NanoVNA User Guide (Translated by google from NanoVNA User Guide | NanoVNA User Guide)

Introduction

This document is an unofficial user guide for NanoVNA. The URL is https://cho45.github.io/NanoVNA-manual/.

It is managed in the GitHub repository.

If you have any fixes, such as discrepancies with the latest firmware, please send a pull-request.

It is also available in PDF format on the GitHub Releases page.

- https://github.com/cho45/NanoVNA-manual/releases

What is NanoVNA

There are several types of NanoVNA hardware, and this document covers the following types of hardware.

- ttrftech Edition (Original) ttrftech/NanoVNA
- hugen79 version hugen79/NanoVNA-H

These hardware components are almost identical in the circuit, and common firmware is available.

Required for operation

At a minimum, the following are required:

- NanoVNA Main Unit

- SMA LOAD 50Ω
- SMA SHORT
- SMA OPEN
- SMA Female to Female Through Connector
- SMA Male to Male Cable × 2

The Basics of NanoVNA

VNA (Vector Network Analyzer) measures the frequency characteristics of reflected and passing power in high-frequency networks (RF networks).

NanoVNA measures the following factors:

- I/Q signal of input voltage
- I/Q signal of reflected voltage
- I/Q signal of the passing voltage

From this, we calculate the following.

- Reflection coefficients S11
- Transmission coefficient S21

You can display the following items that can be calculated from these.

- reflection loss
- pass loss
- complex impedance
    - Resistance
    - reactance
- SWR

and more.

Oscillation frequency of NanoVNA

NanoVNA measures reflectance and transmission coefficients for 101 points in the target frequency band.

NanoVNA's station frequency ranges from 50kHz to 300MHz. Above this frequency, harmonic mode is used. Even in harmonic mode, the fundamental waves are not attenuated. The usage modes for each frequency are as follows.

- Up to 300MHz: Fundamental Wave
- From 300MHz to 900MHz: Third harmonic
- From 900MHz to 1500MHz: 5th harmonic

Especially when checking the amplifier's gain, it is important to note that the fundamental wave input is always present.

In all cases, the input is converted to an intermediate frequency of 5kHz. Signals are converted to analog-digital using 48kHz sampling. Digital data is processed by the MCU.

First thing to do

Before use, you must calibrate it first. At first, calibration is performed as follows.

- Make sure START is 50kHz
- Make sure STOP is 900MHz
- Perform calibration according to the [calibration method]

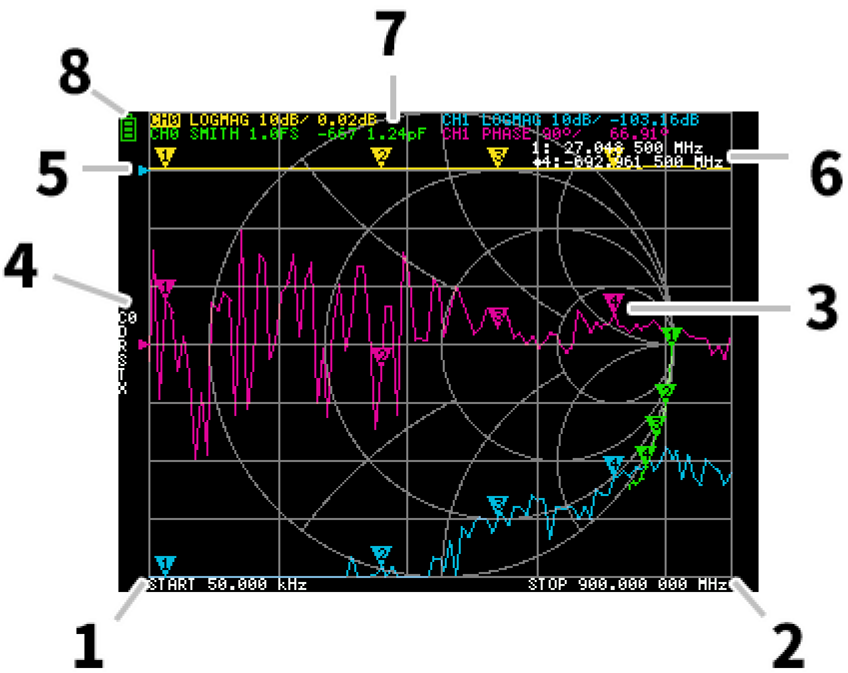
How to enter

NanoVNA has the following inputs:

- Tap and long tap on the touch panel
- Lever switch
    - Long press L / L
    - Press and hold R / R
    - Press / Press and hold
- Power slide switch

How to view the screen

Main Screen



1. START 2. STOP Frequency

Each frequency when you specify start or stop is displayed.

3. Markers

The position of each marker is displayed. Selected markers can be moved in the following ways:

- Dragging markers on the touch panel
- Press and hold the lever switch for L R

4. Calibration Condition

The data number of the calibration being read and the error correction applied will be displayed.

- C0 C1 C2 C3 C4: Each indicates that the calibration data for the corresponding number has been loaded.
- c0 c1 c2 c3 c4: Each indicates that the calibration data for the corresponding number has been loaded, but after loading, the frequency range has changed and error correction has been used.
- D: Indicates that directivity error correction is applied
- R: Indicates that rerection tracking error correction is applied
- S: Source Match indicates that error correction is applied
- T: Indicates that transmission tracking error correction is applied
- X: Isolation (crosstalk) Indicates that error correction is applied

5. Reference Position

Indicates the reference position for the corresponding trace. You can change the position there.DISPLAYSCALEREFERENCE POSITION

6. Marker Status

The active marker you are selecting and one marker that was previously active will be displayed.

7. Tracing Status

The state of each trace format and the values corresponding to active markers are displayed.

For example, in the case of the display , it is read as follows.CH0 LOGMAG 10dB/ 0.02dB

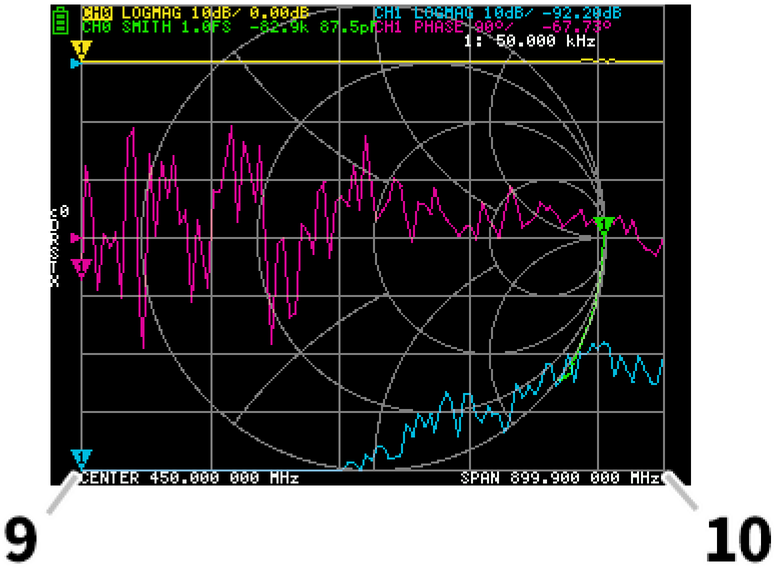
- Channel CH0 (Reflection)
- Format LOGMAG
- The scale is 10dB.
- Current value is 0.02dB

Also, active traces invert the channel display.

8. Battery Status

If the battery is installed and the PCB is already installed, an icon will appear according to the battery voltage.D2

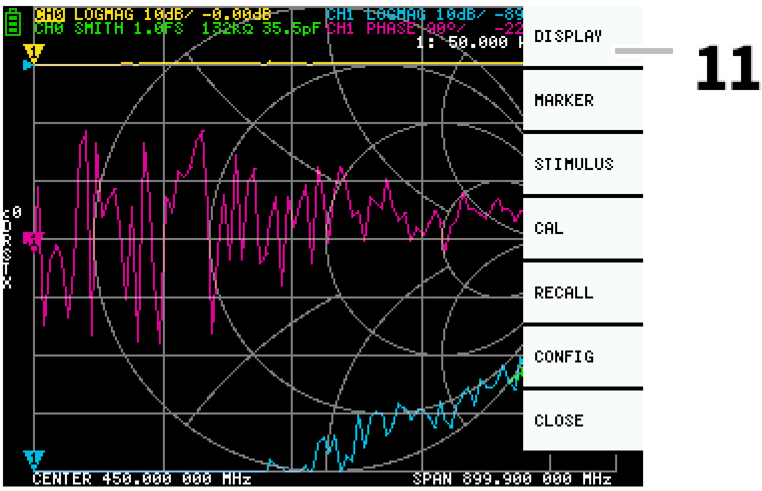
Main Screen 2



9. CENTER Frequency 10. span

The frequencies for each specified center frequency and span are displayed.

Menu screen



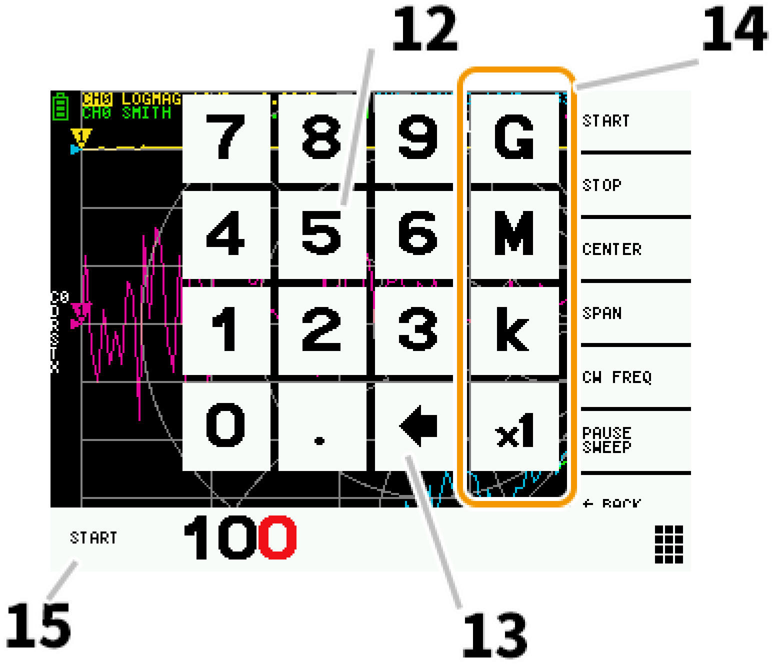
11. Menu

You can display the menu using the following steps.

