

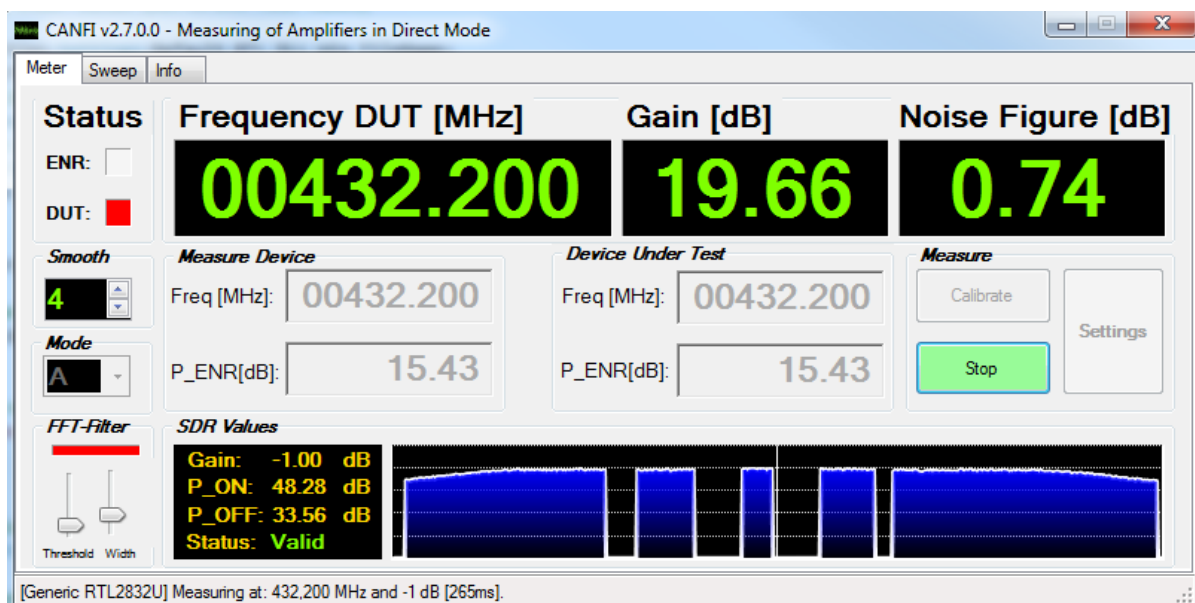
# CANFI

## V.2.7

(C)heap (A)utomatic (N)oise (F)igure (I)ndicator

*last edit: 2018-03-31*

© 2015 DL2ALF, DL8AAU, DF9IC



## Table of Content

1	Introduction .....	3
2	Theory of Noise Figure Measurement .....	4
2.1	Basic Definitions .....	4
2.2	Noise Power Measurement .....	5
2.3	Y-Factor Method .....	5
2.4	Calibration of the Measurement Device .....	6
2.5	Ambient Temperature Correction .....	7
3	Basic CANFI Concepts .....	8
3.1	Measurement .....	8
3.2	Limitations of standard DVB-T Sticks .....	9
3.2.1	Noise Figure .....	9
3.2.2	Dynamic Range .....	10
3.2.3	Thermal Drift .....	10
3.2.4	Birdies and other Irregularities .....	12
3.3	Operation Modes .....	12
3.3.1	Mode A: Measuring of Amplifiers in Direct Mode .....	12
3.3.2	Mode B: Measuring of Converters in Converter Mode .....	13
3.3.3	Mode C: Measuring of Amplifiers in Converter Mode .....	13
3.3.4	Sweep Mode .....	14
3.4	Hardware .....	14
3.5	DVB-T Stick .....	15
3.6	USB to Serial Adapter .....	15
3.7	Switchable 5V to 28V converter .....	15
3.8	Device Examples .....	16
4	Software Manual .....	17
4.1	Installation Instructions .....	17
4.1.1	DVB-T Stick Driver Installation .....	17
4.1.2	USB to Serial Driver Installation .....	17
4.1.3	CANFI Installation Procedure .....	17
4.2	Operating Instructions .....	18
4.2.1	CANFI Meter Window .....	18
4.2.2	CANFI Sweep Window .....	21
4.2.3	CANFI Info Window .....	22
4.2.4	CANFI Settings Window .....	22
5	References .....	25
6	Appendix .....	26

# 1 Introduction

One of the main tasks for an experimenting microwave amateur is to measure the Gain (G) and Noise Figure (NF) of a particular receiving device. For this one will need a Noise Figure Indicator and a (calibrated) Noise Source.

