA was powered via USB, controlled via LPT, the on board switching power supply of the VNWA was active and the on-board 16 bit SoundMAX sound card of my IBM R52 notebook with line-in exposed by a docking station was used. I used auto-clock rate to access the frequency range beyond 500 MHz.

If you can't reproduce the above displayed dynamic range, you may check the following items:

1. Have you selected proper **clock multipliers** to cover your frequency range? If you are not afraid of overclocking, use auto.
2. **Calibration measurements must be performed with the lowest sweep rate** (=longest time per data point) you ever want to use, otherwise the noise of the calibration measurement will dominate your later measurement and you won't see any noise reduction when further increasing the sweep time.
3. To obtain the highest possible dynamic range use the **slowest possible sweep time**.
4. With average 16 bit sound cards the dynamic range is limited to about 90dB. Somebody has demonstrated >100dB dynamic range measurements on the GroupsIO VNWA reflector using a decent **24bit sound card**.
5. Your **VNWA must be boxed and properly soldered to the box walls** to achieve the highest possible dynamic range.
6. **Poor solder joints** on the VNWA board might degrade the dynamic range.
7. If you want to perform high attenuation level transmission measurements, you may consider adding a buffer amplifier to the TX port. Of course, this will cause overload at low attenuation levels.
8. **Audio crosstalk between left and right audio channel** on your sound card may also degrade your dynamic range. The reason is, that there is always the very strong reference signal on one channel, while the RX signal might well be 90+dB down. If the sound card only offers 70dB interchannel isolation, you are stuck to 70dB dynamic range. You can test the interchannel isolations and more with the RightMark Audio Analyzer software from <http://audio.rightmark.org> and run a loopback test using the same audio cable as used on your VNWA. Also, the software Spectrum is useful for

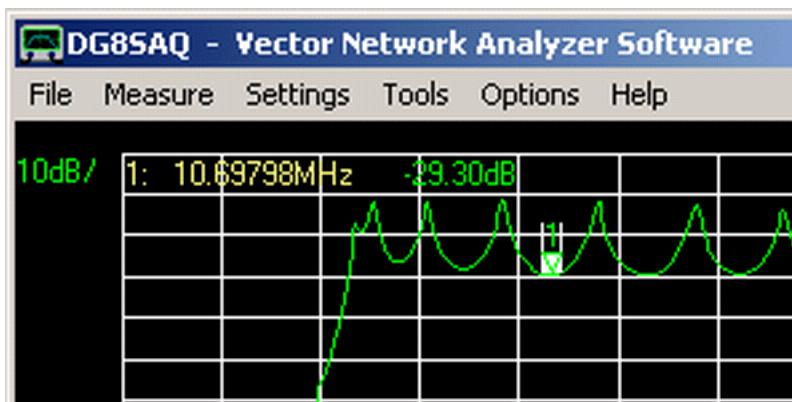
analysis of the VNWA audio signals. Take a look at the audio spectrum from the VNWA. Setup a L+R view side by side, startup the VNWA software and go into the audio test setup screen. Toggle Thru/Reflect mode in the audio level tab and you will see the impact of the stereo seperation.

NON-TRANSPARENT MARKERS

From Software version V34.0 on below issues should be fixed.

Status prior to software version V34.0

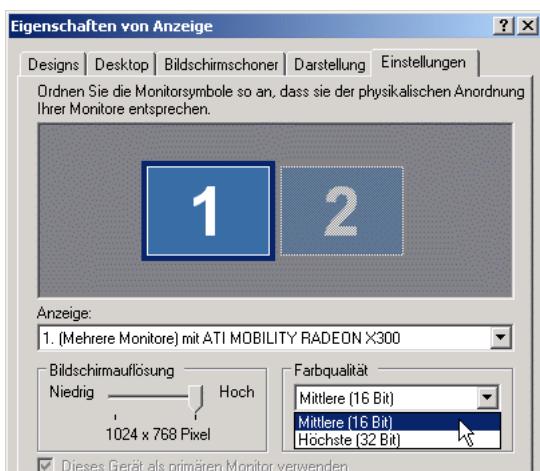
In some cases, the **trace markers are not displayed in a transparent way**, but rather with a white box around them as can be seen below.



The root cause is not clear to me. One way to remedy this is to reduce the color depth of Windows via the control panel to < 32bit:

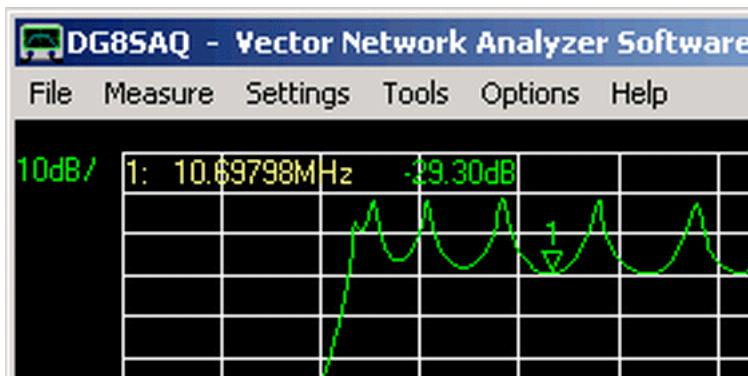


Select a color depth of 16 bit:



►►► **Note:** You must **terminate and restart the VNWA software** after this change!

Now, the markers are displayed correctly:



An alternative approach is to let **VNWA automatically switch the color depth** to 16 bit upon program start and restore the original value upon program termination.

To enable this hidden setting, you must manually edit the file **VNCommon.ini** found in the program directory with a text editor, e.g. with the Windows Notepad:

```

VNCommon - Editor
Datei Bearbeiten Format Ansicht ?
ScreenDepth=-16
RealZ0=6.20000000000000E+001
ImagZ0=5.30000000000000E+001
FFTSource=0
FFTwindow=0
Timeunit=2
TStart=0.00000000000000E+000
TStop=1.00000000000000E-009

```

Change the first line to
ScreenDepth=16

►►► **Note:** No blank characters are allowed within the ini-files.

Positive values specify the desired color depth. With negative values, the command will be ignored.

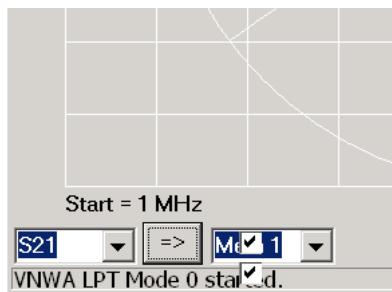
►►► **Note:** Some monitors require some time for syncing upon program start, if automalical switching is activated.

CONTROLS OUTSIDE WINDOW, MISSING CONTROLS

From Software version V32 on below issues should be fixed. V32 onwards allows for 96dpi fonts (standard) as well as 120dpi fonts (= big fonts).
Also, arbitrary Windows styles are supported.

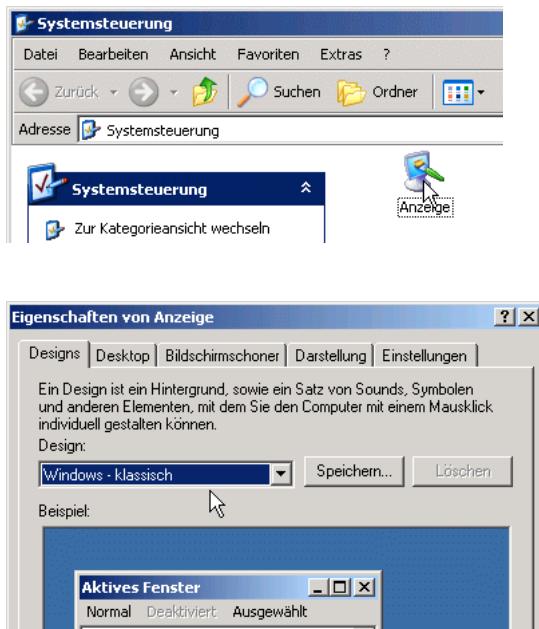
Status prior to software version V32

For some display settings, software controls appear in odd positions or even not at all, as can be seen for the case of the two trace select checkboxes in below example:

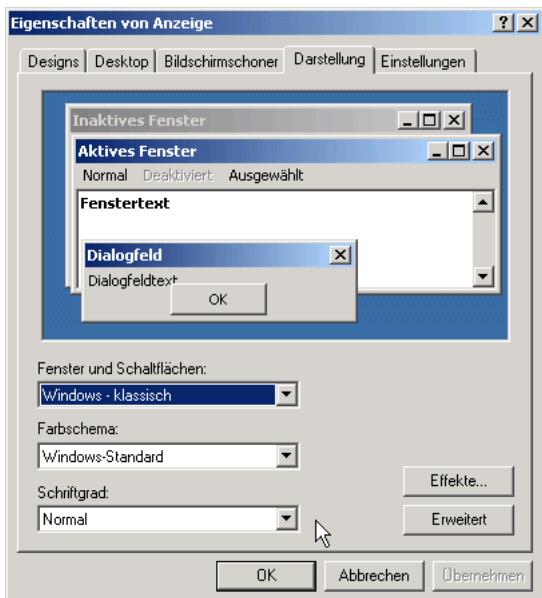


This is caused by the fact, that VNWA was designed for a certain Windows style and VNWA cannot adapt to dramatic style changes like increased font size.

In order to display all VNWA windows in a proper way, it is recommended to choose the following display settings in the Windows control panel:

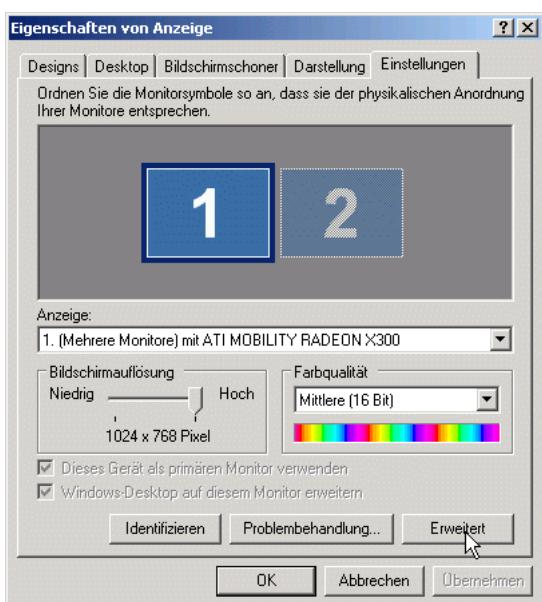


I do use the **Windows - Classic**, but I have verified proper appearance with the **XP-Design** as well.

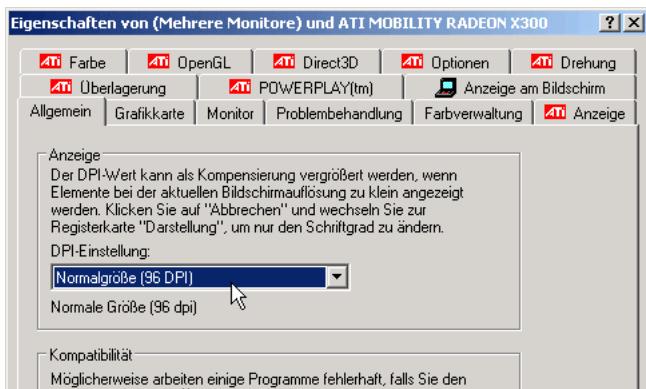


An important setting is the **Font Size** setting above, left of the mouse arrow. You must set this to **normal**. If a bigger font size is selected, all controls become bigger and won't properly fit onto the VNWA windows any longer. Todate, VNWA cannot handle different font sizes.

There is another setting, that influences the font size. In order to access it, you must select the extended display settings:



In the general display settings, the **DPI Settings** must be set to **96 DPI!**



AUDIO ISSUES

Windows Sound Mixer Issues

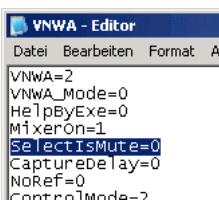
Upon program start, VNWA is designed to enable the selected audio capture device and set the volume controls to the previously stored values for Windows versions XP and older.

Depending on sound card drivers, this procedure sometimes fails, e.g. for some drivers, the **capture device is deactivated instead of activated upon program start**.

Inverting the Mute Logic

The mute-logic can be inverted inside the VNWA software by pressing the key combination **shift+ctrl+x** while the main VNWA window is in focus (VNWA version 36.2.b or newer).

An alternative way to achieve this is to edit the file **VNWA.ini** found in the program directory with an ASCII editor like the Windows Notepad. Change the line "**SelectIsMute=0**" (see below) to "**SelectIsMute=1**". The VNWA software must not be running while doing this modification.



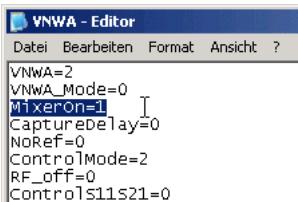
```
VNWA=2
VNWA_Mode=0
HelpByExe=0
MixerOn=1
SelectIsMute=0
CaptureDelay=0
NoRef=0
ControlMode=2
```

Disabling the Sound Mixer

A different approach to solve the mute-problem is to **disable the Windows sound mixer altogether**.

This can also be performed from within the VNWA software by pressing the key combination **ctrl+alt+x** while the main VNWA window is in focus (VNWA version 36.2.b or newer).