- When tapping anywhere other than the touch panel marker
- Push in the lever switch

---

keypad screen



12. Number keys

Tap a number to enter one character.

13. Backkey

1 character removed. 1 If you have not entered any characters, cancel the input and restore the previous state.

14. Unit Key

Multiply the unit corresponding to the current input and immediately finish the input. In ×1, the entered values will be set as they are.

15. Input field

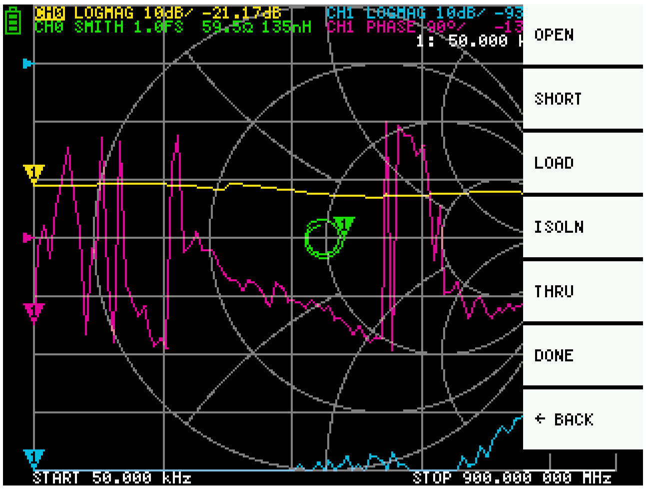
The name of the field to be entered and the number you entered will be displayed.

Start measuring

Basic measurement sequence

1. Set the frequency range to measure
2. to perform calibration
3. Connecting the DUT

Calibration method



Calibration basically needs to be performed every time the frequency range to be measured is changed. If the error has been corrected correctly, the calibration status display on the screen will be . n is the data number being loaded.Cn D R S T X

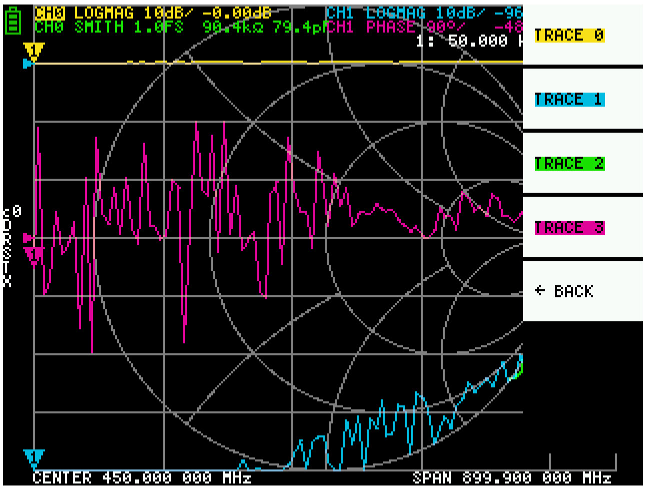
However, NanoVNA can supplement existing calibration information to provide a fairly accurate display. If you change the frequency range after loading calibration data, this state will occur. At this time, the calibration status displayed on the screen will be . n is the data number being loaded.cn D R S T X

1. Resets the current calibration state CAL RESET
2. Connect the OPEN Standard to the CH0 port and run the OPEN standard.CALCALIBRATEOPEN
3. Connect the SHORT standard to the CH0 port and run .CALCALIBRATESHORT
4. Connect the LOAD standard to the CH0 port and run .CALCALIBRATELOAD
5. Connect the LOAD standard to ports CH0 and CH1 and execute . If there is only one load, the CH0 port may remain unconnected.CALCALIBRATEISOLN
6. Connect cables to ports CH0 and CH1, connect cables with through connectors, and run the program.CALCALIBRATETHRU
7. Calibration is complete, and error correction information is calculated CAL CALIBRATE DONE
8. Specify the data number before saving.CAL CALIBRATE SAVE SAVE 0

※ Each calibration data must be imported only after the display is sufficiently stable.

Features

Trace display



You can display up to four traces, one of which is an active trace.

Traces can display only what is necessary. To switch the display, select .DISPLAYTRACETRACE n

Here are some ways to switch active traces.

- Tap the trace marker you want to activate
- DISPLAY TRACE TRACE n Select and display it. (If it is already shown, it must be temporarily hidden.)

Trace Format

Each trace can be formatted individually. To change the format of an active trace, select the format you want to change.DISPLAYFORMAT

The display of each format is as follows.

- LOGMAG: Logarithm of the absolute value of the measurement
- PHASE: Phase from -180° to +180°
- DELAY: Delay
- SMITH: Smith Chart
- SWR: Standing Wave Ratio
- POLAR: Polar coordinate format
- LINEAR: Absolute value of the measurement
- REAL: The actual number of measurements
- IMAG: Imaginary value of measurement
- RESISTANCE: The resistance component of the measured impedance
- REACTANCE: Among the measured impedances, the reactance component

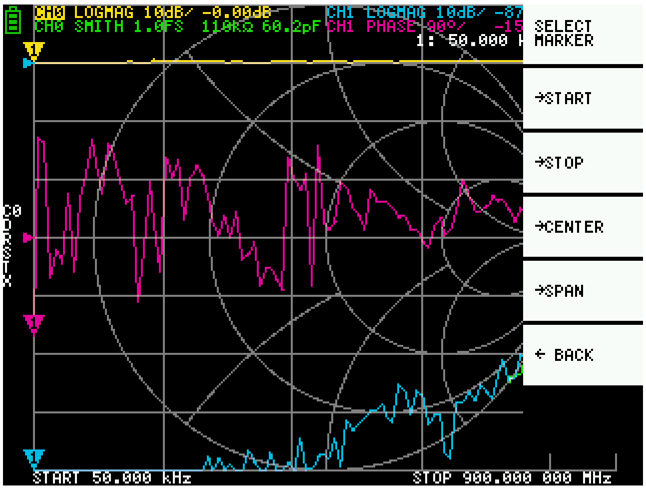
Trace Channel

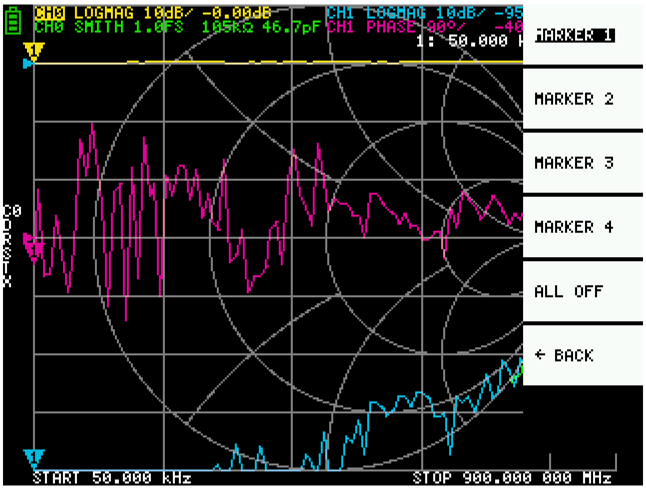
NanoVNA has two ports. The following S-parameters can be measured at each port.CH0CH1

- CH0 S11 (Reflection Loss)
- CH1 S21 (Insertion Loss)

To change the tracing channel, select either or .DISPLAYCHANNELCH0 REFLECTCH1 THROUGH

marker





Up to four markers can be displayed. Marker display is done from . When a marker is displayed, the active marker is set to the one displayed.MARKERSELECT MARKERMARKER n

Time domain operations

NanoVNA can simulate temporal domain measurements by processing frequency domain data signals.

Select Convert measurement data to a time domain. If enabled, measurement data is immediately converted and displayed in the time domain.DISPLAYTRANSOFRMTRANSFORM ONTRANSFORM ON

The relationship between the time domain and the frequency domain is as follows.

- The higher the maximum frequency, the higher the temporal resolution becomes.
- The narrower the interval between measurement frequencies (i.e., the lower the maximum frequency), the longer the maximum duration increases.

From this, maximum time length and time resolution are in a trade-off relationship.

If we rephrase long time as distance, the following can be explained.

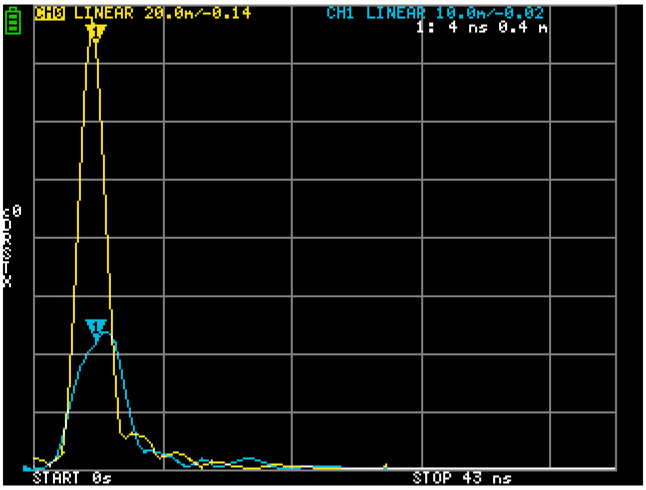
- If you want to increase the maximum measurement distance, you need to lower the maximum frequency
- To accurately determine distances, the maximum frequency needs to be increased.

Time domain bandpath

In bandpass mode, you can simulate the DUT's response to impulse signals.

The trace format can be set to .LINEARLOGMAGSWR

Below is an example of an impulse response for a bandpass filter.