There are a number of commercial devices available from different vendors at prices which will exceed an amateur's budget by many times. A lot of them can be found on the surplus market but this doesn't help very much. A combination of both meter and noise source is barely sold below the 2.000€ margin.

Since a lot of cheap DVB – T sticks became available the idea was born to use it together with a homebrew noise source as a very cheap alternative to commercial devices [1].

It is now possible to build a suitable solution within a budget of 100 – 200€. Using a PC with USB port for communication and power supply such a device is very compact and almost compatible to an industrial solution. Special software gives a convenient user interface.

Last not least you can reuse the DVB-T stick (together with the preamplifier) as a sensitive receiver along with SDR software.

*CAUTION: This simple setup does not replace a precise commercial measurement solution. It is cheap but it has its limitations. If you expect professional results so stop reading here.*

## 2 Theory of Noise Figure Measurement

A very good description of noise measurement is given by Agilent Technologies in their application note 57-2 [7]. The following chapters are based on this document and on Wikipedia articles [5], [6].

### 2.1 Basic Definitions

The noise measurement is characterized by the following basic figures:

- $T$  is the noise temperature [K]
- $P$  is the noise power [W]
- $B$  is the total bandwidth [Hz]
- $k_B$  is the Boltzmann constant ( $1.381 \times 10^{-23}$  J/K)
- SNR is the Signal-To-Noise-Ratio, e.g. the ratio between the level of a desired signal and the level of background noise
- ENR is the Excess Noise Ratio of a noise source [linear or in dB], calibrated at  $T_0 = 290\text{K}$
- $F$  is the Noise Factor [linear]
- $NF$  is the Noise Figure [dB]
- $G$  is the Gain of a amplifier stage or a DUT
- $DUT$  is the Device Under Test

**Noise Power** is defined as the total noise per bandwidth unit measured at the input or output of a device when a signal is not present [9]. Measurement of noise power with a DVB-T means getting a set of statistical distributed I/Q - samples over the time. The noise power is calculated as:

$$(1) \quad P = \frac{1}{n} \sum_{k=0}^n (x_{I,k}^2 + x_{Q,k}^2),$$

where:

- $n$  is the sample count
- $x_{I,k}$  is the  $k$ -th I - sample
- $x_{Q,k}$  is the according Q - sample

We are also able to calculate a DC - offset out of the samples with:

$$(2) \quad P_{DC} = \left( \frac{1}{n} \sum_{k=0}^n (x_{I,k} + x_{Q,k}) \right)^2$$

According to the definition, any kind of noise (incl. "White Noise") has a mean value = 0 and should be free of any DC - offset. If we see one, this is an artefact and most likely caused by the tuner receiving technology. It should be removed from the result so that we get:

$$(3) \quad P_{NOISE} = P - P_{DC} = \frac{1}{n} \sum_{k=0}^n (x_{I,k}^2 + x_{Q,k}^2) - \left( \frac{1}{n} \sum_{k=0}^n (x_{I,k} + x_{Q,k}) \right)^2$$

To express the noise power in dB we use:

$$(4) \quad P_{NOISE}[dB] = 10 \times \log_{10}(P_{NOISE})$$

**Signal-To-Noise-Ratio** is defined as the ratio of the power of a signal (meaningful information) and the power of the background noise [10]. The linear expression is:

$$(5) \quad SNR = \frac{P_{SIGNAL}}{P_{NOISE}}$$

where P is the average power measured at the same or equivalent points in a system and within the same system bandwidth. Expressing in decibel we get:

$$(6) \quad SNR[dB] = 10 \times \log_{10}(SNR)$$

According to the quotient rule for logarithms for both power values in decibel:

$$(7) \quad SNR[dB] = P_{SIGNAL}[dB] - P_{NOISE}[dB]$$

**Noise Temperature T** is one way of expressing the level of available noise power introduced by a component or source. T is the temperature a real ohmic resistor must have to produce the equivalent amount of noise power [6]:

$$(8) \quad T = \frac{P}{k_B \times B}$$

**Noise Factor F** is the amount of degradation of the signal-to-noise ratio (SNR) in a signal chain. It is expressed in its equivalent noise temperature  $T_e$  in relation to the standard noise temperature  $T_0$  at 290K.