Alternatively, you must edit the file **VNWA.ini** found in the program directory with an ASCII editor like the Windows Notepad:



```
VNWA=2
VNWA_Mode=0
MixerOn=1
CaptureDelay=0
NoRef=0
ControlMode=2
RF_Off=0
Controls11s21=0
```

Change the line

MixerOn=1

to

MixerOn=0

►►► **Note:** No blank characters are allowed within the ini-files.

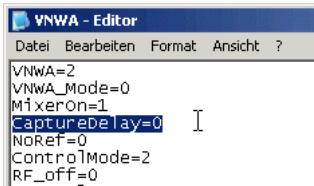
By doing so, VNWA will not touch any sound settings any longer except for selection of the capture device. In this case, you must activate the capture device manually and use the Windows sound mixer or an equivalent software of your sound card's manufacturer to adjust the recording volumes.

Note, that this switch will have no effect on Windows Vista, as **the mixers are always disabled on Vista**.

Capture Startup Issues

Some sound cards fade in the volume upon capture start. Clearly, this is poison for this accurate measurement system, as the measurement results around the start frequency are questionable in this case.

The hidden Capture Delay software switch is a workaround to bypass this problem. In order to activate it, you must edit the file **VNWA.ini** found in the program directory with an ASCII editor like the Windows Notepad:



```
VNWA=2
VNWA_Mode=0
MixerOn=1
CaptureDelay=0
NoRef=0
ControlMode=2
RF_off=0
```

Increase the number in the **CaptureDelay** line. A value of 100 means, that the acual measurement is delayed by 100 milliseconds after opening the capture device. Of course, this will slow down the sweep rate for continuous sweeps by the specified value.

FREQUENTLY ASKED QUESTIONS

Q1: What does "No Sync Found" mean?

A1: During sweep start, the VNWA does not detect a synchronization pulse.

Possible causes:

- the selected sound device/audio codec is the wrong one
- VNWA sound devices swapped (VNWA3E only)
- the wrong audio codec channel is selected as reference channel
- the sound device is not configured for stereo usage (Vista, W7 only)
- external system clock is selected but no external clock source is connected
- defect in audio chain (very unlikely)

Possible solution:

Download the latest software version or at least VNWA36.3.0, open **Options-Setup**, press the **auto-setup** button on the very bottom of the **Audio Settings tab**. This should automatically fix all audio codec setup problems.

Q2: What does "Overload" mean?

A2: During data acquisition at least one of the audio codec inputs was driven into saturation.

Possible causes:

- codec recording level settings too high
- codec unrelated to VNWA hardware selected
- sweep start frequency below 25kHz selected

Possible solution:

Download the latest software version or at least VNWA36.3.0, open **Options-Setup**, press the **auto-setup** button on the very bottom of the **Audio Settings tab**. This should automatically fix all audio codec setup problems for standard measurements with start frequency >25kHz.

For start frequencies below 25kHz see page "Very Low Frequency Measurements". Either manually reduce the codec recording level(s) (for wideband sweeps with start frequency 0Hz) or reduce the codec sampling rate (for narrowband audio sweeps).

Q3: Why do I see a cascade of "Range Check Errors"?

A3: This indicates a USB communication problem between PC and VNWA controller.

Possible causes:

- broken USB cable
- PC unable to provide sufficient DC power
- USB connector disconnected while VNWA is sweeping

Possible solution:

Try a different cable, a different USB port, a self powered USB hub in between VNWA and PC.

Q4: What does "Error code x on node y" mean?

A4: This indicates a USB communication problem between PC and the VNWA audio codec(s).

This error is issued by the Windows audio driver and the VNWA software can only detect but not avoid it.

Possible causes:

- broken USB cable
- PC unable to provide sufficient DC voltage for codecs
- USB connector disconnected while VNWA is sweeping

Possible solution:

Try a different cable, a different USB port, a self powered USB hub in between VNWA and PC.

Q5: Why does Windows 10 all in a sudden no longer detect the VNWA codecs?

A5: The most likely reason is an automatic Windows update that has deactivated the codecs. On rare occasions hardware problems may be the cause.

Possible causes:

- Deactivated codecs after a Windows update
- Windows security settings no longer allow VNWA software to access the codec after Windows update.
- PC unable to provide sufficient DC voltage for codecs.

Possible solution:

- Right-click the Windows recording devices list and select "show deactivated devices". Activate the VNWA codecs there.
- Check Windows security settings to make sure the VNWA software is allowed to access recording devices.
- Try a different USB cable, a different USB port, a self powered USB hub in between VNWA and PC.

Q6: How do I provide debug information in case of problems?

A6: The VNWA software has a built-in facility to compile and email debug information. Note, that no debug data will be emailed without your confirmation.

a) In case of general issues not related to audio codecs:

1. Make sure that the VNWA is connected to the PC.
 2. Launch the VNWA application
 3. Open main menu Help-About.
 4. There, open the system info tab and press the email button.
If you use Outlook, a mail window with attached debug file will open.
Add any comments you wish, e.g. identify yourself and describe your problem.
If you do not use Outlook, then locate the debug zip file in the VNWA installation folder send it manually to dg8saq@darc.de.
- b) In case of audio codecs:**
1. Make sure that the VNWA is connected to the PC.
 2. Launch the VNWA application
 3. Press ctrl-d. This will make the software enter debug mode after the next software restart.
 4. Close and restart the VNWA application
 5. Go to Setup-Audio Settings and press the Auto Setup Audio button
 6. Close setup and go to main menu item Help-About
 7. There, open the system info tab and press the email button.
If you use Outlook, a mail window with attached debug file will open.
Add any comments you wish, e.g. identify yourself and describe your problem.
If you do not use Outlook, then locate the debug zip file in the VNWA installation folder send it manually to dg8saq@darc.de.

VERIFICATION OF PROPER OPERATION

Note, that the VNWA does not contain user adjustable components or adjustable firmware settings. So, it does not require regular factory recalibration in order to maintain its measurement accuracy.

If you are in doubt of proper operation of your VNWA, you may repeat the final commissioning test and compare to the measurement sheets you have received together with your instrument upon purchase.

The test is described in the chapter "VNWA Final Commissioning Test".

A more in depth test procedure will be described in the following. This is a systematic step by step guide to verify proper interaction of the VNWA and the corresponding software. The procedure will detect both hardware problems and erroneous software settings.

0. Are you running a supported operating system?

- NOT supported: Win95 and older
- partially supported: Win98, Win98SE, WinME (USB mode only, NOT LPT mode)
- Fully supported: Win2000, WinXP, Win Vista, Win 7, Win8, Win10.
- LPT interface is still supported, but only available on WinXP and older. LPT drivers have to be installed manually.

1. Start with a fresh installation of the latest VNWA software release to be found on the SDR-Kits web page at <https://www.sdr-kits.net/DG8SAQ-VNWA-software-documentation-user-guide/>.

2. Start the software and perform the setup as described in section "Getting Started".

The tests assume, that nothing is connected to the VNWA test ports unless stated otherwise.

Also, all calibrations should be switched off unless stated otherwise.

The software is to be run in VNWA mode, NOT in spectrum analyzer mode.

Averaging must be switched off.

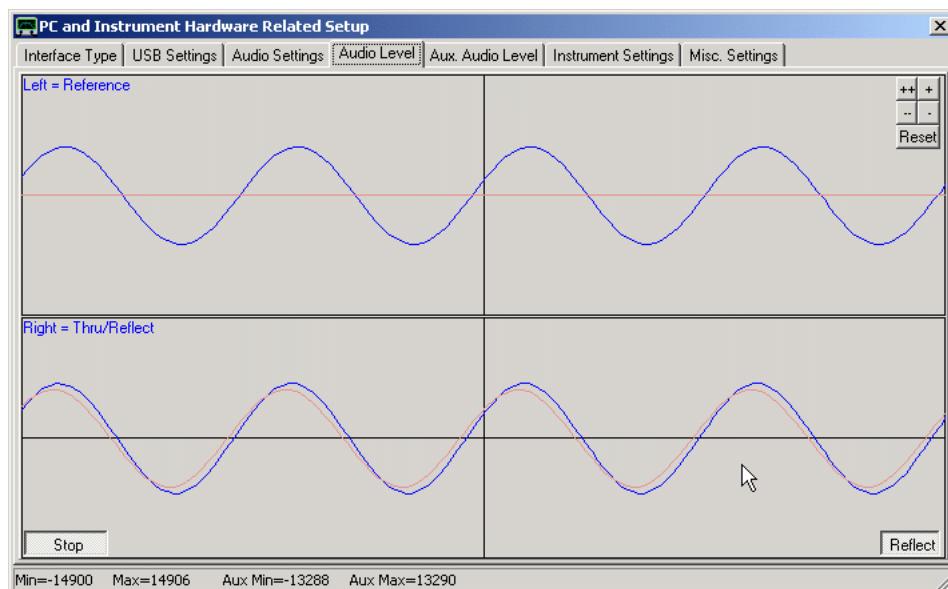
All of the above will automatically be fulfilled for a clean new installation.

Check the following:

- correct sound card and capture device selected?
- correct reference channel selected?
- audio level set to about 50% of full scale level?
- does the Audio Level screen show two blue sine-waves for reflect or one blue sine wave for through?
- auto clock multipliers selected if overclocking is desired?

A first simple functional test at the predefined test frequency of 5MHz can be performed in the setup - **audio level tab**.

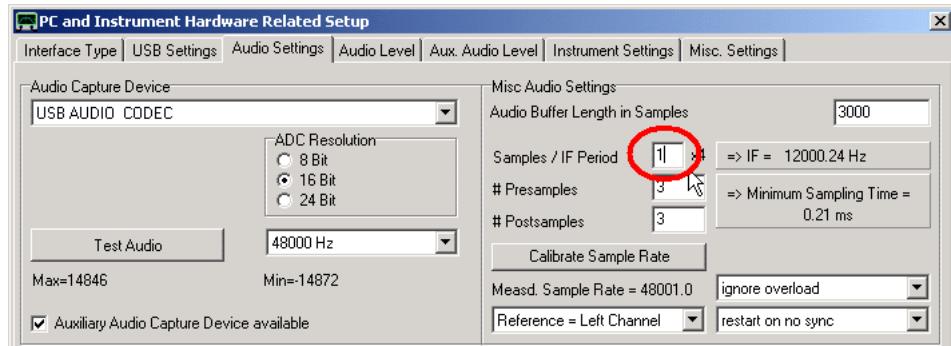
2.a) Press the **test audio** button and make sure you have selected **reflect** and nothing is connected to the VNWA TX port. You should see two similar blue sine waves. One blue sine wave is the Reference signal, the other is the Reflect signal, which is in phase to the reference signal for the open TX port:



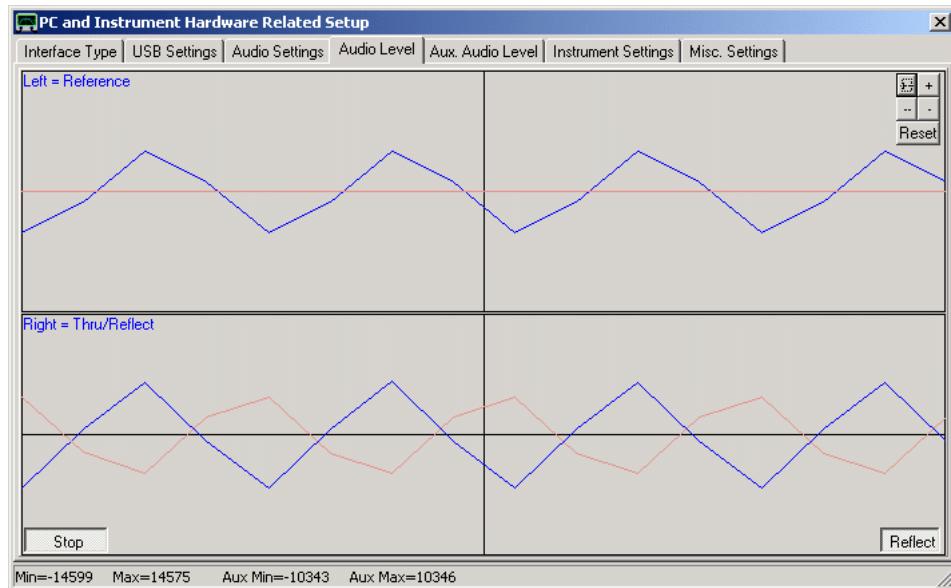
Note, that for VNWA3E you additionally see a faint red sine wave which is the reference signal (and a flat line for the

thru signal) sampled with the second built-in codec. These signals can be observed in greater detail in the Aux. Audio Level tab. Note that the main and aux reference signals need not stay in perfect sync as the codecs may run on slightly different sample rates.

Note, that the sine waves in above screen shot are sampled with the default setting of 10x4 samples per IF period, which makes them look smooth and sine-like. If you reduce the number of samples per IF period to 1x4 as is required for ultrafast sweeping ...