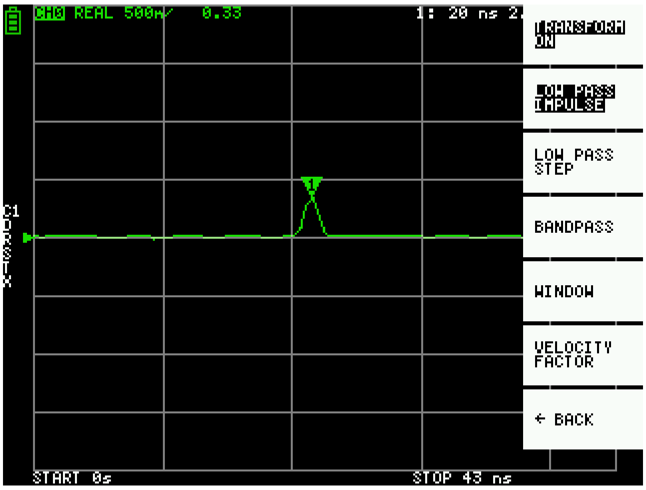


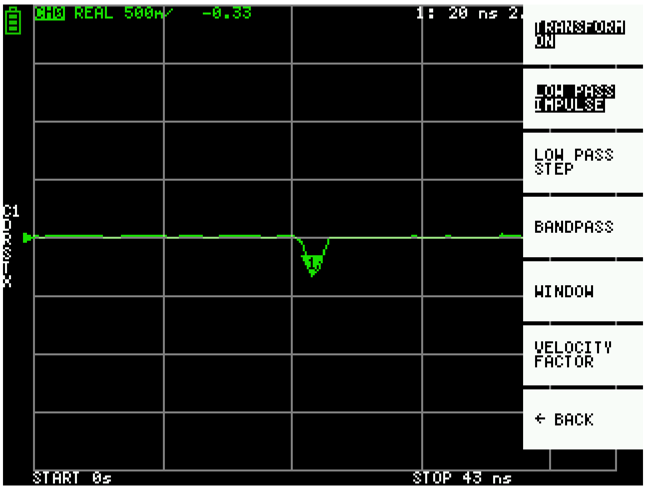
Time domain low-pass impulse

In low-pass mode, TDR can be simulated. In low-pass mode, the start frequency should be set to 50kHz, and the stop frequency should be set according to the distance you want to measure.

You can set the trace format to .REAL

Below are examples of open-state step responses and short-state impulse responses.



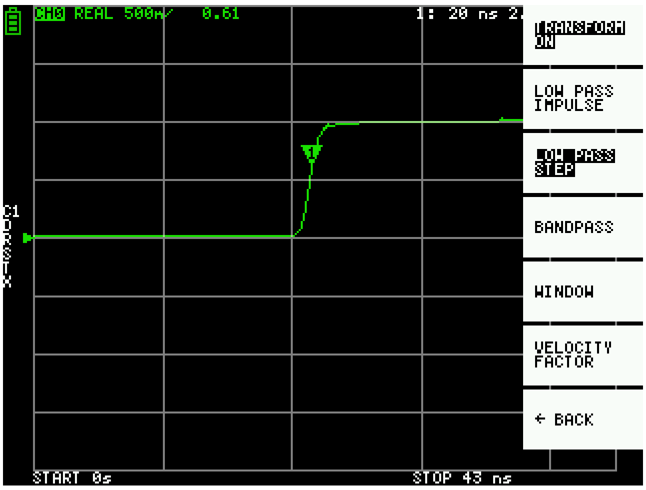


Time domain pathway steps

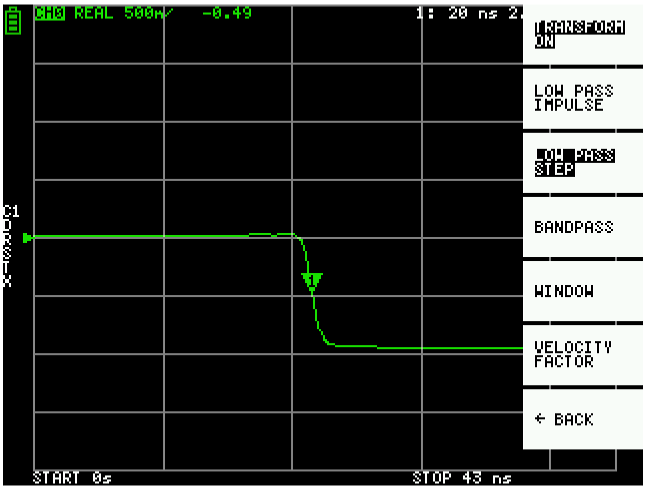
In low-pass mode, TDR can be simulated. In low-pass mode, the start frequency should be set to 50kHz, and the stop frequency should be set according to the distance you want to measure.

You can set the trace format to .REAL

Open:

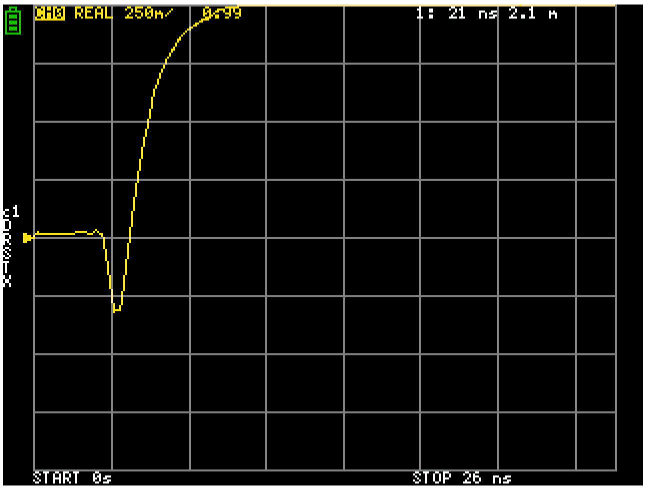


Short:

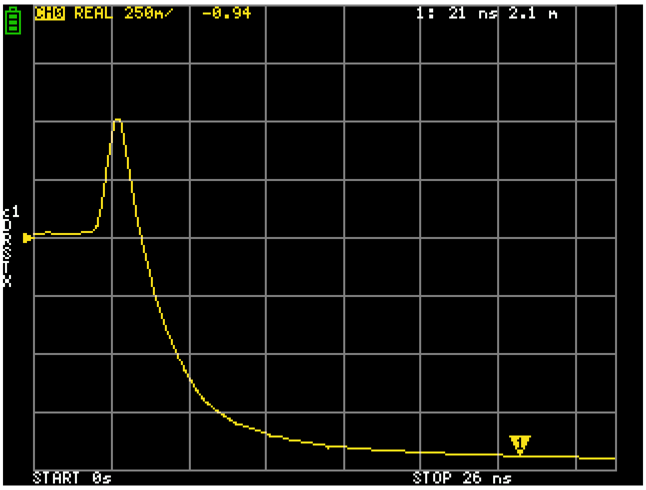


Example of a Step Response

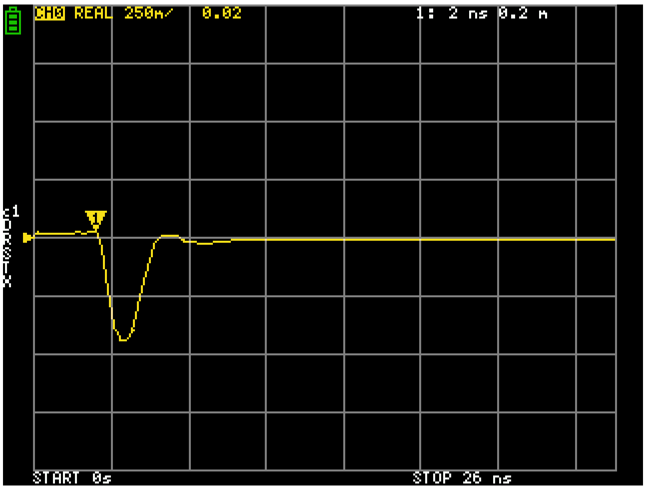
Capacitive short:



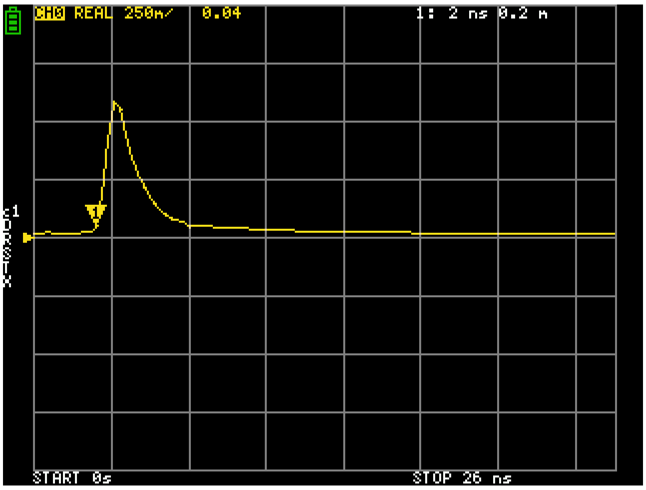
Inductive Short:



Capacitive discontinuity (parallel C):



Inductive discontinuity (L in series):



Time Domain Window

The range that can be measured is finite, with both minimum and maximum frequencies. The window can be used to smooth out this discontinuous measurement data and reduce ringing.

The window has three levels.

- MINIMUM (No window, i.e., same as rectangular window)
- NORMAL (equivalent to β=6 in the Kaiser window)
- MAXIMUM (equivalent to Kaiser window β=13)

MINIMUM results in maximum resolution, while MAXIMUM results in a maximum dynamic range. NORMAL is somewhere in between.

Setting the Velocity Factor in the Time Domain

The transmission speed of electromagnetic waves in cables varies depending on the material. The ratio of electromagnetic wave propagation speed in a vacuum is called the velocity factor (Velocity of propagation). This is always specified in the cable specifications.