$$(9) \quad F = 1 + \frac{T_e}{T_0}$$

**Noise Figure NF** is simply the Noise Factor F expressed in [dB]:

$$(10) \quad NF = 10 \times \log_{10}(F)$$

## 2.2 Noise Power Measurement

## 2.3 Y-Factor Method

The Y-factor method is a widely used technique for measuring the gain G and the noise factor F of an amplifier [5].

The principle is to measure the noise power output of a system at two different (and known) noise temperatures. A calibrated, switchable noise source is used to generate these two points:

- Noise source OFF (Cold): thermal noise of a resistor connected to the input
- Noise source ON (Hot): thermal noise of this resistor + noise from a noise source

The noise power for both cases ( $P_{ON}$ ,  $P_{OFF}$ ) is measured at a distinct (constant) bandwidth and together with the known Excess Noise Ratio (ENR) of the noise source we get:

$$(11) \quad Y = \frac{P_{ON}}{P_{OFF}} = \frac{T_{ON}}{T_{OFF}}$$

and the noise factor for the whole system as:

$$(12) \quad F = \frac{ENR}{(Y-1)}$$

As we are measuring a ratio between  $P_{ON}$  and  $P_{OFF}$  the absolute gain of the system does not affect the result. But there are some other facts that have influence (see next chapters).

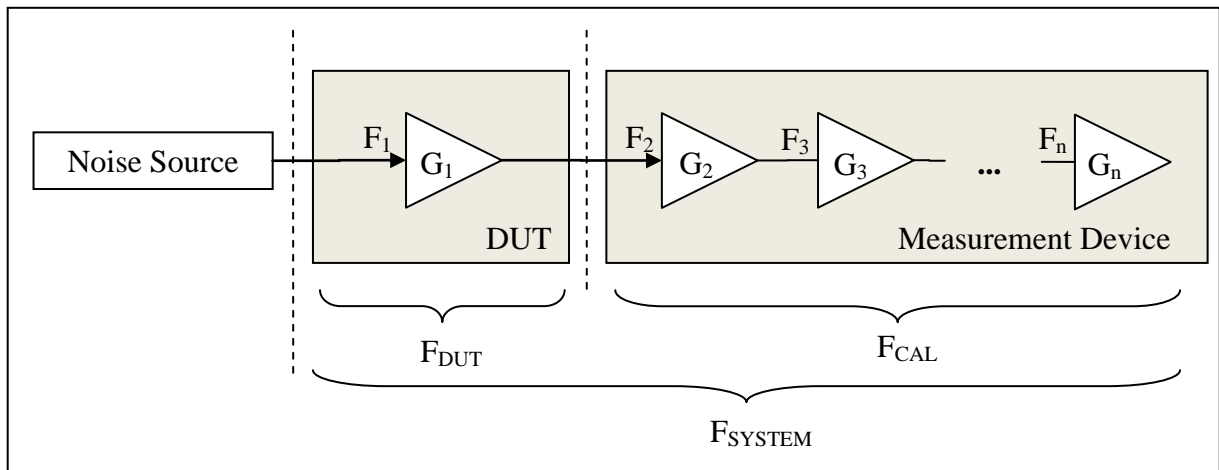
## 2.4 Calibration of the Measurement Device

The simple, uncalibrated measurement of the noise factor  $F_{SYSTEM} = F$  of the whole system according to (12) has one main deficiency: instead of measuring the noise figure of the DUT itself we get the "overall" noise figure of the whole system (DUT + measurement device).