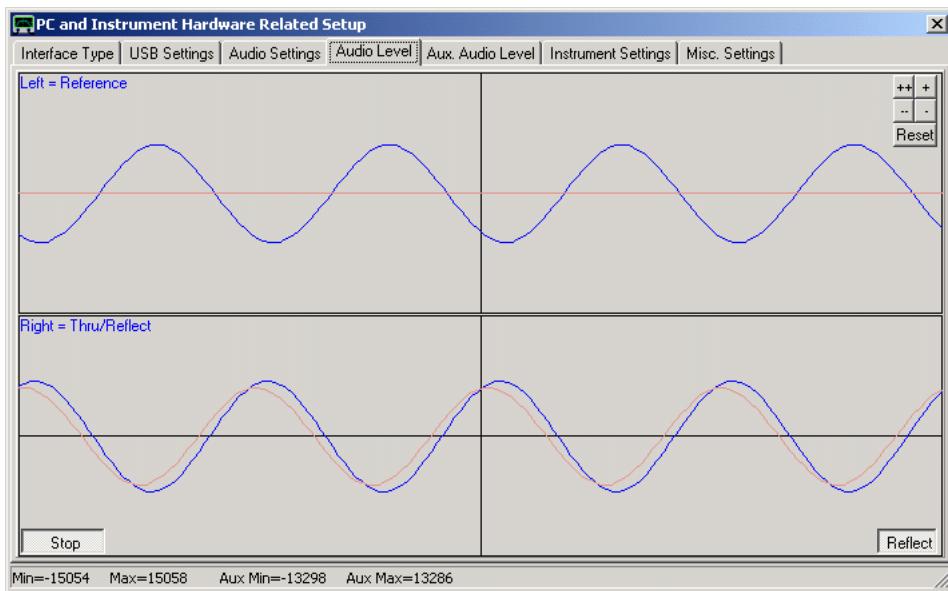


... the sine waves will rather look triangular or trapezoidal ...

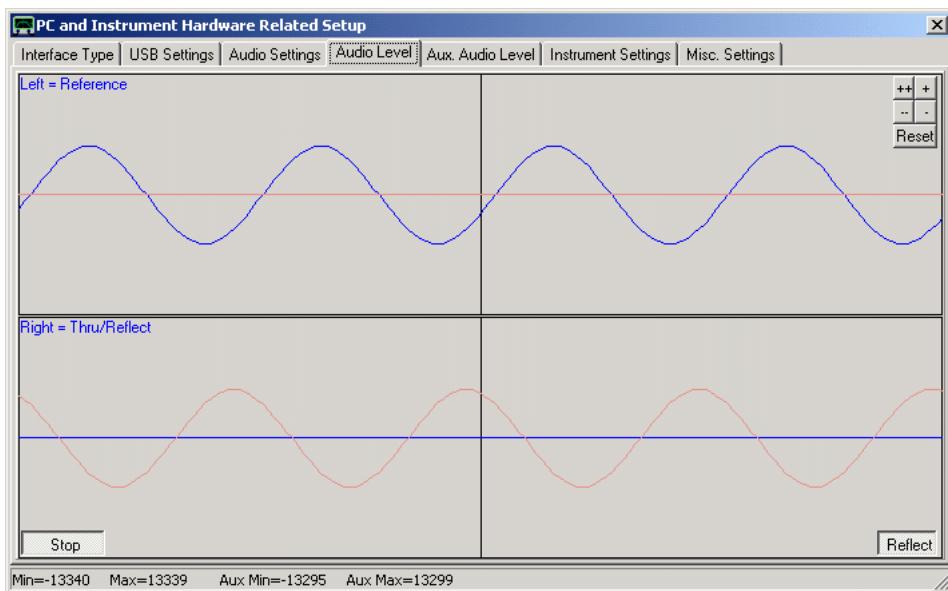


... which is perfectly ok! They are still sine waves which are approximated in the graph with 4 straight lines connecting the 4 sampled data points per period. You may also proceed the tests with this setting.

2.b) Next, connect a short to the VNWA TX port. This will cause the blue Reflect signal to be phase shifted by 180° to the blue Reference signal as can be seen below:

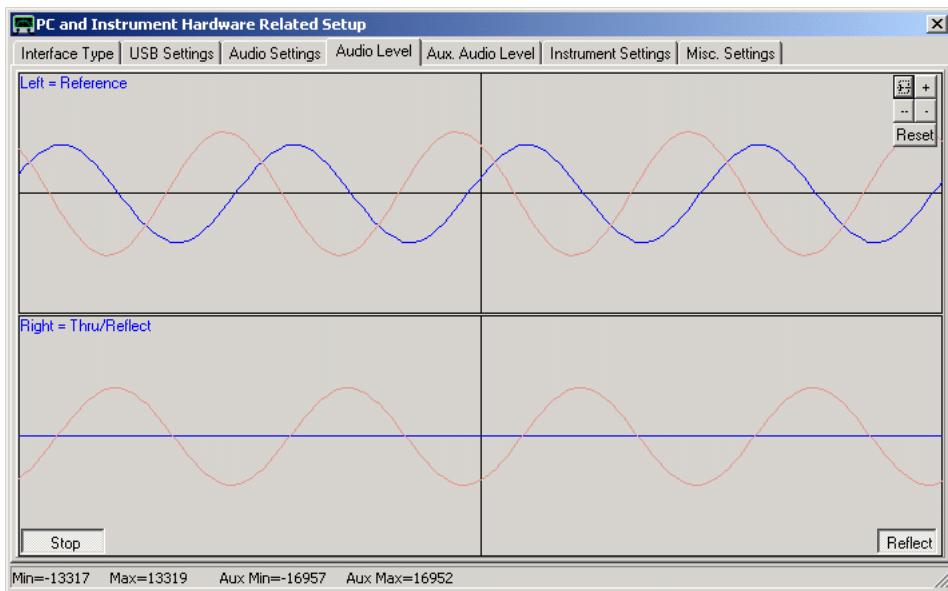


2.c) Next, connect a 50 Ohms load to the TX port. As the load absorbs all incident power, the blue Reflect signal becomes zero:

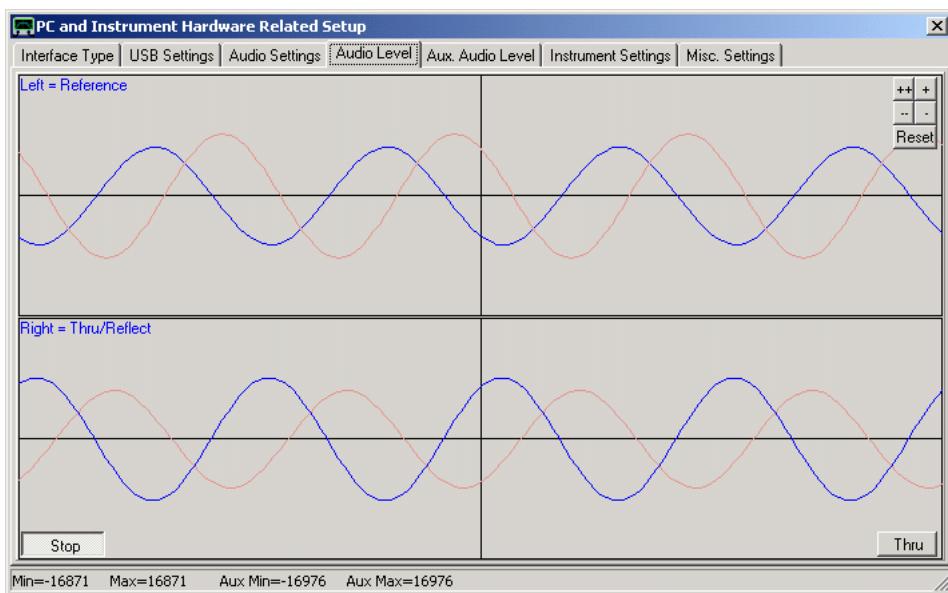


With the above tests you have verified that the reflection bridge works properly. From the last image (load) you can deduce, that indeed the left (=top) channel is the reference channel as is selected since it never changes.

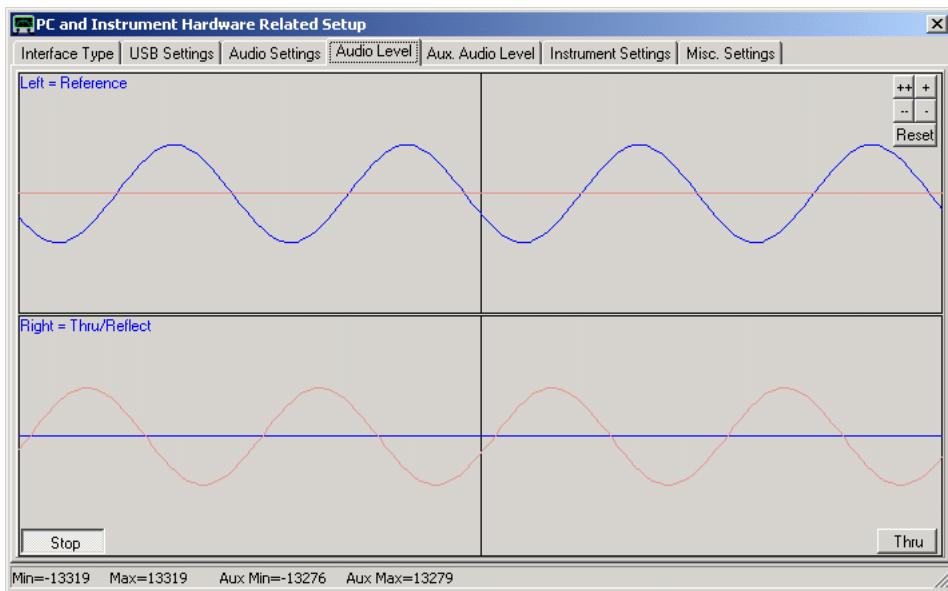
2.d) Connect TX port with RX port by means of a short coaxial cable. As the RX port is designed to have 50 Ohms impedance, you should obtain the same result as in c) above, except that you should see two faint red sine waves now:



2.e) Next toggle the button showing reflect to **Thru**. On the top, a blue sine wave appears again, which is the Thru signal transmitted through the coax cable from d). Thru and Reference signal happen to be out of phase which is of no relevance for the proper function of the VNWA:



2.f) Disconnect the coax cable between TX and RX port. You should see the upper blue and lower red thru signal disappear, as no signal is fed into the RX port any longer:



Tests d)-f) prove, that the RX signal chain works properly.

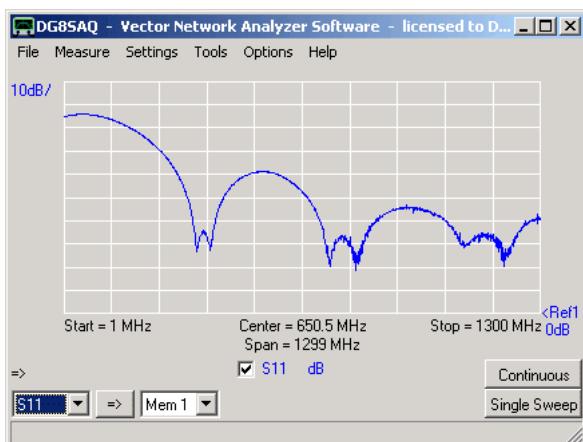
►►► **Note:** Should you always see the same sine wave on both channels, then your operating system (Vista, Windows7, Windows 8) assumes, that your sound device is mono. You must instruct Windows to open it as a stereo device in this case.

3. Proper operation of audio capture

►►► **IMORTANT:** The following tests 3.a)...4.a) must be performed **WITHOUT CALIBRATION** or with calibration switched off, otherwise test results will be nonsense!

3.a1) Go to Options-Setup-Misc Settings and tick "do not normalize to reference channel". Note, that the phase information will be lost with this setting and thus Smith charts do not make sense.

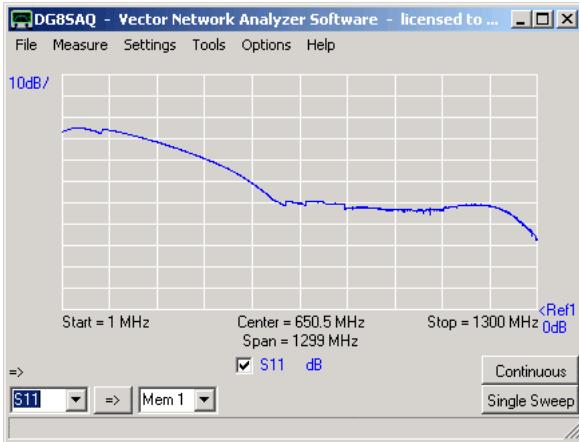
Close the setup, remove all connectors from TX and RX ports, display S11 only in dB mode and perform a single sweep with about 1ms per data point and 1000 data points. Using the **default clock multipliers of 10/11**, the result should look like below:



Dips occur at integer multiples of the DDS clock frequencies, i.e. $n \cdot 10 \cdot 36\text{MHz}$ and $n \cdot 11 \cdot 36\text{MHz}$, where the DDS output power drops to zero. To overcome these signal dips, **select auto clock multipliers**.

►►► **Warning:** You are using **auto clock multipliers** at your own risk. By doing so, **you select to severely overclock** your DDSes from the allowed 400MHz up to 750MHz.