In the time domain, you can display the displayed time converted into distance. The wavelength reduction rate used for distance display can be set to . For example, if you are measuring the

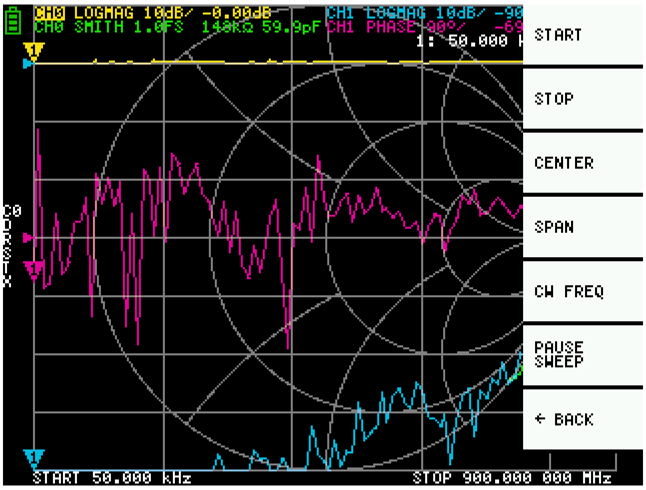
TDR of a cable with a wavelength reduction rate of 67%, specify .DISPLAYTRANSFORMVELOCITY FACTORVELOCITY FACTOR67

Set the frequency from the marker

You can set the frequency range from the marker as shown below.

- MARKER →START Set the frequency of the active marker to the start frequency.
- MARKER →STOP Set the frequency of the active marker to the stop frequency.
- MARKER →CENTER Set the frequency of the active marker to the center frequency. The span is adjusted to maintain the current range as much as possible.
- MARKER →SPAN Set the two displayed markers to span, including active markers. If only one marker is displayed, nothing happens.

Setting the measurement range



There are three types of measurement range settings.

- Set start and stop frequencies
- Set the center frequency and span
- Zero span

Set start and stop frequencies

Select and configure each of the options.STIMULUSSTARTSTIMULUSSTOP

Set the center frequency and span

Select and configure each of the options.STIMULUSCENTERSTIMULUSSPAN

Zero span

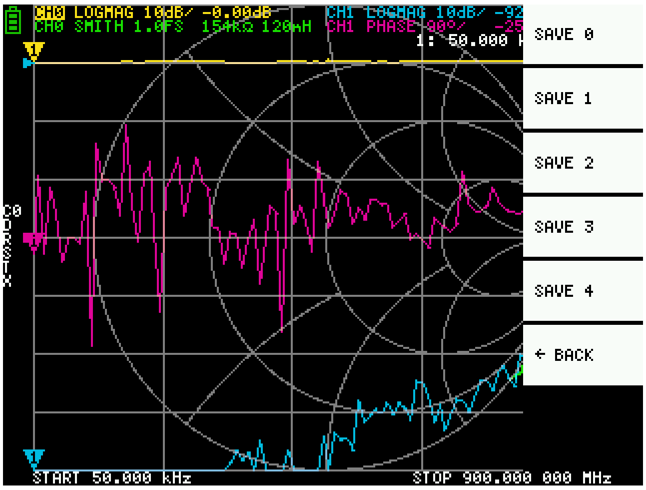
Zero span is a mode that transmits a single frequency consecutively without frequency sweeping.

STIMULUS CW FREQ Select and set it.

Temporarily stop measurement

STIMULUS PAUSE SWEEP When selected, measurement will be temporarily stopped.

Calibrating and Setting Calls



Up to 5 calibration data points can be stored. Immediately after launching, NanoVNA loads data number 0.

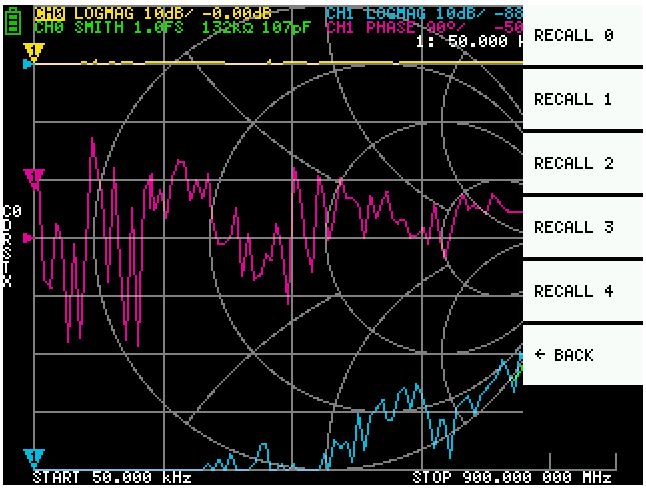
Calibration data refers to data that includes the following information.

- Frequency setting range
- Error correction at each measurement point
- Trace Settings
- Marker Settings Status
- Domain mode settings
- Setting the wavelength reduction rate
- electrical delay

CAL SAVE SAVE n You can save your current settings by selecting it.

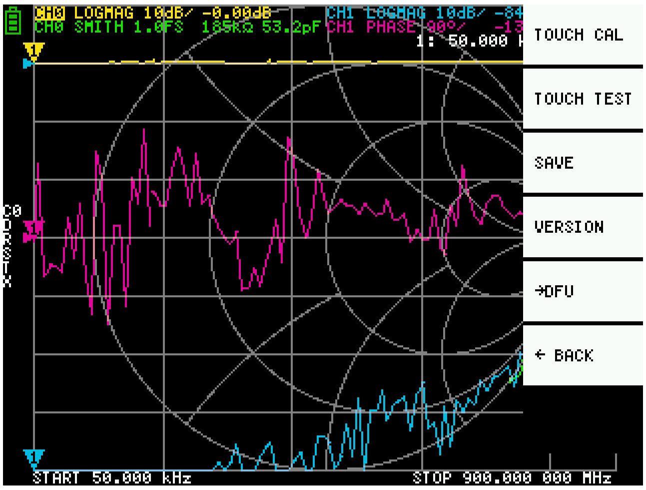
CAL RESET You can reset the current calibration data by selecting it. If you want to recalibrate, you need to do so.RESET

CAL CORRECTION indicates whether the error has been fixed. You can select this to temporarily stop error fixing.



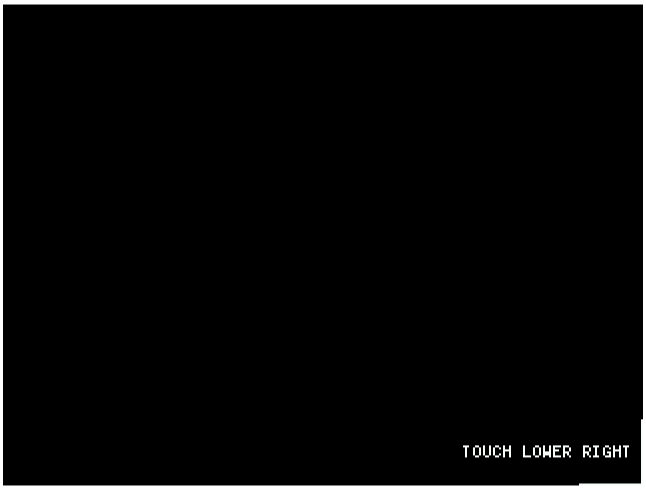
RECALL RECALL n You can retrieve the saved settings by selecting them.

Device Settings

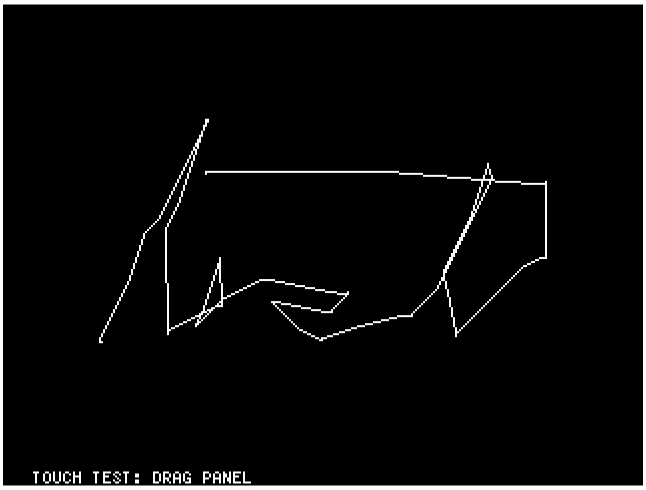


CONFIG Below, you can perform general device settings and other settings.

Calibration and testing of touch panels



CONFIG TOUCH CAL Select this to perform calibration on the touch panel. If there is a significant difference between the actual tap position and the recognized tap position, this can be resolved by doing so. After performing the settings, perform the set to confirm it is configured correctly, and save the settings with .TOUCH CALTOUCH TESTSAVE



CONFIG TOUCH TEST By selecting it, you can test the touch panel. While tapping the touch panel, lines are drawn. When you release it from the touch panel, it returns to its original state.

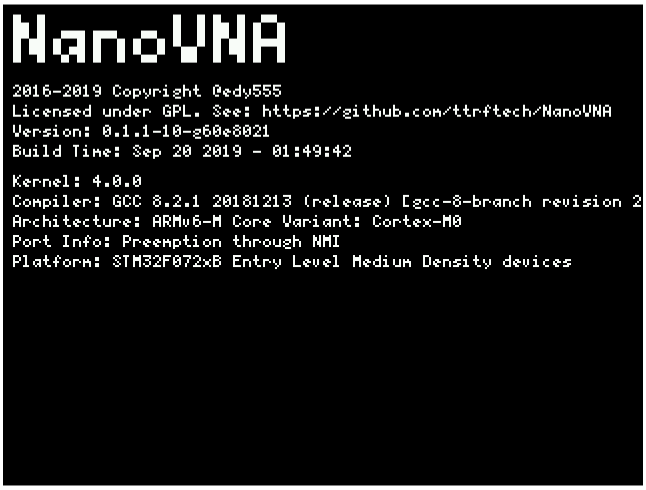
Saving device settings

CONFIG SAVE By selecting it, you can save the overall settings of your device. General device settings include data that includes the following information.