The resulting noise factor of such cascaded devices is described by the Friis equation [8]:

$$(13) \quad F = F_1 + \frac{F_2 - 1}{G_1} + \frac{F_3 - 1}{G_1 G_2} + \dots + \frac{F_n - 1}{G_1 G_2 \dots G_{n-1}},$$

where  $F_n$  is the noise factor of the n-th device and  $G_n$  is the power gain of the n-th device (all linear values). Reflecting the formula to our measurement configuration it looks like this:



Now treating the whole measurement device as "one big stage No.2": if we were able to measure the noise factor  $F_2 = F_{CAL}$  and  $G_1 = G_{DUT}$  we could easily calculate  $F_{DUT}$  out of  $F_{SYSTEM}$ . That is exactly what a calibration process is intended for.

We are starting the calibration with connecting the measurement device itself to the noise source without the DUT in place. A calibration point consists of  $P_{CAL\_ON}$ ,  $P_{CAL\_OFF}$  and  $F_{CAL}$  measured at  $ENR_{CAL}$ .

At first we can calculate the gain of the DUT:

$$(14) \quad G_{DUT} = \frac{(P_{ON} - P_{OFF})}{(P_{CAL\_ON} - P_{CAL\_OFF})} \times \frac{ENR}{ENR_{CAL}} \quad (\text{while } ENR = ENR_{CAL} \text{ in most cases})$$

With  $G_{DUT}$  as a result and calculating  $F_{CAL}$  according to (12) :

$$(15) \quad F_{CAL} = \frac{ENR_{CAL}}{(Y_{CAL} - 1)}$$

we can correct the value of  $F_{SYSTEM}$  from (12) to:

$$(16) \quad F_{DUT} = F_{SYSTEM} - \frac{(F_{CAL} - 1)}{G_{DUT}}$$

When in AGC auto mode CANFI is performing a calibration through all available gains. The calibration needs some cycles to complete: the higher the measured noise figure, the more calibration cycles are needed to get a statistical significant value. The results are kept in memory for the following measurements.

*CAUTION: The calibration is only valid for one distinct frequency. Changing the frequency will invalidate all calibrations immediately.*

## 2.5 Ambient Temperature Correction

Noise measurements refer to an ambient temperature  $T_0$  of  $290K \approx 17^\circ C$  by definition. The real ambient temperature  $T_{AMB}$  is higher in most cases and is affecting  $T_{OFF}$  of the noise source (and further: its real ENR and the measured noise factor). In order to work with the calibrated ENR values and to refer the noise factor of the DUT to  $T_0$ , a temperature correction is necessary.

The correction is done in uncalibrated mode:

$$(17) \quad F_{SYSTEM} = \frac{ENR}{(Y - 1)} + 1 - \frac{T_{AMB}}{T_0}$$

and in calibrated mode:

$$(18) \quad F_{DUT} = F_{SYSTEM} - \frac{\left(F_{CAL} - \frac{T_{AMB}}{T_0}\right)}{G_{DUT}}$$

*CAUTION: The temperature correction does only ensure the correct reference of the measurement. It does not correct the higher noise factor of your DUT at higher temperatures!*