3.a2) The same test as in a1), but using auto clock multipliers looks like this:



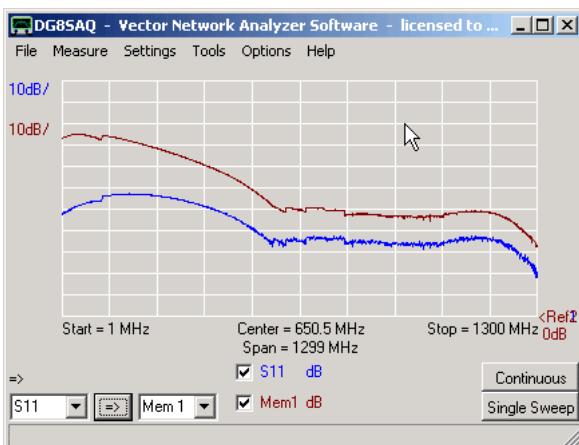
The dips are gone and you see steps, where the clock multiplier values are being switched.

All the following results are obtained by using auto clock multipliers.

Save the sweep in a2) to Mem1, display Mem1 as well and repeat the sweep several times to verify that the result is stable and repeatable.

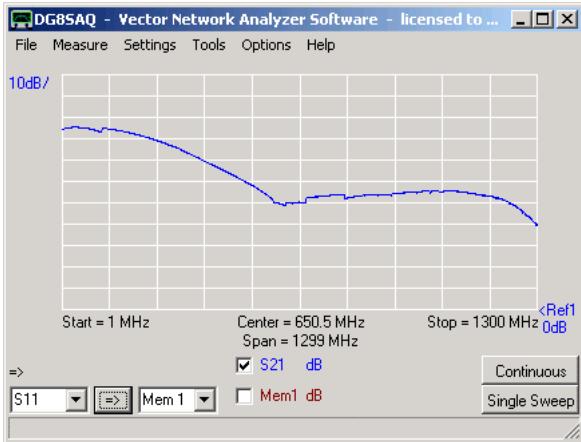
This plot directly shows the available signal amplitude in ADC readings. Note, that the plot shown is valid for a 16bit sound card. If a 24bit sound card is used, amplitudes will be higher by a factor of 256. Also note, that the steps in the trace prove proper operation of auto clock multiplier switching if selected.

3.b) Next, connect a 50 Ohms load to the TX port and repeat the sweep:



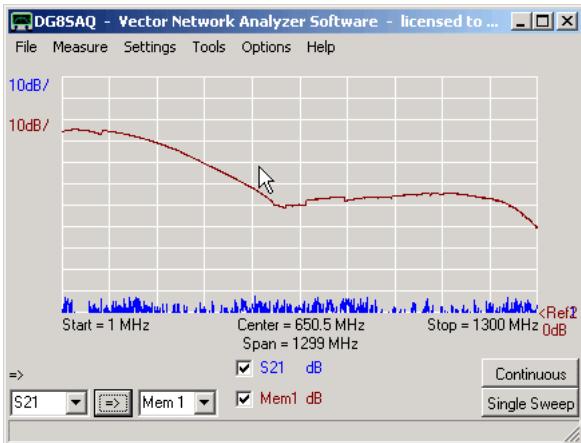
The trace should move down in amplitude. The difference between the previous trace (Open) and the current trace (Load) is the directivity of the VNWA bridge. The above plot proves, that the bridge is operational. If you see no trace change when sweeping the load, you might have missed to select the correct audio channel as reference channel or you might have selected the wrong polarity setting for the S11/S21 switch.

3.c) Next, connect VNWA TX port with the RX port by means of a coaxial cable. Select S21 and do a sweep:



The result should be similar to the S11 result with open TX port. The trace shows the available signal in transmission. Save the sweep to Mem1, display Mem1 for comparison and repeat the sweep several times to verify stability and repeatability.

3.d) Next completely remove the coaxial cable from TX to RX and sweep again:

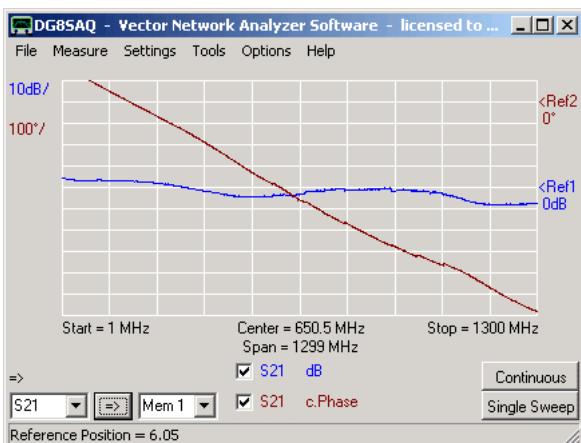


This test shows the noise floor of the RX. The achievable dynamic range in transmission is the vertical distance between the two traces. Note, that the test was performed at a high sweep rate and thus the noise floor is rather high. This test proves that the RX is operational.

4. Proper operation of calibration

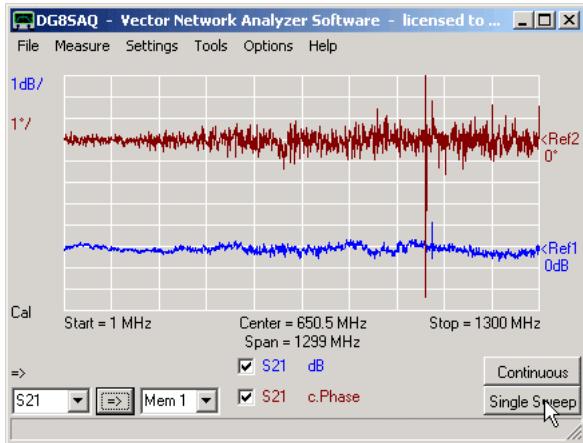
For the following tests you **MUST UN-CHECK "don't normalize to reference channel"** in the setup-misc settings.

4.a) Connect TX port with RX port and do an S21 sweep. Now the result should look similar to this:



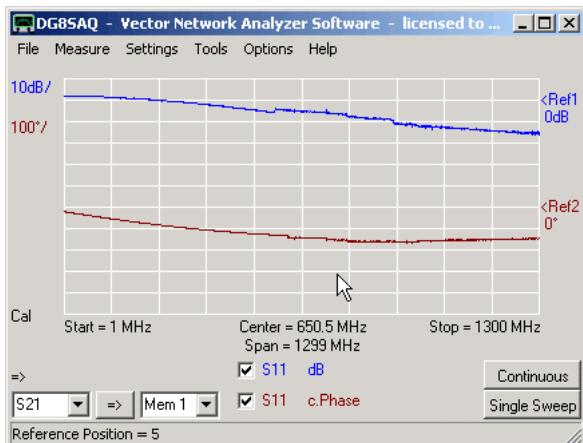
The amplitude should be around 0dB and there should be phase information available. This proves that the reference signal chain is operational. Repeat the sweep several times to verify that the result is stable and repeatable.

4.b) Next, do a Thru-calibration (Thru only, no Crosstalk, no Thru Match Calibration!) and repeat the S21 sweep after the Thru-calibration. You should get two straight lines for amplitude and phase then:



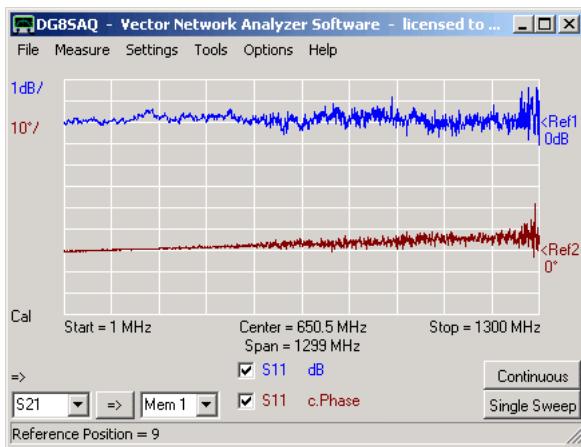
Repeat the sweep several times to verify that the result is stable and repeatable.

4.c) Next, disconnect the coaxial cable, display S11 and perform a sweep. Note, that you have not done a reflect calibration yet. This is what you should approximately get, an amplitude around 0dB and a fairly constant phase:



This proves that normalizing also works for the reflect signal. Repeat the sweep several times to verify that the result is stable and repeatable.

4.d) Next perform an SOL calibration and repeat your sweep of the open TX port. You should get two straight horizontal lines at 0dB and 0°:



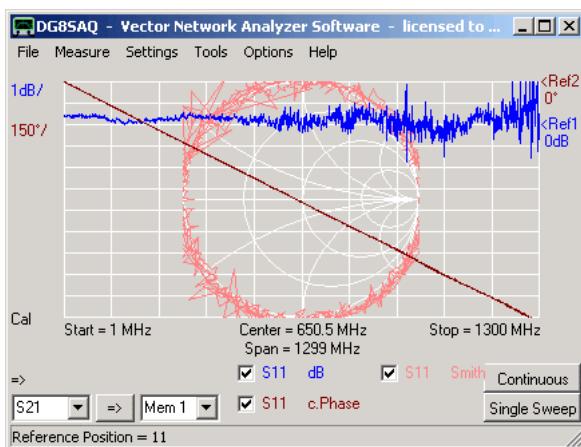
This proves that the reflect calibration is operational. You should also verify, that you can reproduce the reflection coefficients of your other two cal standards (Short, Load).

Repeat the sweeps several times to verify that the results are stable and repeatable.

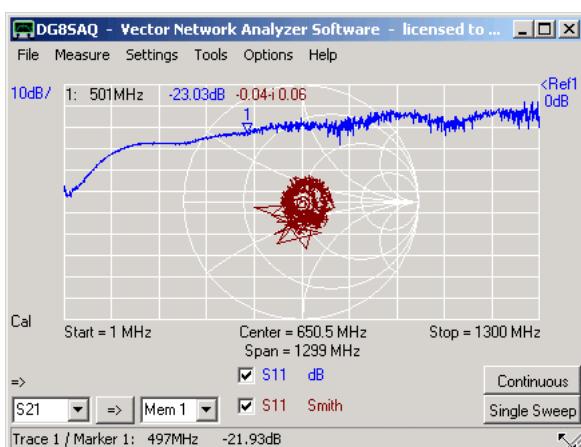
Note: For the **Load**, the phase information will be arbitrary, as the reflection coefficient of an ideal load is zero, which has arbitrary phase. Or put in math: $0 \cdot \exp(j\beta) = 0$ for any angle β .

So, all phase information you will get to see for the load is noise. Simply ignore it.

4.e) Next, connect a short coaxial cable (about 10cm) to the TX port and leave the other cable end open. Sweep and observe, that you should still obtain a magnitude of 0dB but a linear phase increase which leads to a circular trace in the Smith chart:



4.f) Next connect the TX port with the RX port by means of the short coax cable and perform an S11 sweep. This way, you measure the input reflection coefficient of the RX port:



The RX reflection coefficient should be better than -20dB up to 500MHz.

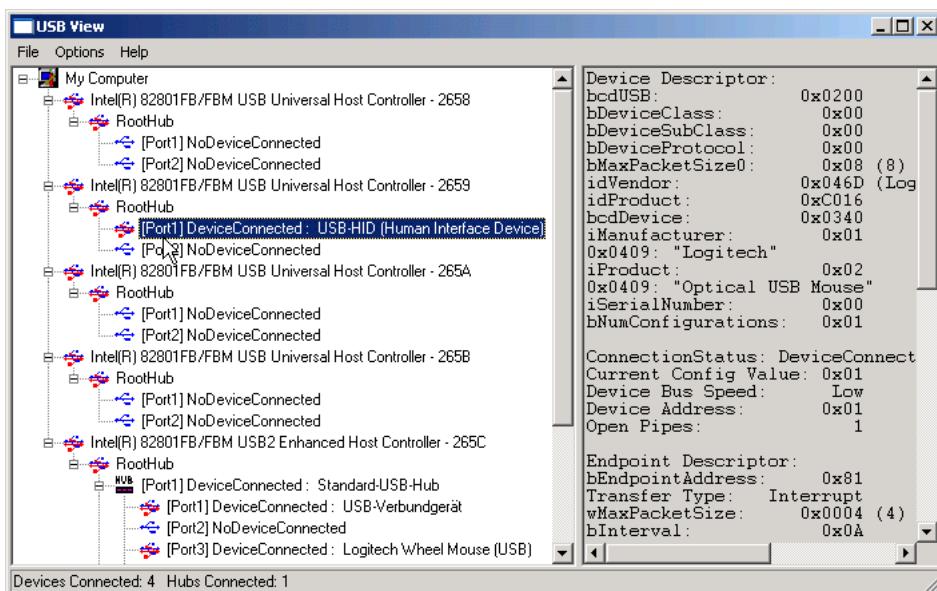
If your tests fail at a certain step, please cite the exact step, at which the test procedure failed.

USB TROUBLESHOOTING GUIDE

Using the Microsoft utility "**USB View**" it is possible to detect problems of USB devices. In case the DG8SAQ USB_VNWA device is not properly detected by Windows, do the following diagnosis procedure.