- Touch panel calibration information
- Grid Color
- trace color
- Calibration data number loaded by default

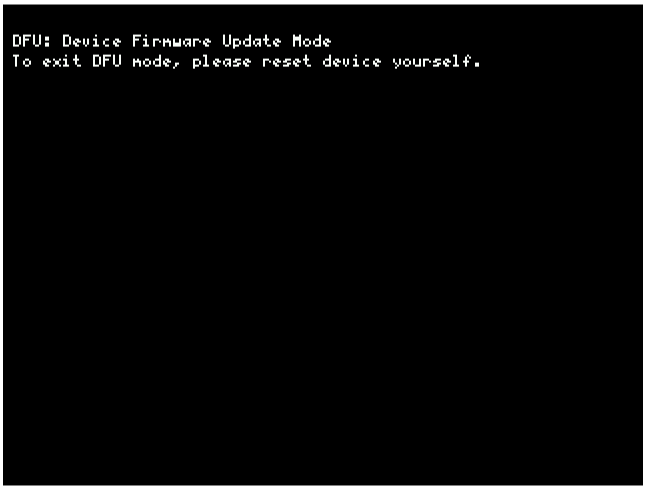
Apart from the calibration information on the touch panel, there is currently no way to set it up.

Show version



CONFIG VERSION By selecting it, you can view the device version information.

Firmware Update



CONFIG →DFU RESET AND ENTER DFU Once selected, the device will be reset and enter DFU (Device Firmware Update) mode. In this mode, firmware updates can be done via USB.

How to update firmware

How to obtain firmware

ttrftech firmware

This is the original firmware. It is version-controlled and frequently developed.

- GitHub releases
- CircleCI build

GitHub releases have some stable release firmware available.

CircleCI has all firmware for every commit. Use this if you want to try the latest features or check for bugs.

hugen79 Firmware

- Google Drive

The latest firmware is installed on Google Drive.

Build it yourself

You can easily clone a GitHub repository and build it yourself.

How to write firmware

There are various ways to write data, but here we will explain using dfu-util. dfu-util is a cross-platform tool, and Windows also provides binaries, so you can use it simply by downloading it.

Writing with dfu-util (Ubuntu)

DFU-util is available in the standard package repository.

sudo apt-get install dfu-util

dfu-util --version

Boot your device in DFU mode. Enter DFU mode by one of the following methods.

- Jump on the BOOT0 pin on the PCB while powering it on. (After powering on, remove the jumper.) The screen will turn completely white, but it is normal.
- CONFIG →DFU RESET AND ENTER DFU Select

Run the following command. build/ch.bin describes the path up to .bin of the downloaded firmware file.

dfu-util -d 0483:df11 -a 0 -s 0x08000000:leave -D build/ch.bin

Writing with dfu-util (macOS)

It is recommended to install it using homebrew.

Install the brew command.

ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

Install the dfu-util command.

brew install dfu-util

Make sure the dfu-util command starts successfully.

dfu-util --version

Boot your device in DFU mode. Enter DFU mode by one of the following methods.

- Jump on the BOOT0 pin on the PCB while powering it on. (After powering on, remove the jumper.) The screen will turn completely white, but it is normal.
- CONFIG →DFU RESET AND ENTER DFU Select

Run the following command. build/ch.bin describes the path up to .bin of the downloaded firmware file.

dfu-util -d 0483:df11 -a 0 -s 0x08000000:leave -D build/ch.bin

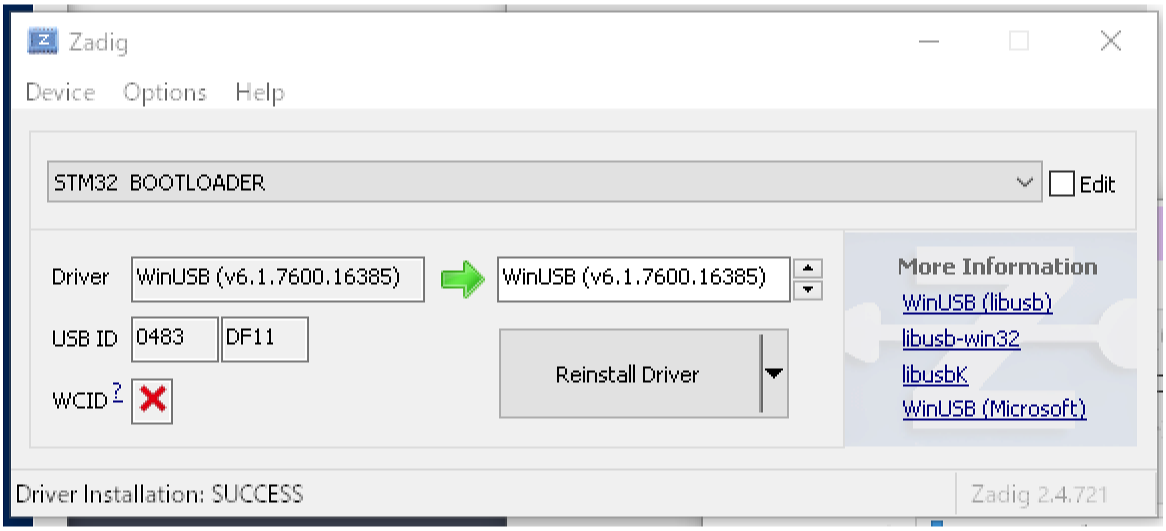
Writing with dfu-util (Windows 10)

On Windows, connecting DFU mode NanoVNA automatically installs the device driver, but this device driver does not support DFU-util. Here, you will use [Zadig]() to switch drivers.

Boot your device in DFU mode. Enter DFU mode by one of the following methods.

- Jump on the BOOT0 pin on the PCB while powering it on. (After powering on, remove the jumper.) The screen will turn completely white, but it is normal.
- CONFIG →DFU RESET AND ENTER DFU Select

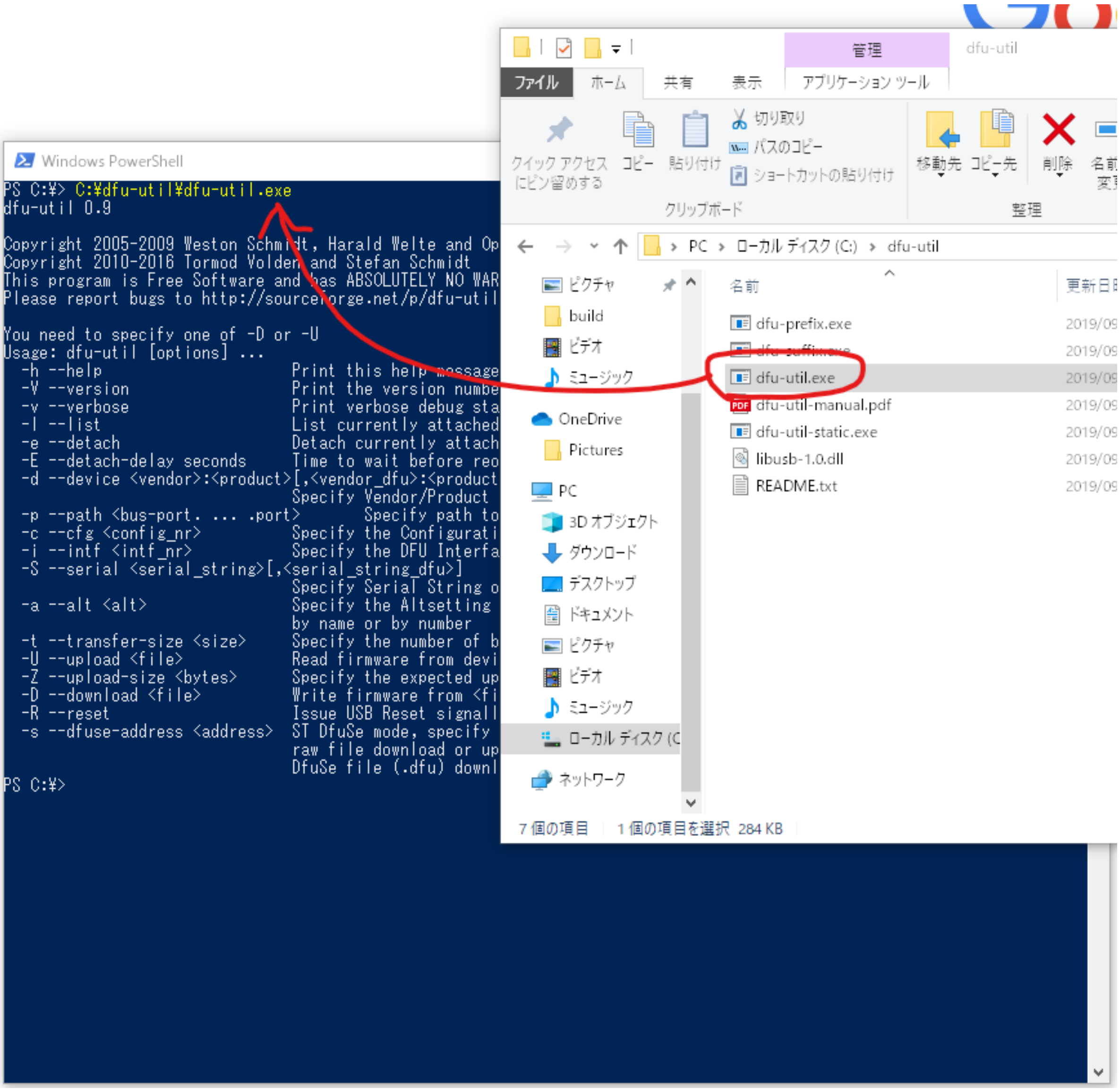
Launch Zadig with a NanoVNA connected in DFU mode, and use WinUSB as a driver for the STM32 BOOTLOADER as shown below.