### 3 Basic CANFI Concepts

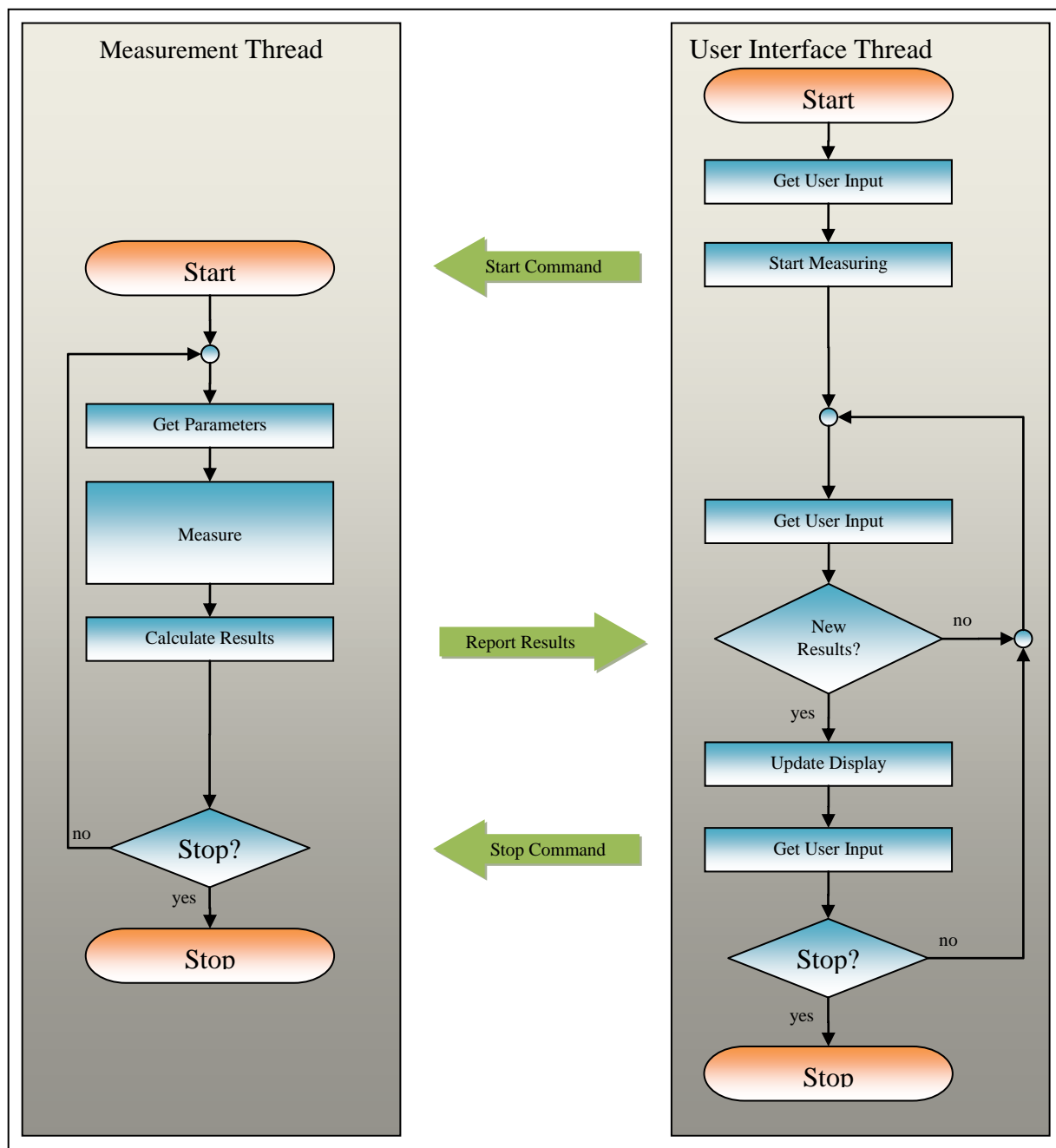
#### 3.1 Measurement

The basic idea behind CANFI is to have a smart software solution to switch a noise source ON/OFF via serial port and to control a DVB-T stick to set a distinct frequency and to get sampled values for noise calculation.

CANFI software consists of two main parts:

- The user interface (UI thread)
- The measurement thread

In a simplified form the flow chart looks like this:





The user interface takes all the user input (e.g. mode, frequency, start/stop, settings), controls the measurement thread and presents the measurement results.

The measurement thread, once started, controls the serial port and DVB-T stick (e.g. frequency, gain) and gets the samples in a loop. After measurement is done, the results are reported back to the user interface. Running two threads this way is separating the user input from the cyclic (and non-interruptible) measurement.

The set of sampled I/Q-values over the time is representing the “White Noise” from the noise source. The noise signal should have an uniform distributed spectrum. It is possible to calculate the power from these raw values directly. In order to have a frequency display and a filter algorithm, CANFI is transforming the raw values from time domain into frequency domain with a Fast Fourier Transformation (FFT). The results are values for each single frequency in the pass band. As they are representing the same “White Noise”, the values should be as uniform distributed as the raw values.

## 3.2 Limitations of standard DVB-T Sticks

The use of a cheap consumer product not intended for use as a noise figure meter is resulting in a couple of limitations as they are:

- High noise figure of the input stage
- 8bit ADC is limiting the dynamic range
- Thermal instabilities are causing drift of power measurement
- Birdies and “holes” may occur in the frequency response

Some of the limitations can be covered by the software. It is necessary to consider the limitations in detail to understand what CANFI can do and what it cannot.