Troubleshooting procedure

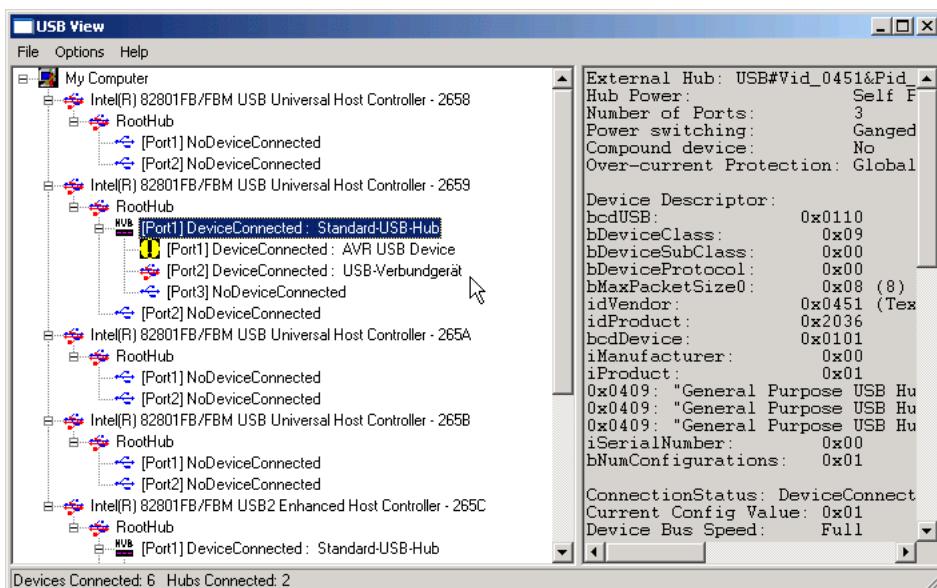
1. Download the Microsoft utility "USB View".
2. Start "USB View" and find the USB outlet in "USB View" where you want to plug in your USB_VNWA interface by e.g. plugging in a known good USB device like a mouse or a memory stick. Make sure you activate the options "Auto Refresh" and "Config Descriptors". In below example a USB_HID mouse is detected. The grey field on the right gives more details about the highlighted USB device. This is proof that your USB PC interface works.



3. Plug in your USB_VNWA interface into the very same USB outlet and observe the change in "USB View". As seen below, the following devices should be detected:

- a standard USB hub
- an AVR USB device (the VNWA controller)
- a USB audio codec

Make sure, the display of "USB View" is refreshed after replugging.



Should at least one of the above devices fail to show operational, then you likely have a hardware problem on your USB_VNWA interface or, if the AVR device fails to show, you may be confronted with a corrupted flash memory. See the next page on how to deal with corrupted flash.

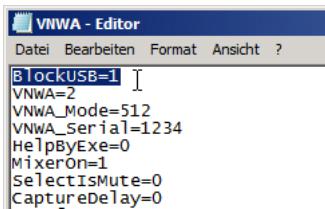
If the above tests perform positive and yet Windows refuses to detect a new device, then you might be confronted with a Windows driver problem. You could try to install the driver manually from the Windows control panel.

RECOVERING CORRUPTED FLASH MEMORY

If a power failure or a PC crash is encountered during flashing a new VNWA firmware, the firmware will become corrupted and non-functional. Since the USB core of the firmware is to some extent protected, there is still a chance that the firmware can be restored via USB. If the USB core is still fully functional, the AVR-device will show up in usbview, as shown in the previous section.

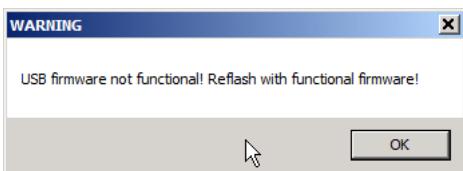
Case 1: VNWA2 or VNWA3, AVR device still shows up in usbview:

Since the firmware is corrupted, VNWA will issue multiple errors upon program start. To avoid these error messages, open the file **VNWA.ini** with a text editor (e.g. notepad.exe) and add the line "BlockUSB=1":



```
BlockUSB=1  
VNWA=2  
VNWA_Mode=512  
VNWA_Serial=1234  
HelpByExe=0  
Mixeron=1  
SelectIsMute=0  
CaptureDelay=0
```

Save and close VNWA.ini and restart the VNWA software. A message telling that the firmware is not functional will be shown:



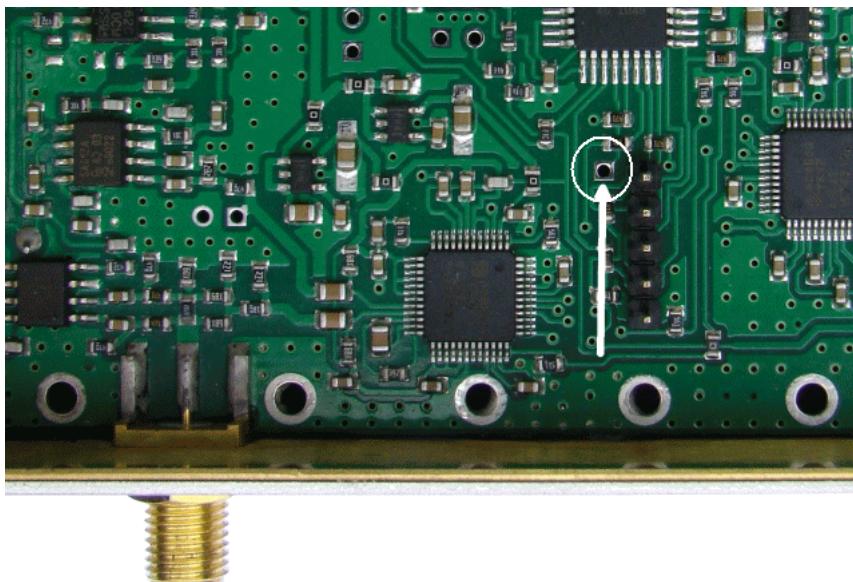
Confirm and VNWA will start. Since all USB functions except reflash and rescan the bus are blocked now, VNWA will issue no more error messages. Go straight to "Setup"- "USB Settings" and try to reflash the firmware with the version that failed to flash previously.

If you are lucky, your VNWA will behave normally after closing and restarting the VNWA application then. This procedure has proved successful in one case already!

Case 2: VNWA3, AVR device no longer shows up in usbview, but USB core routines are still intact:

Since the core routines are write-protected to some extent, there is a chance that they are still intact even though the AVR device no longer shows up on the USB bus.

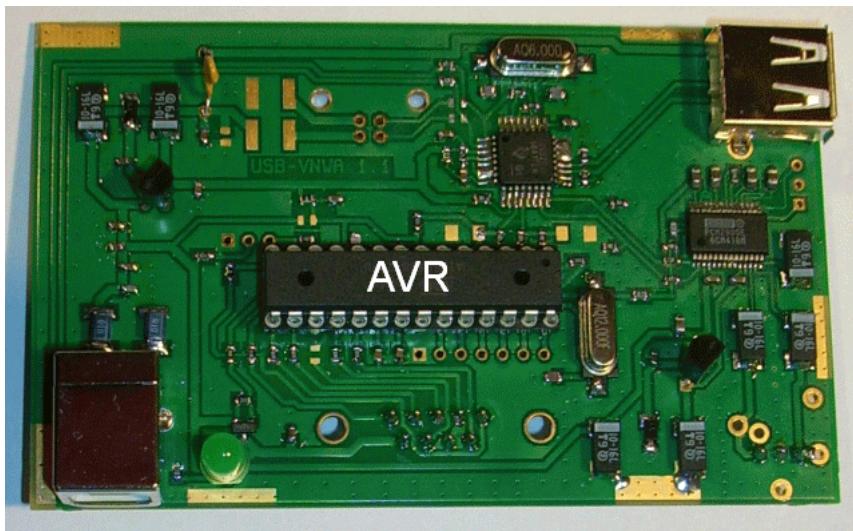
From firmware 5.12 (five dot twelve!) onwards you can force the USB core to skip the likely corrupted initialization code by pulling the following test point to ground during power-up:



It is recommended to use a test cable with a crocodile clip on one end and a pointed test tip on the other. Disconnect the VNWA from the computer. Then clip the croc to the brass chassis while pressing the test tip to above marked test point. While do, reconnect the VNWA to the computer. Now, you may remove the test cable. Use usbview to check if this measure has restored USB connectivity of the AVR USB device. If this measure was successful, **do not start the VNWA software** immediately, but proceed as described above in case 1. If this procedure does not solve the problem, proceed as described in case 4.

Case 3: VNWA2, AVR device no longer shows up in usbview, device must be reprogrammed using an in circuit programmer (ISP):

The simplest way to fix this problem is to obtain a new AVR chip from SDRKits and replace the corrupted one, see image below. When inserting the new chip, make sure that none of the pins get bent!



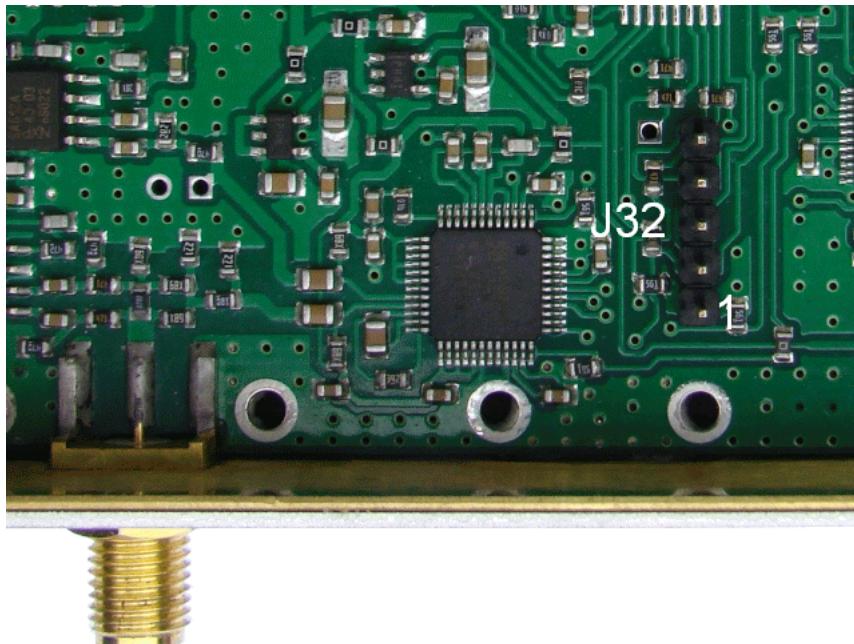
If you have a programmer to flash Atmel AVR microcontrollers, you can reprogram the chip yourself.

►►► **IMPORTANT:** Before reflashing the AVR, make sure to read out the EEPROM so you can restore it after reflashing. The EEPROM contains your license key. Flash the latest v4.* firmware. Note, that firmware v4.15 (four dot fifteen) is newer than v4.9 (four dot nine). After flashing, restore your EEPROM with the data previously saved.

Case 4: VNWA3, AVR device no longer shows up in usbview, device must be reprogrammed using an in circuit programmer (ISP):

Generally, you have to return your VNWA3 to SDRKits to get it reprogrammed in this case.

If you have an ISP programmer to flash Atmel AVR microcontrollers, you can reprogram your VNWA yourself, though. This is achieved by connecting the programmer to the VNWA3 ISP interface J32:



J32 ISP interface pinning:

Pin 1: Ground

Pin 2: Reset

Pin 3: SCK

Pin 4: MOSI

Pin 5: MISO

►►► **IMPORTANT:** Before reflashing the AVR, make sure to read out the EEPROM so you can restore it after reflashing. The EEPROM contains your license key. Select device Atmega328P in your programming software. Flash the latest v5.* firmware. Note, that firmware v5.16 (five dot sixteen) is newer than v5.9 (five dot nine). After flashing, restore your EEPROM with the data previously saved.

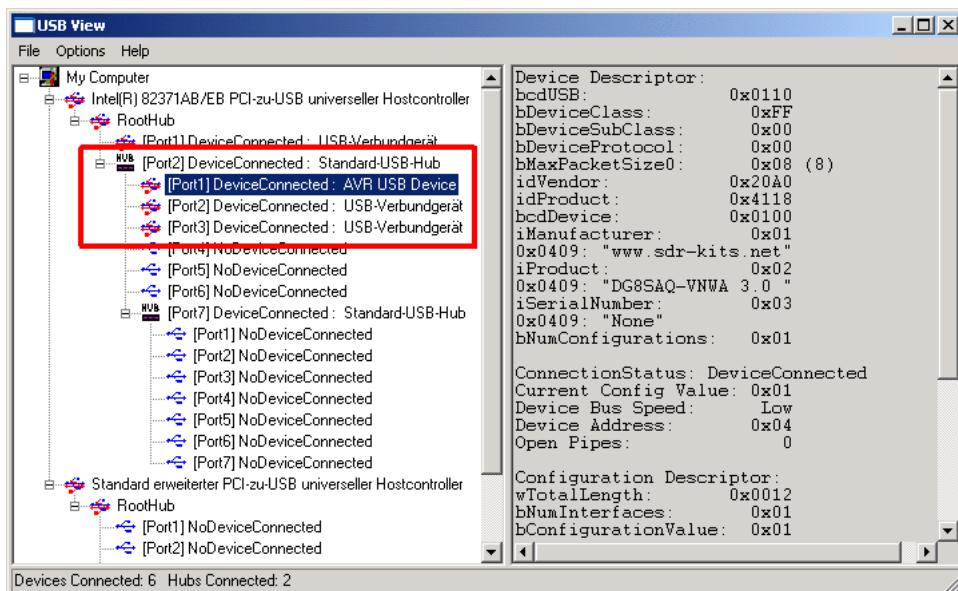
HARDWARE TROUBLESHOOTING

This section describes how to diagnose specific VNWA hardware failures without using additional test equipment hardware.