* If you want to undo the driver, go to "Device Manager" under "Universal Serial Bus Controller" and search for the relevant device and execute "Uninstall Device." Unplugging the USB connector and reinserting it will automatically install the driver.

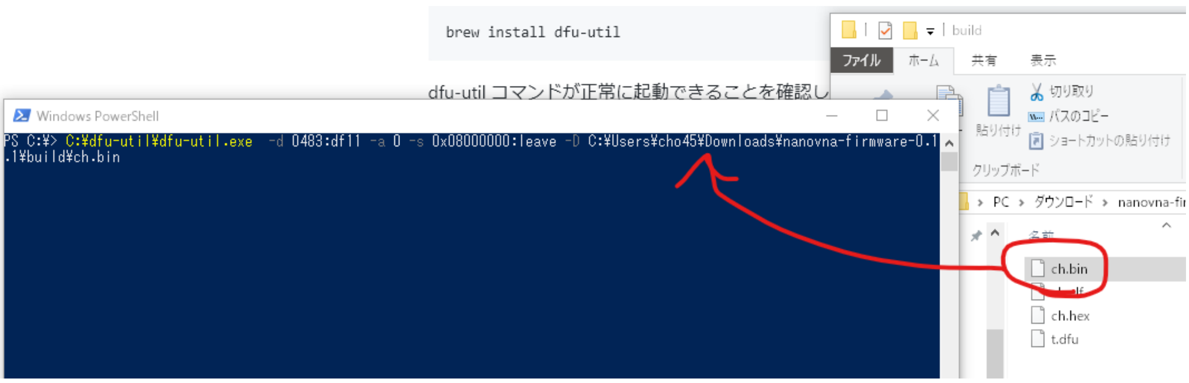
Next, deploy dfu-util. Download [and expand the dfu-util-0.9-win64.zip]() from Releases. Here, as an example, we will deploy it to C:\dfu-util (anywhere is fine).

Right-click the Start menu and select Windows PowerShell. The shell screen will open.



When you drag and drop dfu-util.exe into PowerShell from Explorer, the path will be automatically inserted. If you start it with the following set enabled, you can display the version of dfu-util.--version

C:\dfu-util\dfu-util.exe --version



Similarly, you can enter the firmware file by dragging and dropping it from Explorer into PowerShell.

Run the following command. build/ch.bin describes the path up to .bin of the downloaded firmware file.

C:\dfu-util\dfu-util.exe -d 0483:df11 -a 0 -s 0x08000000:leave -D build\ch.bin

How to Write Firmware (Windows GUI)

For those unfamiliar with CUI, some steps are a bit more complicated, but I'll also introduce a method for writing using the DfuSE Demo tool provided by ST as a reference.

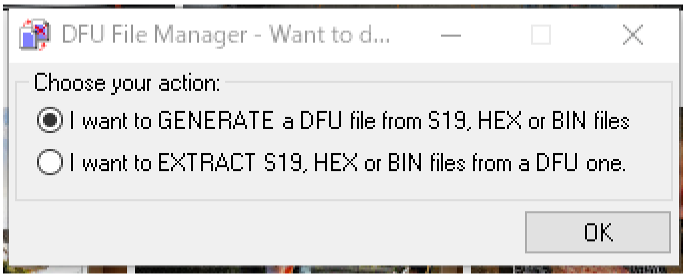
Download STSW-STM32080 from the ST website.

- DFU File Manager: A tool to create .dfu files from .bin or .hex
- DfuSe Demo: A tool to write .dfu files to your device

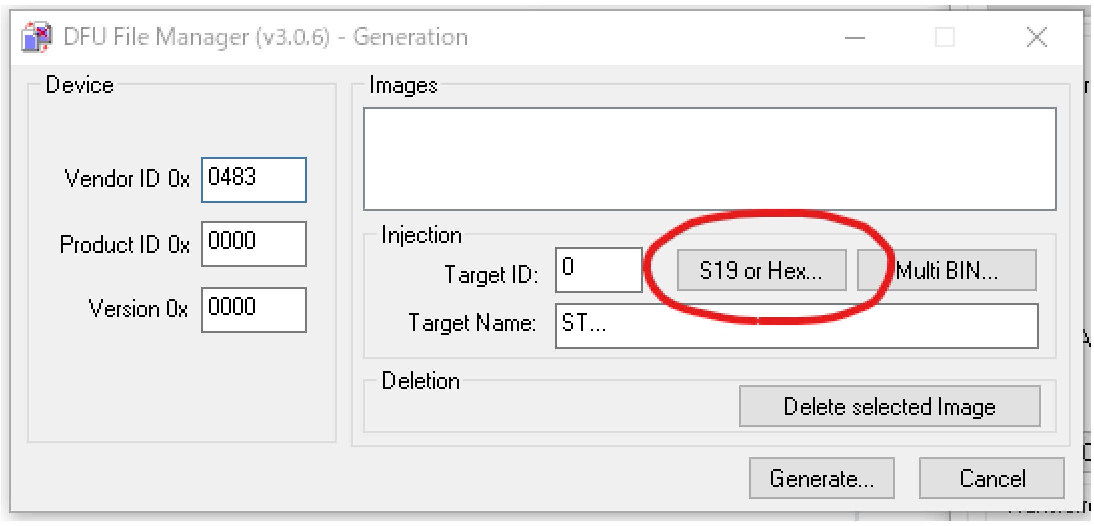
These are included.

Convert file formats using DFU File Manager.

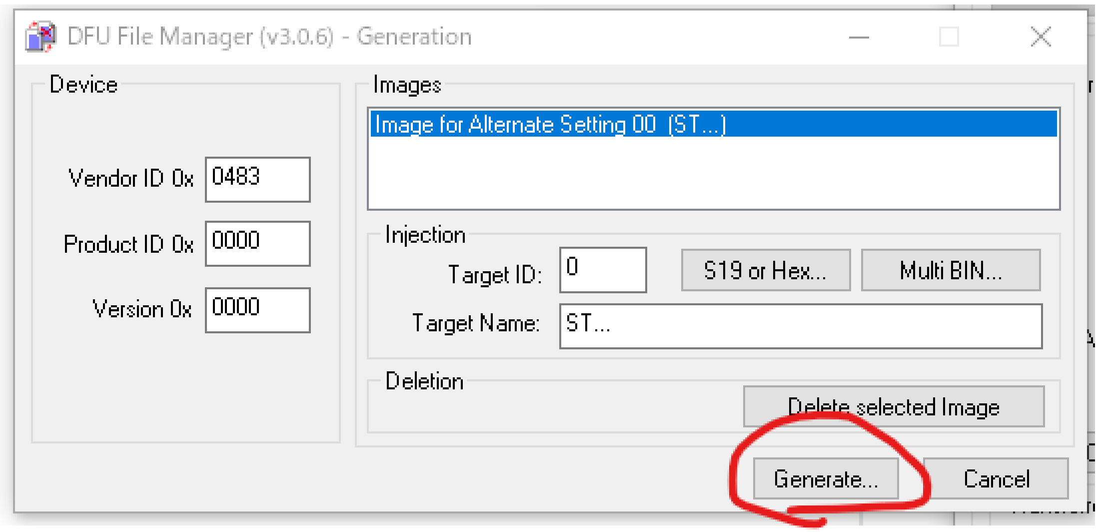
First, launch DFU File Manager.



I want to GENERATE a DFU file from S19, HEX or BIN files Select it.



S19 or Hex... Click the button. and so on.ch.hex



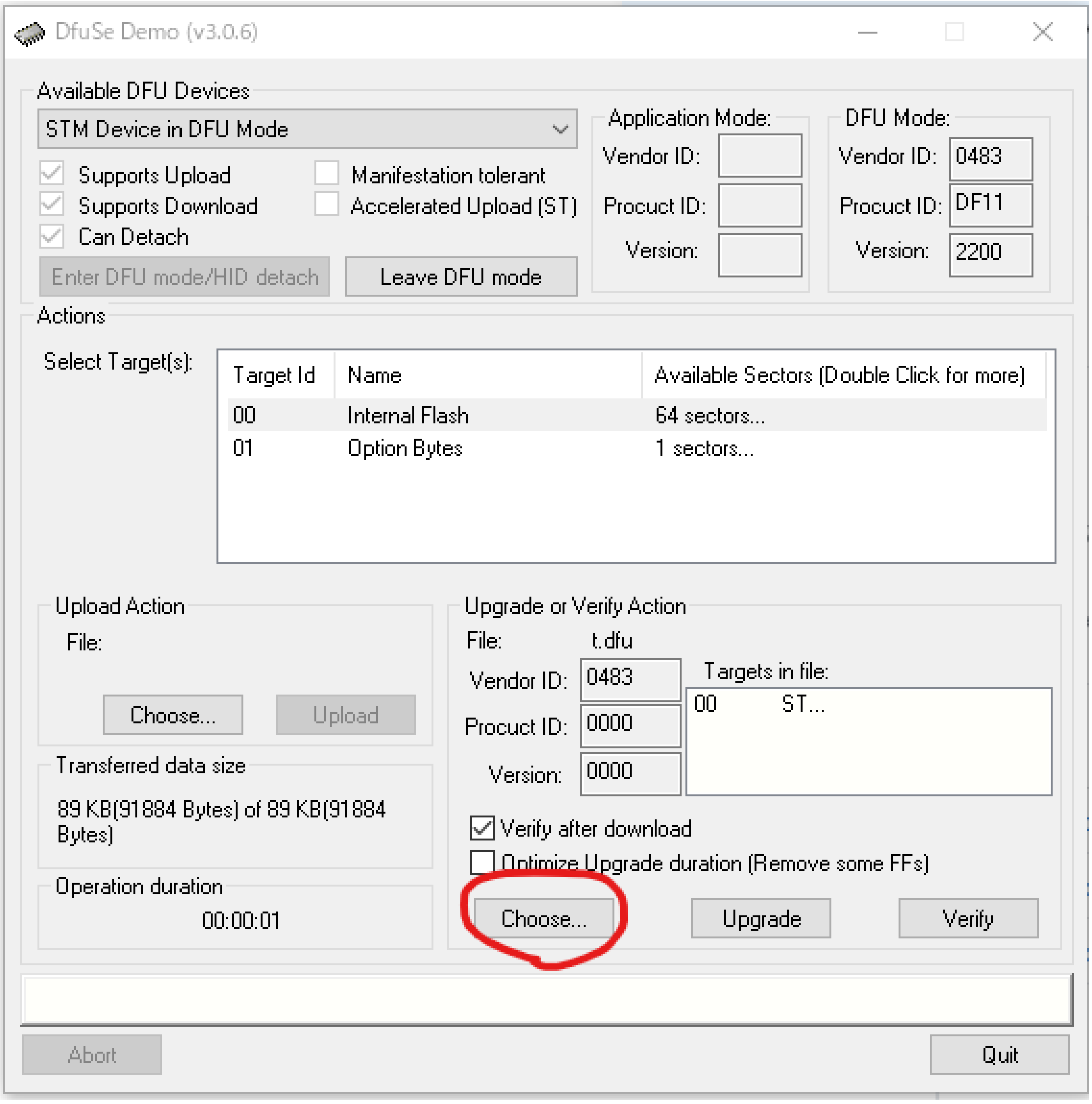
Generate... Click the button to create a .dfu file with an appropriate name.

Writing firmware with DfuSe Demo

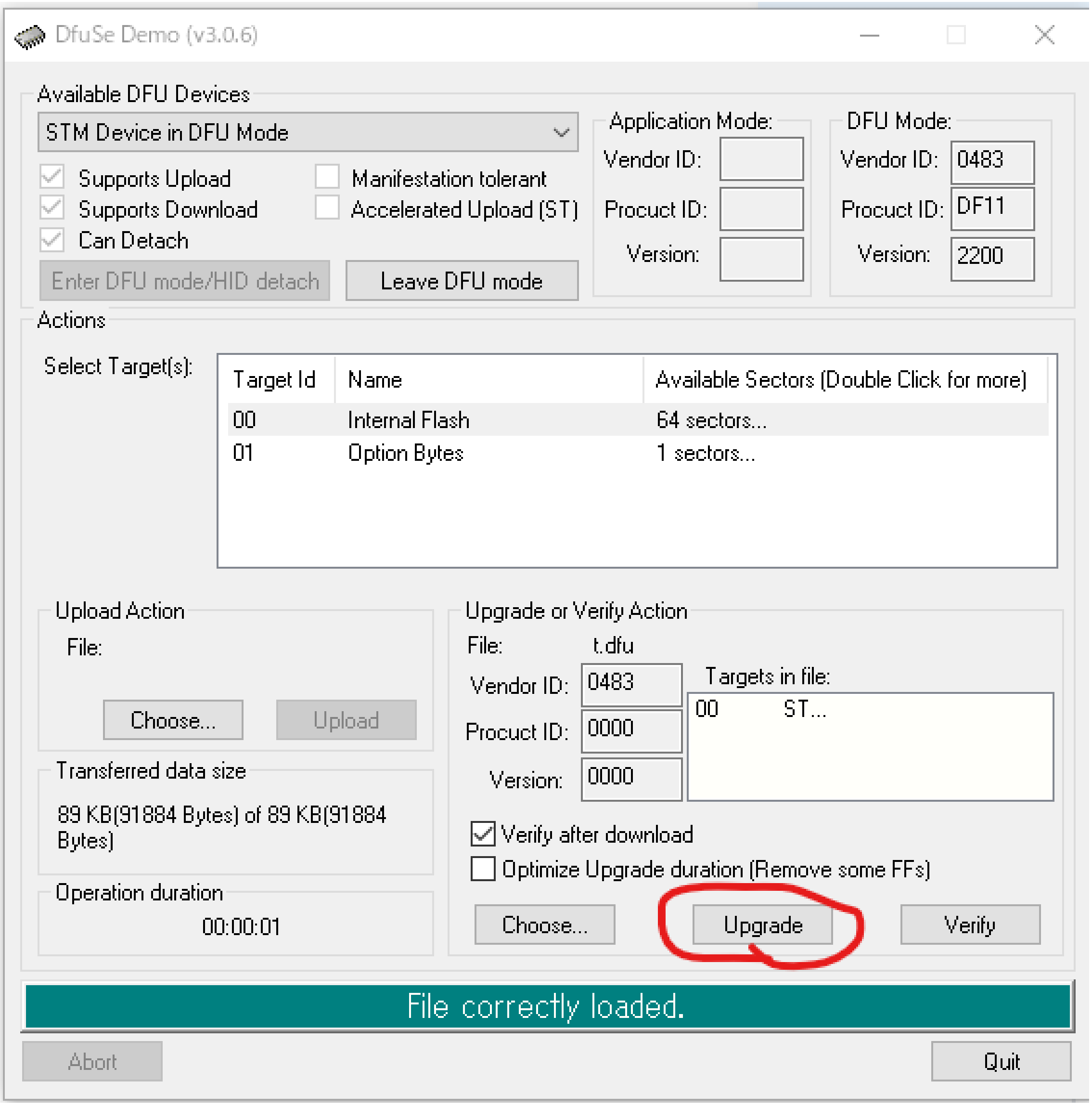
First, launch your device in DFU mode. Enter DFU mode by one of the following methods.

- Jump on the BOOT0 pin on the PCB while powering it on. (After powering on, remove the jumper.) The screen will turn completely white, but it is normal.
- CONFIG →DFU RESET AND ENTER DFU Select

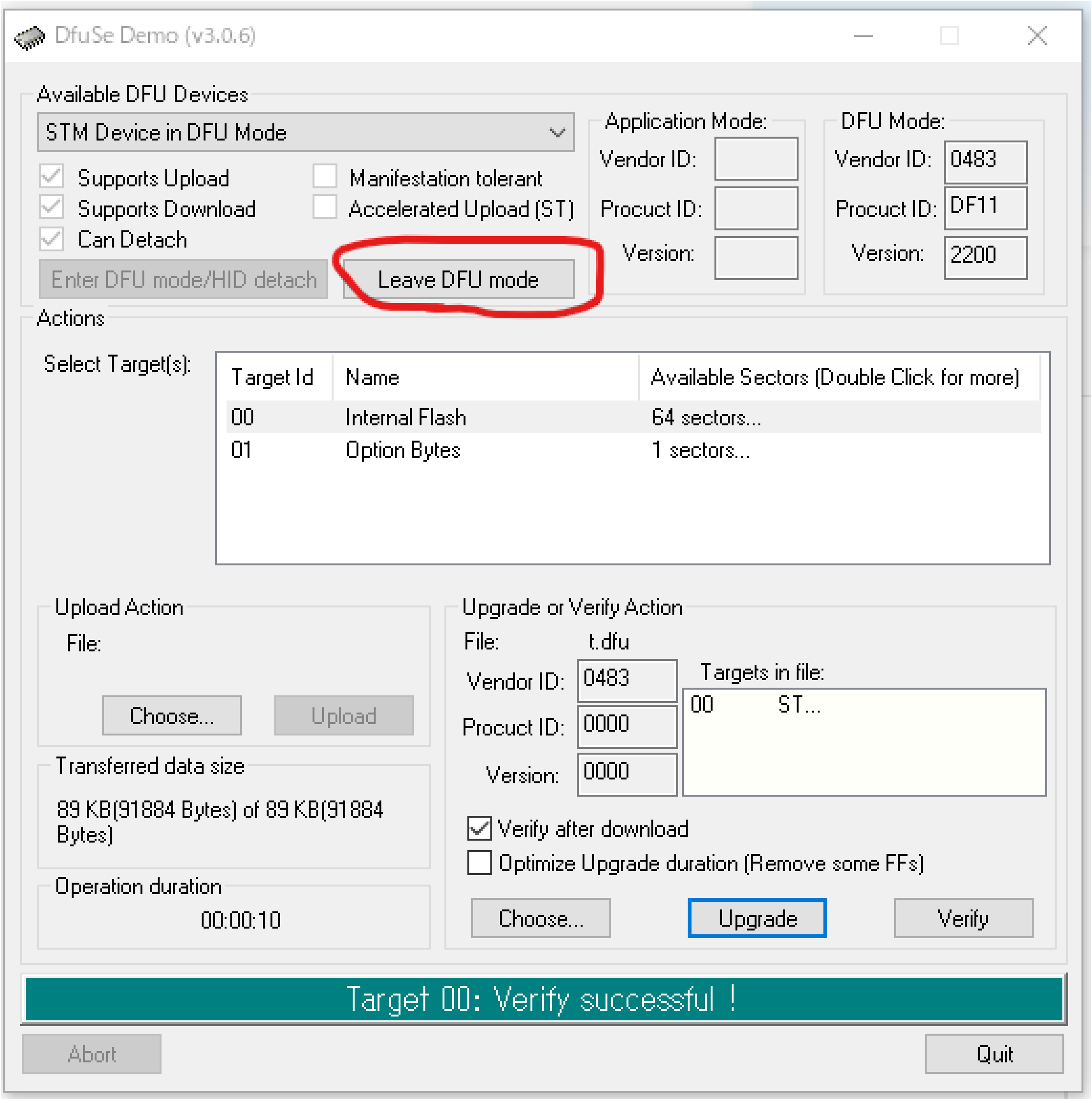
Launch DfuSe Demo. Make sure Available DFU Devices is available, then click .STM Device in DFU ModeChoose...



Select the .dfu file you just saved.



Upgrade Click the button.



Once the writing is complete, this screen will appear; click the button to exit DFU mode. The device will reset and boot with the new firmware.Leave DFU mode

Firmware Development Guide

The necessary development of NanoVNA firmware is as follows:

- Git
- gcc-arm-none-eabi
- make

If these are already available, you can build firmware using .make

git clone git@github.com:ttrftech/NanoVNA.git

cd NanoVNA

git submodule update --init --recursive

make

Building with Docker

With Docker, you can build without any hassle. Docker is a free cross-platform container utility. It can be used to quickly recreate specific environments (in this case, build environments).