1. Check Connectivity

Use usbview or a similar software as described in chapter "USB Troubleshooting Guide" to detect if the VNWA hardware connects to your computer.

You should see the VNWA (AVR USB Device) and one (VNWA2, VNWA3) or two (VNWA3E) sound chips being connected to the same USB hub:

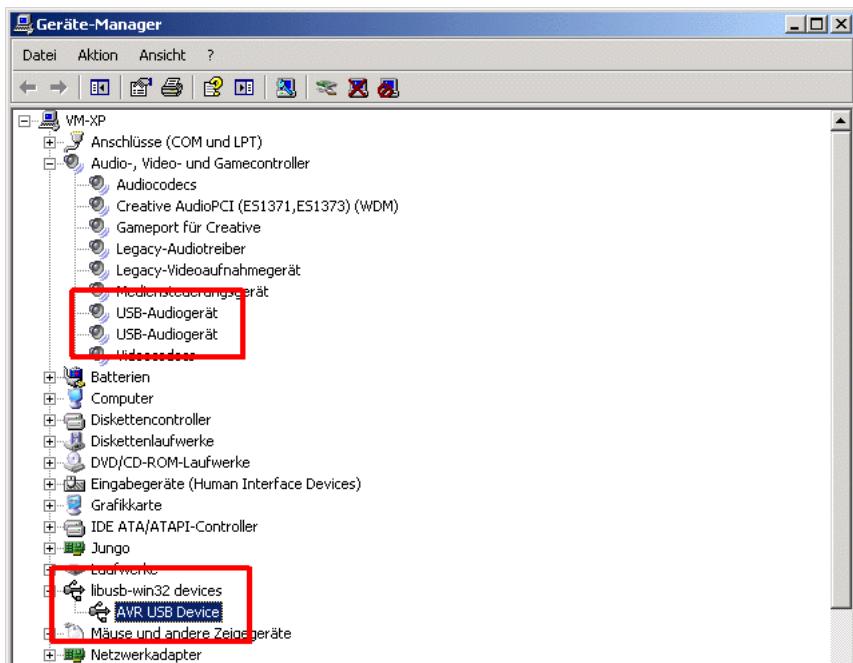


Note, that the above test does NOT require any VNWA software or driver software being installed.

If successful, you may ...

2. Check if hardware drivers are properly installed

Open the Windows device manager and check if both the VNWA device and the VNWA USB sound device(s) are shown as operational:

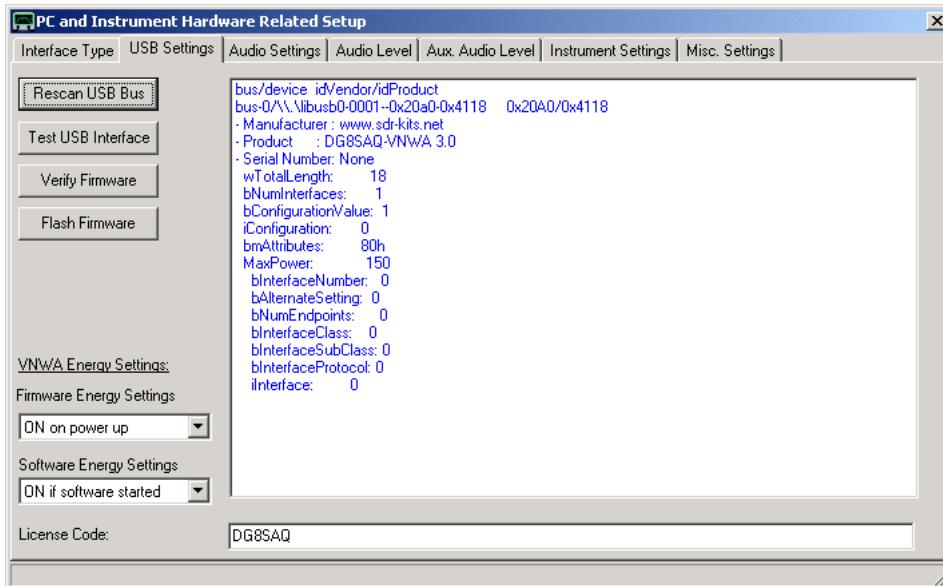


If successful, you may...

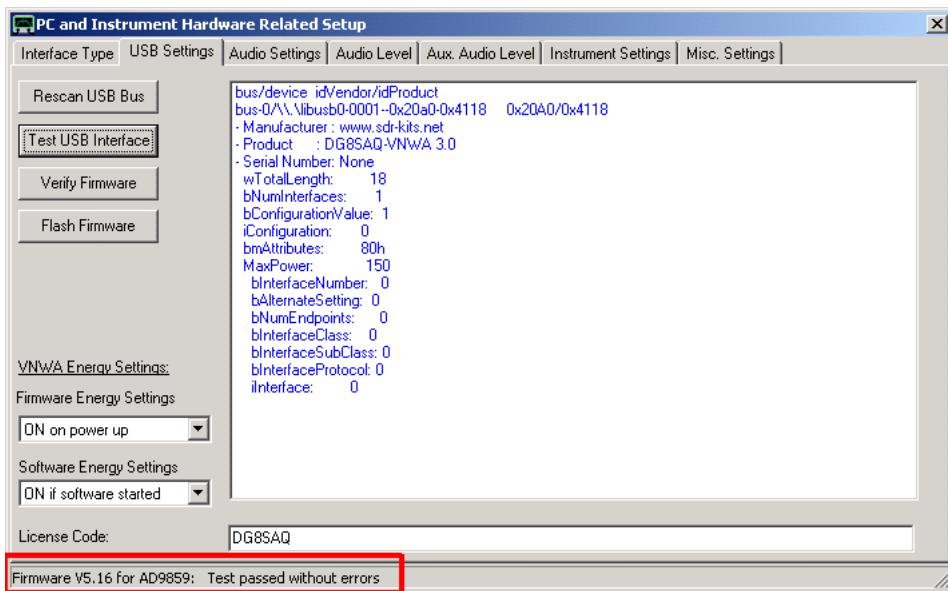
3. Check if communication with the VNWA is possible

Launch the VNWA software and go to Options-Setup-USB Settings. Make sure, that your license code is entered correctly.

Press "Rescan USB Bus". You should observe your VNWA being detected:



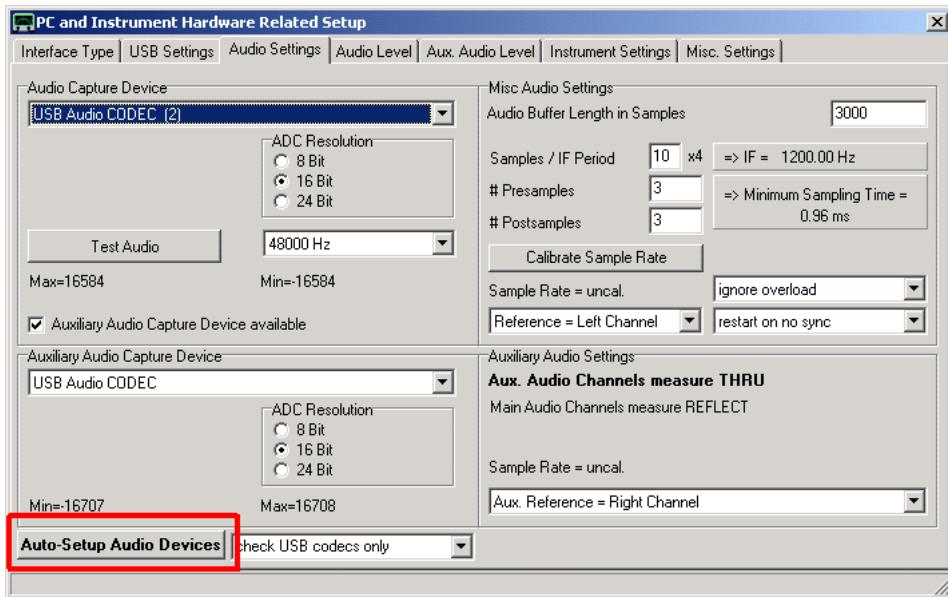
Next, press "Test USB Interface". You should see a blue progress bar indicating the progress of the test and upon completion information about your firmware and test result in the bottom status line:



If successful, you may ...

4. Perform a total system test

Go to the Audio Settings tab and press the button **Auto-Setup Audio Devices**:



This automatic setup procedure also serves as a sensitive hardware test. Follow the on screen instructions. If the setup passes, your VNWA hardware is very likely ok.

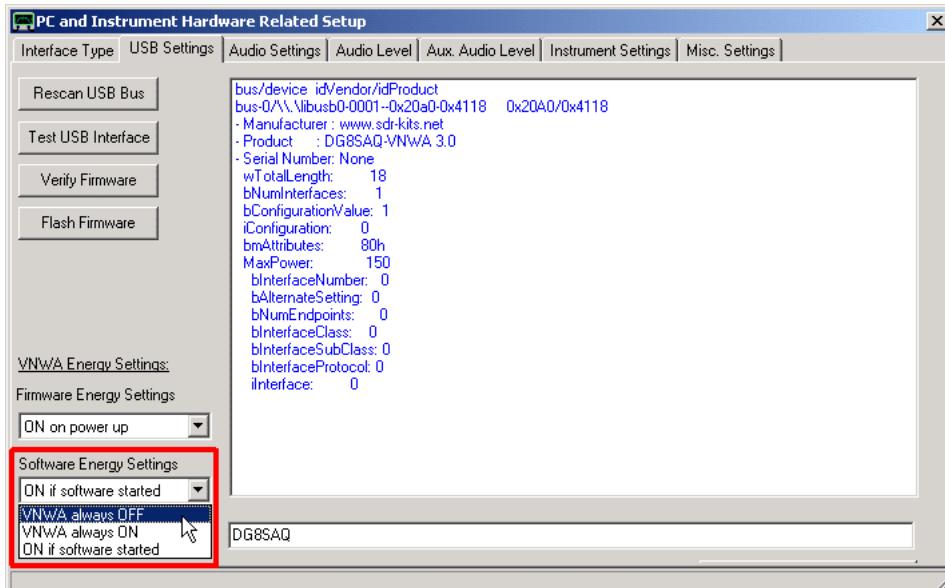
Failure of this test may have several reasons:

- a) You might have forgotten to disconnect all VNWA RF ports prior to the test.
- b) You might have failed to set your VNWA sound devices to 48 kHz / Stereo.
- c) You are encountering a hardware problem.

The following tests deal with diagnosing hardware problems more specifically:

5. Checking power supply

Go to Options-Setup-USB Settings and toggle the **Software Energy Settings** between OFF and ON:



While doing so, observe the LED on the back side of your VNWA, which should toggle between OFF and ON in turn:



When done with the test, make sure that your energy settings are "ON if software started".

6. Testing audio codec(s)

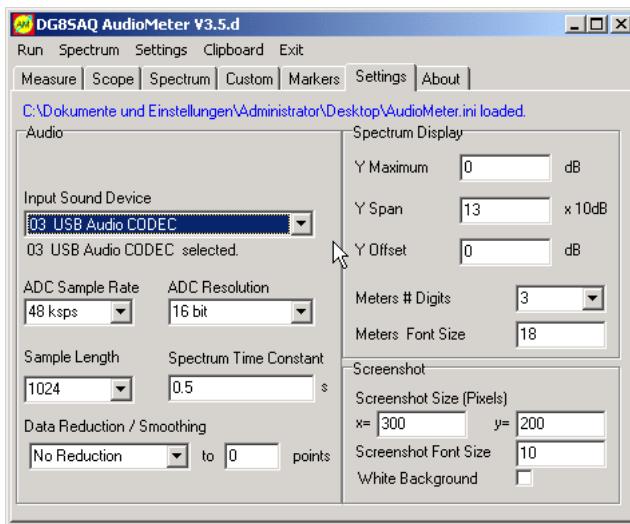
Make sure, that the VNWA is connected to your PC, but the VNWA software is NOT running and the VNWA power LED on the back side is OFF (see previous test).

Also, use a coaxial cable to **connect your VNWA TX port with the RX port!**

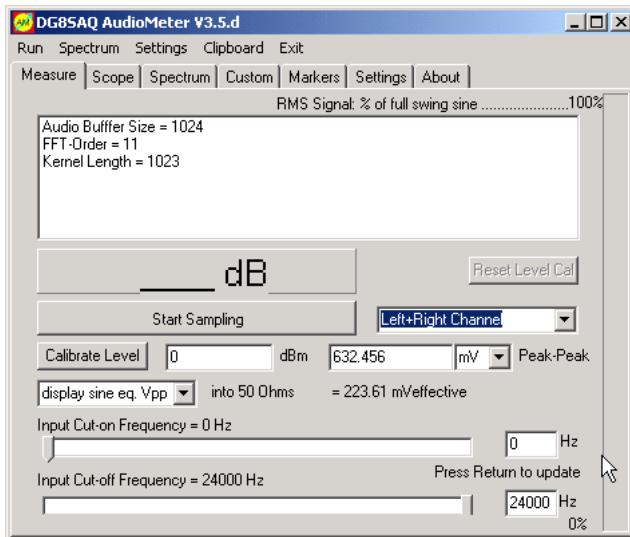
For this test you need the DG8SAQ AudioMeter3 software, which can be downloaded from <http://www.sdr-kits.net/DG8SAQ/other/AudioMeter3.zip>.