After installing Docker, simply run the following command.

docker run -it --rm -v $(PWD):/work edy555/arm-embedded:8.2 make

On-chip Debugging Using Visual Studio Code

Visual Studio Code (hereafter VSCode) is a multi-platform code editor provided free of charge by Microsoft. Cortex-Debug By introducing Extensions, on-chip debugging can be performed via GUI.

While the platform-dependent part will be omitted, in addition to the above, the following are required.

- openocd
- VSCode
- Cortex-Debug

Cortex-Debug searches for and installs from VSCode's Extensions.

tasks.json

First, define a "task" to make the entire NanoVNA on VSCode.

```
{
  "tasks": [
    {
      "type": "shell",
      "label": "build",
      "command": "make",
      "args": [
      ],
      "options": {
        "cwd": "${workspaceRoot}"
      }
    }
  ],
  "version": "2.0.0"
}
```

Now you can make it as a task on VSCode.

launch.json

Next, define how to launch during debugging. Follow the instructions in Cortex-Debug to configure the settings.

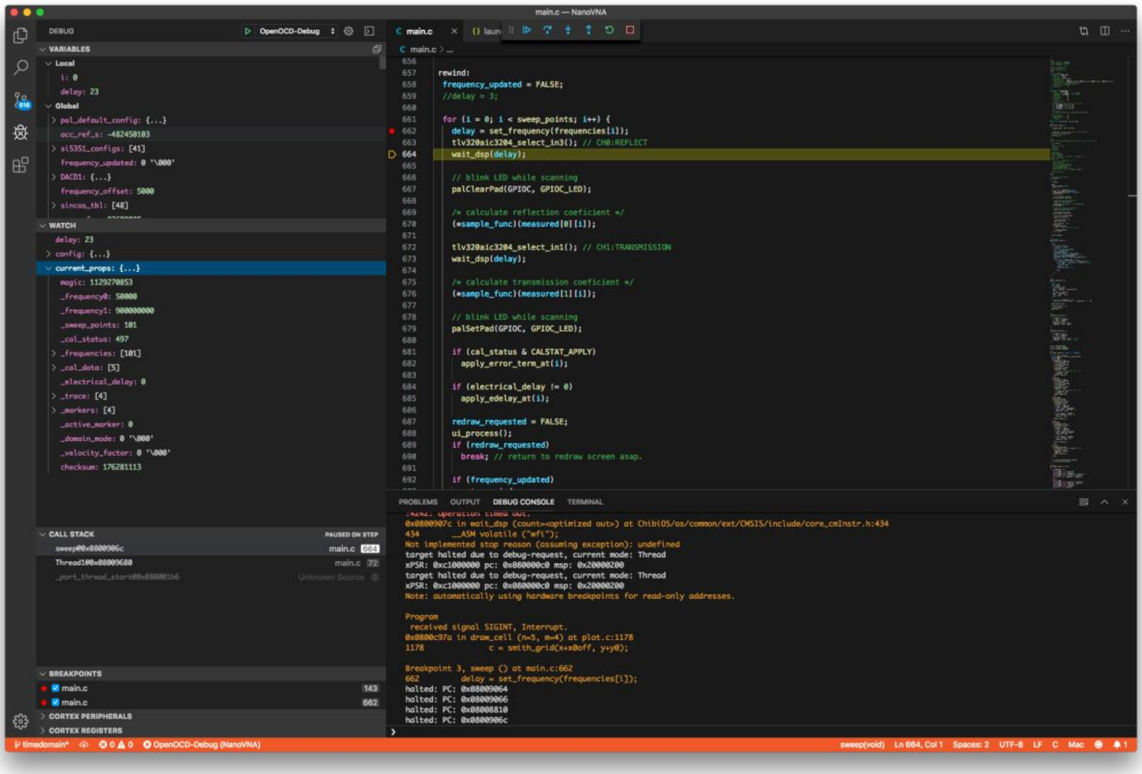
Below are the settings when using ST-Link. If you use J-Link, replace it with .interface/stlink.cfginterface/jlink.cfg

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "type": "cortex-debug",
      "servertype": "openocd",
      "request": "launch",
      "name": "OpenOCD-Debug",
      "executable": "build/ch.elf",
      "configFiles": [
        "interface/stlink.cfg",
        "target/stm32f0x.cfg"
      ],
      "svdFile": "./STM32F0x8.svd",
      "cwd": "${workspaceRoot}",
      "preLaunchTask": "build",
    }
  ]
}
```

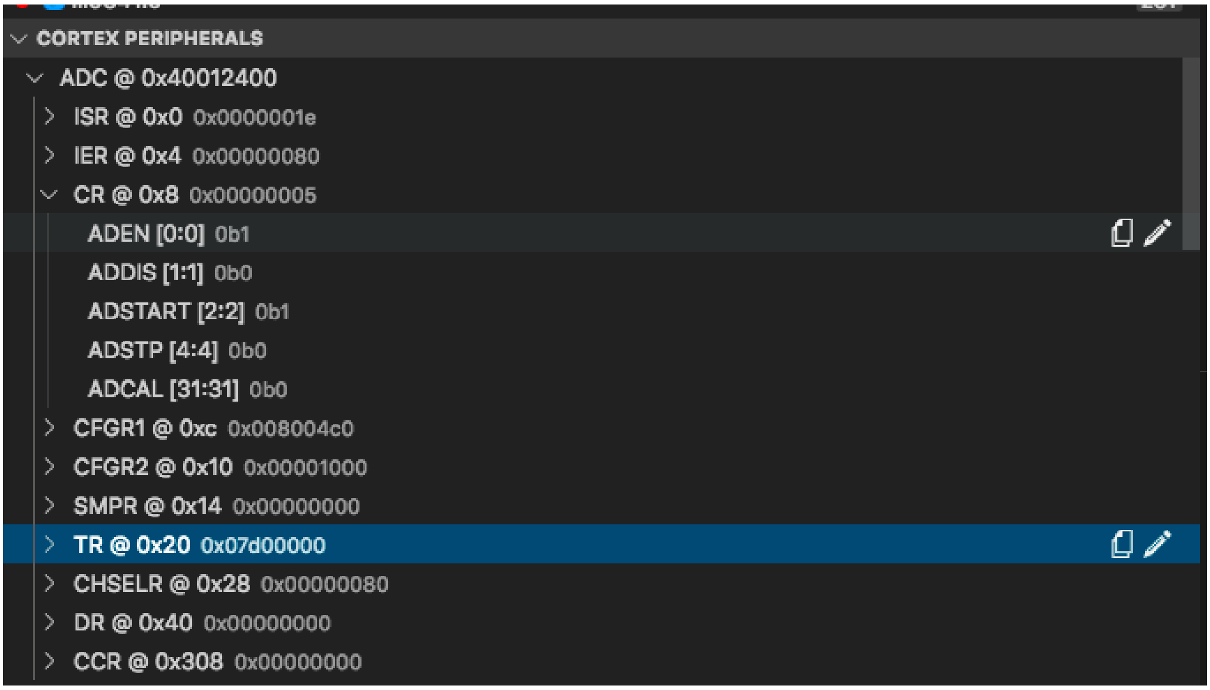
svdFile You can download the files specified in the Staple Archive website. There is no problem with operation even if you do not specify the specified option.svdFile

Start debugging

When Start Debugging () is performed, OpenOCD automatically launches and transfers the firmware after the build is made. After the transfer is complete, the reset handler will break the state.F5



svdFile If this is specified, the predefined MCU registers will be displayed on the debug screen.

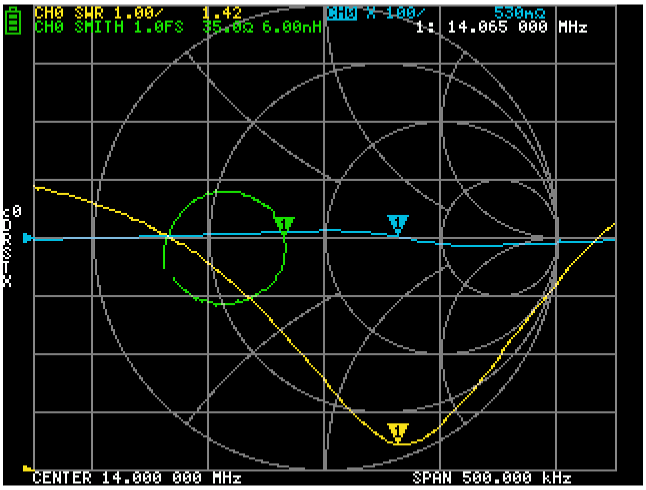


Usage Examples

Adjusting the bandpass filter

TODO

Antenna adjustment



Here is an example of using NanoVNA as an antenna analyzer.

The following two points are important when adjusting antennas.

- Whether the antenna is in a synchronized or resonant state (i.e., close to zero at the frequency targeted by reactance)
- Is the antenna's SWR low (is it a sufficient match?)

Setting up traces

Only CH0 is used for antenna adjustment, so calibration is performed for all items except and .THRUISOLN

The trace settings are as follows.

- Trace 0: CH0 SWR
- Trace 1: CH0 REACTANCE
- Trace 2: CH0 SMITH
- Trace 3: OFF

Set the frequency you want to synchronize the antenna to and set it appropriately.CENTERSPAN

Look for frequencies where trace 1 showing reactance is close to 0. Since that frequency is the tuning point, if it's off, adjust the antenna so that the tuning point matches the desired frequency.

If the tuning point is at the desired frequency, check whether the trace 0 showing SWR is sufficiently low (close to 1) SWR. If the SWR is not sufficiently (2 or less), matching is performed using the Smith chart. At this time, you can use an antenna tuner or similar device located just below the antenna for matching.

If SWR drops, it synchronizes to the desired frequency, completing the adjustment of antennas with low SWR.

Check the cable

By using the low-pass mode of the time domain, TDR can be simulated. By using TDR, you can detect faults in transmission paths.

TODO

Common Mode Filter Measurement

TODO