Launch it and select settings as described in the following.

Select a sound device belonging to your VNWA:

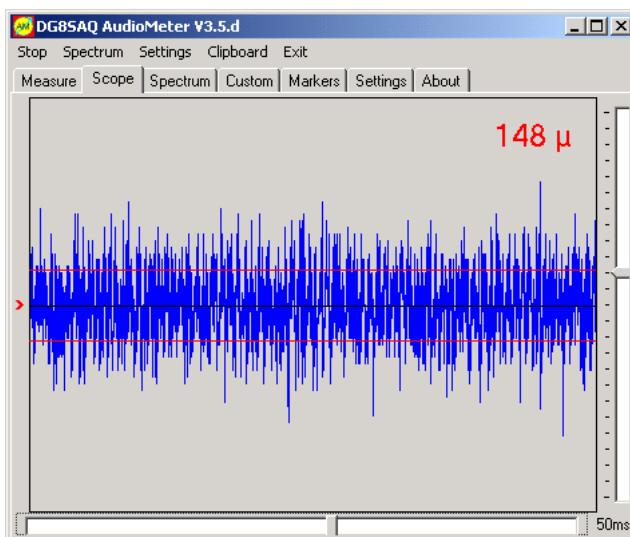


Select "Left + Right Channel" and make sure the sliders on the bottom are set as shown:



Press main menu "Run" and change to the Scope tab.

Use the sliders there to adjust the time axis and the amplitude axis. You should see low level noise coming from the sound device:



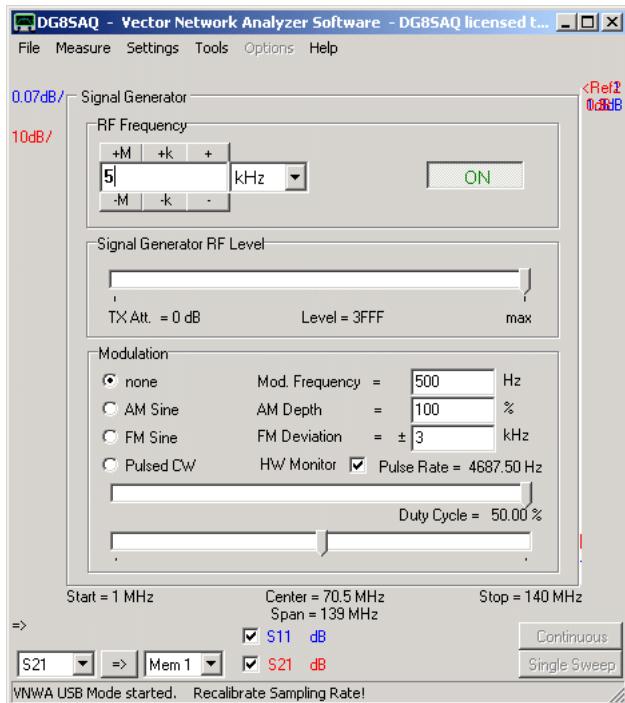
If you have a VNWA3E, repeat this test with the second VNWA sound device.

If your audio codec(s) are operational, you may...

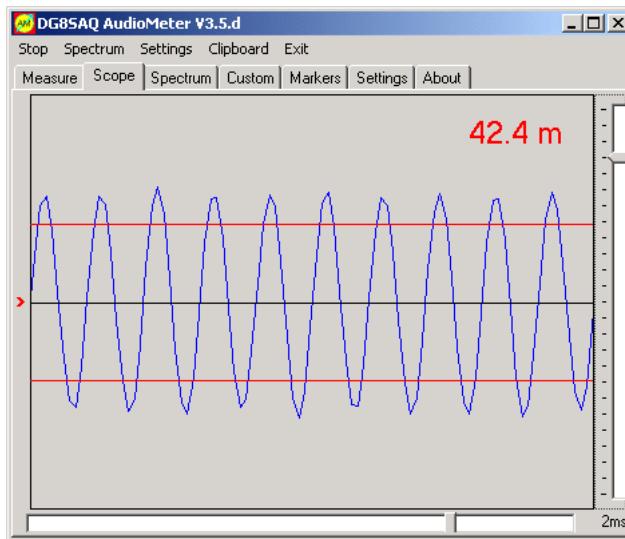
7. Test the RF signal source

Make sure that the VNWA RX port is still connected to the VNWA TX port.

With AudioMeter3 running as in the previous test, launch the VNWA software in "Signal Generator Mode" and set it up to produce a continuous 5kHz signal at maximum output level:



Now, you should see a strong signal in AudioMeter:



Check the Spectrum tab to observe that the signal you are seeing is indeed located at 5kHz:

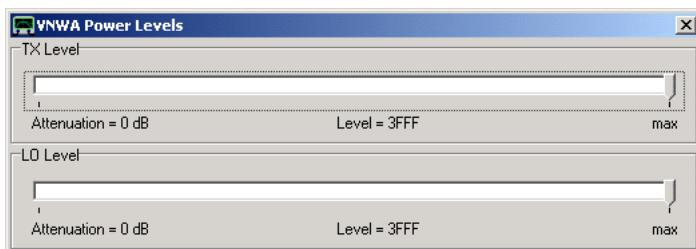


If you have a VNWA3E, repeat this test with the second VNWA sound device.

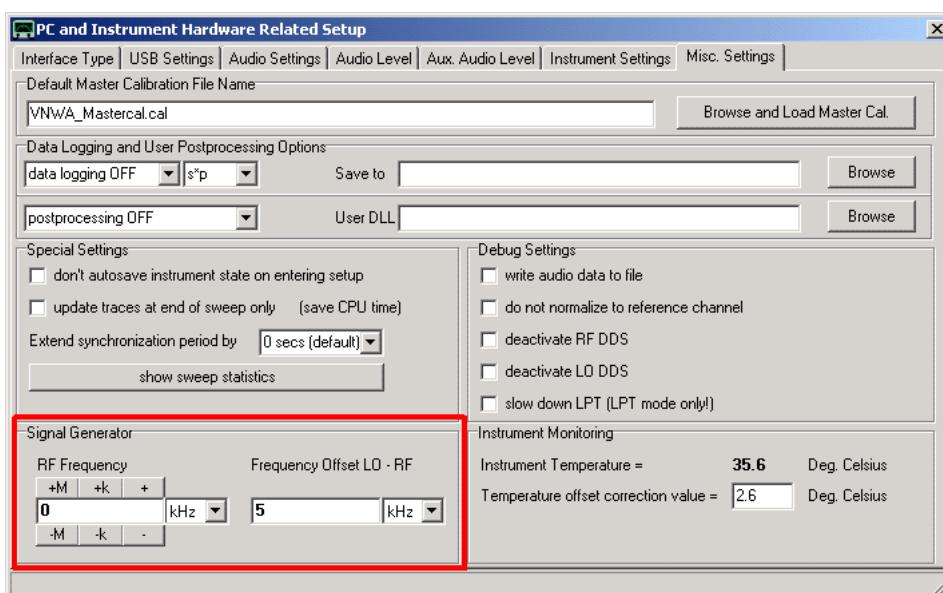
8. Test the LO signal source

Make sure that the VNWA RX port is still connected to the VNWA TX port.

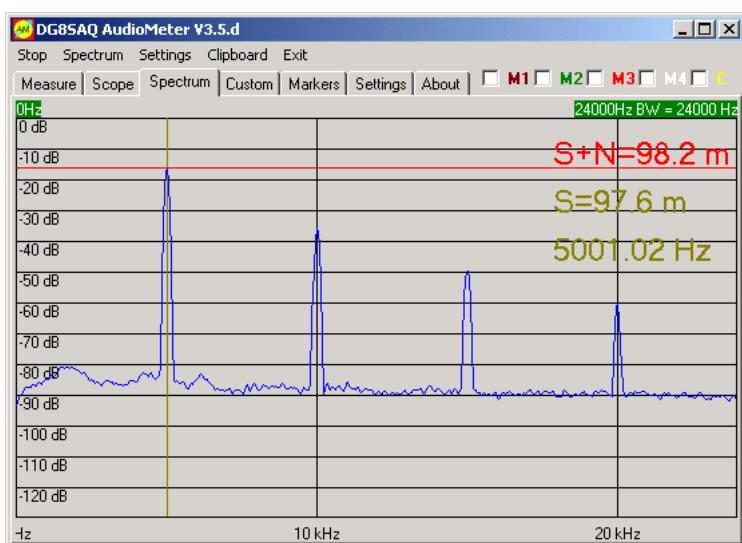
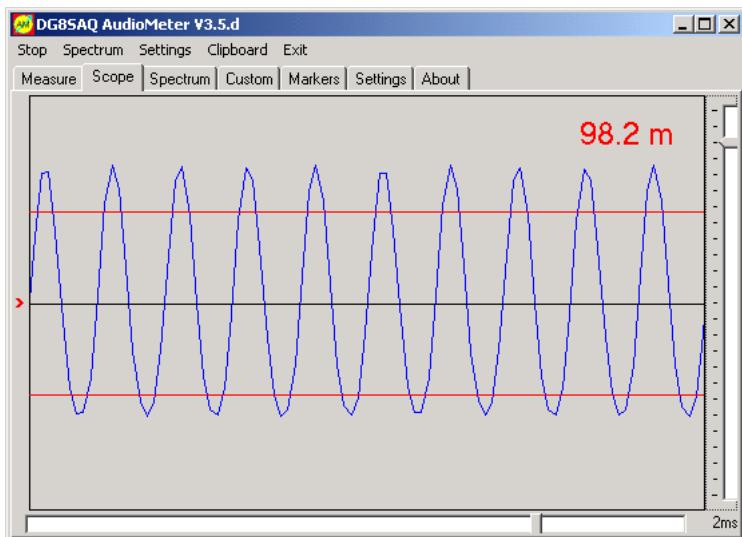
Return to normal VNWA operation mode by closing the signal generator (right-click onto the signal generator panel). Next, open the level control window by pressing the "L" key or via main menu "Settings-Power Levels-Set Levels":



Make sure, both sliders are on maximum. Return to the VNWA main window and open "Options-Setup-Misc. Settings". There, set **RF Frequency** to zero and **Frequency Offset** to 5kHz:



Again, you should see a fairly strong signal at 5kHz in AudioMeter:



If you have a VNWA3E, repeat this test with the second VNWA sound device.

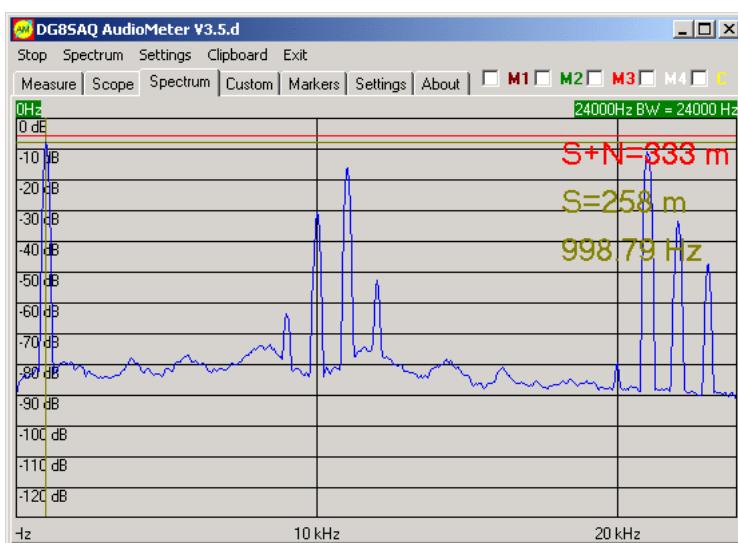
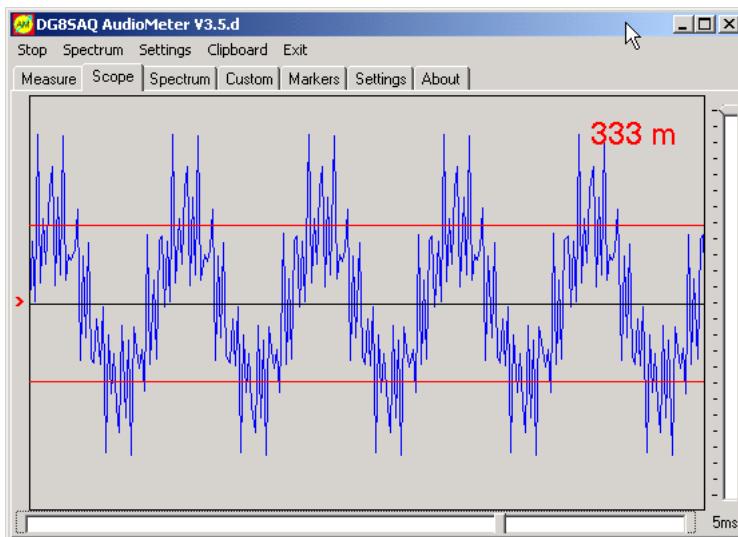
If RF and LO signal source tests have passed, you may...

9. Test the mixers

Make sure that the VNWA RX port is still connected to the VNWA TX port.

In Setup / Misc. Settings set **RF Frequency** to 10kHz and **Frequency Offset** to 1kHz:

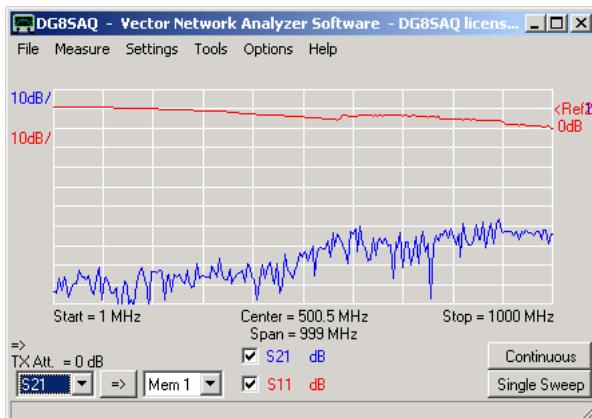
In AudioMeter you should see a strong 1kHz and 21kHz signal superposition:



10. Checking the S11/S21 switch performance

For this test you must DISCONNECT both TX and RX ports. Also, your VNWA must be properly set up. If you have a VNWA3E, you must disable the auxillary audio codec for this test.

Measure the uncalibrated S21 response of the open VNWA ports. The S21 response should be well below -60dB and rising with increasing frequency:



If your open S21 response shows a considerably higher level and is DECREASING with increasing frequency like in below example, the S11/S21 switch is likely broken.



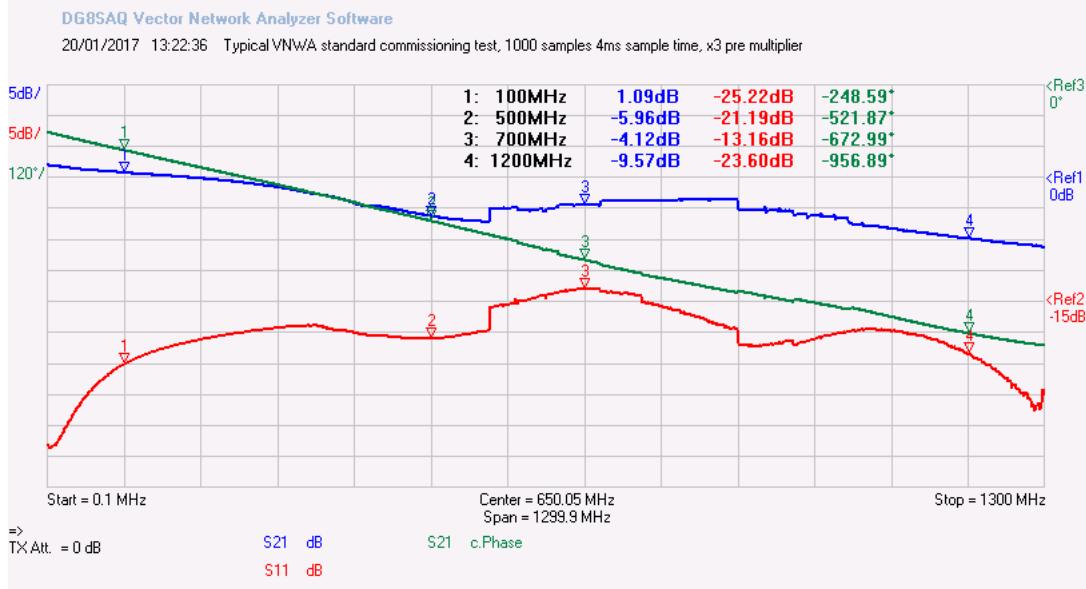
VNWA FINAL COMMISSIONING TEST

- The VNWA does not contain user adjustable components or adjustable firmware settings.
- Every assembled VNWA is supplied with a standard "**Commissioning Graph**" which shows the S11 and S21 performance when the VNWA was commissioned and tested prior to dispatch of the VNWA to the customer.
- Whenever it is deemed necessary, VNWA performance may be checked and compared against the original Commissioning Graph by using the following procedure:

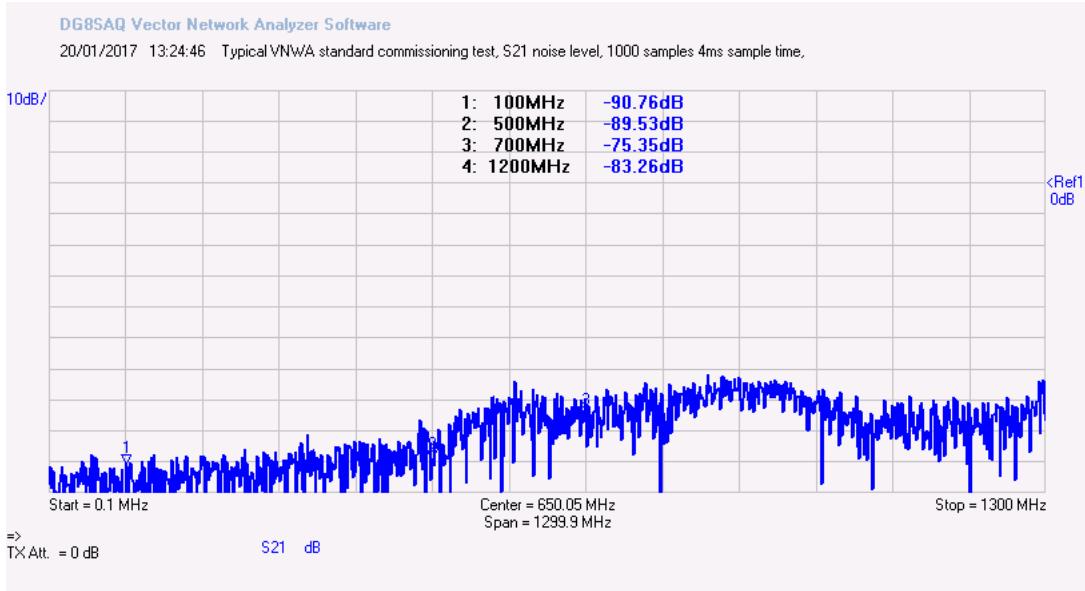
VNWA Final Commissioning Test Procedure

- Start VNWA application as normal.
- Note, that calibration must be **disabled** as it would mask hardware problems
- Set Frequency Range to cover 0.1MHz to 1300MHz and set sweep to 1000 points and a 4ms sample time (4 seconds total sweep time).
- Set Trace 1 to S21 to measure dB, Trace 2 to S11 to measure dB and Trace 3 to S21 to measure C.phase.
- Ensure that Trace Scales are set to same Unit/Div as were used on the Commissioning Graph.
- Connect TX port to RX port using the 20cm RG223 cable supplied by SDR-Kits (supplied with the VNWA in all shipments since 2015).
- Press "Continuous Sweep" and compare graph produced with the Commissioning Graph for your VNWA. Both Graphs should be identical - If there are differences, change the clock multipliers from "auto" and "3" to "auto" and "auto" and check whether the scales for the S11 and S21 traces are correct. Now run the test again.

Example of Commissioning Graph:



- Next remove the RG223 cable between TX and RX Port, change S21 dB Scale to 10dB/Div and untick the S11 dB and S21 c.phase traces.
- Again ensure that "Calibration" and "Smoothing" must be disabled as calibration could mask hardware problems.
- Now, run the test again by running a sweep. Typically the noisefloor shown in the graph below should be shown.



If the results of your VNWA differ from the Commissioning Graphs supplied then a hardware issue is suspected.
As a quick check: Disconnect power from the VNWA and use a multimeter to measure the resistance (Ohm) at the SMA TX port between center pin and GND.
For a good VNWA this should measure between 60 and 62 Ohms.
Similarly, measuring the SMA RX port, you should see around 50 Ohms on the multimeter. If these measurements yield different values, then the VNWA TX bridge and/or the RX attenuator are likely to have been damaged by RF or dc overload.

In case of suspect hardware problems please contact Support@SDR-Kits.net for advice. SDR-Kits will respond and issue an RMA number and return instructions for your VNWA.

HOW TO REDUCE CPU LOAD

If your CPU load is 100% during the sweep, the collected data will be corrupt!

Here are a few things, you can do, if your PC is too slow:

- 1) Reduce the audio sampling rate
- 2) Increase the audio buffer size
- 3) Display as few curves during the measurement as possible
- 4) Use the smallest number of data points possible
- 5) Switch off progress bar and progress display
- 6) Set display background color to "none"
- 7) Use smallest possible line widths for traces and grid lines

As the VNWA itself does not own any intelligence, all data capturing and analysis as well as the graphics have to be handled by the PC at the same time! The added intelligence in the DG8SAQ USB_VNWA interface relaxes the host PC CPU load considerably.

GENERATING DEBUG OUTPUT

The VNWA software lets you **generate a debug file** which gives information about your Windows version, your VNWA hardware and its status.

> The simplest way to report a software problem is to let the VNWA software automatically compile a debug info email. This can be done via the "Help/About-System Info-Tab".

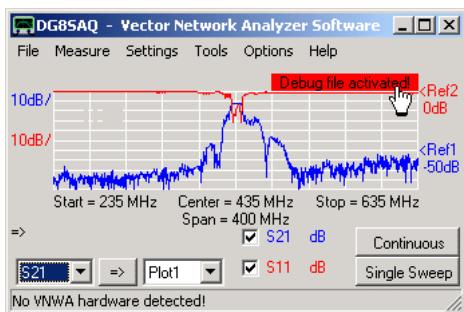
> The **debug output is activated manually** by simply creating an empty textfile with name **Debug.txt** in the VNWA program directory while the VNWA software is **NOT RUNNING**.

> Alternatively, you can press **ctrl-d** while the VNWA software is running and the VNWA main window is in focus. If no debug file exists yet, it will be created and an according message will be shown:



Note, that the debug mode is not activated yet. To activate it, terminate the VNWA software as instructed in the message.

Next, start VNWA and observe the red label indicating that VNWA is running in debug mode:



Note that VNWA runs very slow in debug mode, thus debug mode should generally be avoided. Debug mode can be deactivated either by right-clicking the debug label and selecting to exit debug mode or by manually deleting the debug file.

Once, VNWA is running in debug mode, immediately terminate VNWA again and observe the now modified file **Debug.txt**:

```
DebugLevel=0
VNWA Version Beta 33.h started at      0,005 ms
App. initialized at    24,729 ms

System Information:
-----
IsWin9x       : False
IsWinNT       : True
IsServer      : False
IsTablet      : False
IsMediaCenter : False
IsWOW64       : False
Product       : osWinXP
BuildNumber   : 2600
ServicePack   : Service Pack 3
ServicePackMajor: 3
ServicePackMinor: 0
ProductType   : ptNTWorkstation
Edition       : Professional

Delphi RTL Win32XXX Globals
-----
Win32Platform     : 2
Win32MajorVersion : 5
Win32MinorVersion : 1
Win32BuildNumber  : 2600
Win32CSDVersion   : Service Pack 3

Extended Win32XXX Globals
-----
Win32HaveExInfo   : True
Win32ServicePackMajor: 3
Win32ServicePackMinor: 0
Win32SuiteMask     : 256
Win32ProductType  : 1

All Forms created at    1222.592 ms
Data initialized at    1504.185 ms
```

For **extended debug information**, edit the file Debug.txt with a text editor (e.g. notepad) and change the first line from Debuglevel=1 to **Debuglevel=5** (not necessary when entering debug mode via pressing ctrl-d). Close the file and start and stop the VNWA software again. The debug file has grown considerably now. Besides your system and software version info it contains your firmware and a log of all USB communication:

The screenshot shows a Windows command-line window titled "Lister - [D:\VNWA\VNWA3.0\debug.txt]". The window displays a large amount of debug information from the file "debug.txt".

Key sections of the output include:

- System Information:**

```
IsWin9x      : False
IsWinNT      : True
IsServer     : False
IsTablet    : False
IsMediaCenter : False
IsWOW64      : False
Product      : osWinXP
BuildNumber   : 2600
ServicePack   : Service Pack 3
ServicePackMajor : 3
ServicePackMinor : 0
ProductType   : ptNTWorkstation
Edition       : Professional
```
- Delphi RTL Win32XXX Globals**

```
Win32Platform   : 2
Win32MajorVersion : 5
Win32MinorVersion : 1
Win32BuildNumber  : 2600
Win32CSDVersion   : Service Pack 3
```
- Extended Win32XXX Globals**

```
Win32HaveExInfo   : True
Win32ServicePackMajor : 3
Win32ServicePackMinor : 0
Win32SuiteMask     : 256
Win32ProductType   : 1
Attempting to open USB Device at      537.728 ms
-----> opening USB device at          537.740 ms
Getting USB blocked info at          563.900 ms
Getting USB Flash Address at         563.906 ms
USB Device opened at                579.727 ms
Activation key correct at           624.771 ms
```
- Firmware v4.9**

```
Flash
4BC0880C264C063C062C061C060C05FC0
5EC05DC05CC05BC054C0E7C058C057C0
56C055C054C053C052C051C050C04FC0
4EC04DC0040309041A03730064007200
2D006B006900740073002E006E006500
```

This extended debug info is very valuable for troubleshooting.

T-CHECK

The T-check is a quick test invented by R&S to give an indication whether a calibrated VNA is yielding reasonable results.

See here for more details: **Rohde & Schwarz T-Check**

English:

https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1ez43/1ez43_0e.pdf

Deutsch:

https://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/application_notes/1ez43/1ez43_0d.pdf

----- SHM ----- page 194 -----