### 3.2.1 Noise Figure

The noise figure stated in the data sheets of both tuner chips (E4000, R820T) is far from the real noise figure of the complete DVB-T stick! There are at least two limiting facts:

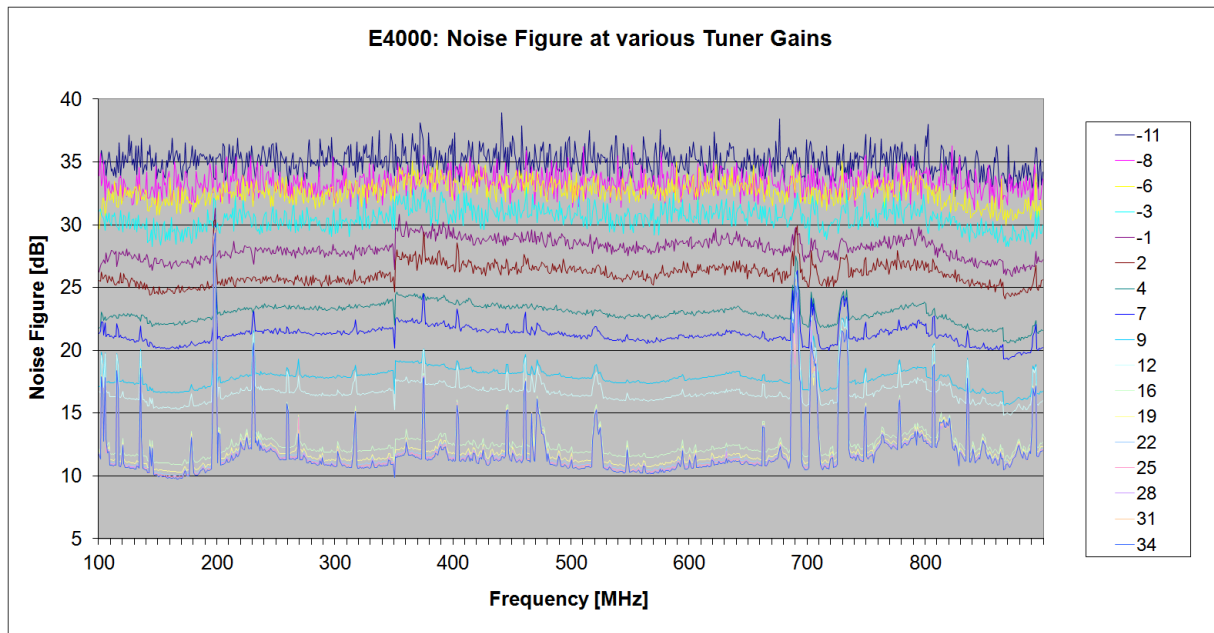
- The antenna input circuitry is containing EMC diodes and other protection stuff
- The input impedance is 75 Ohms instead of 50 Ohms
- The noise figure dramatically increases at lower gain settings of the tuner stages

There could be other facts as well, like the PCB layout which will give different noise figures on several DVB-T sticks with basically the same tuner chip.

Some measurements on industrial DVB-T sticks gave in the range from 100MHz ... 1300MHz (full sensitivity!):

E4000:	10dB ... 14dB	(Data sheet: 3.8 ... 4.5dB @ 75Ohms)
R820T:	6dB ... 12dB	(Data sheet: 3.5dB @ 75Ohms)

As an example, this is a noise measurement of an industrial DVB-T stick with E4000 tuner in the range of 100MHz .. 1000MHz (FFT filter OFF):



As you can see, the noise figure is abt. 10dB @ full sensitivity over a wide frequency range which is well above the specification. Furthermore, a lot of spurious measurements with heavily increased noise figures is seen, probably caused by external or internal carriers.

Such high values (especially on higher frequencies) will seriously affect the measurement results. As a conclusion the DVB-T sticks should not be used without calibration or with preamplifier.

### 3.2.2 Dynamic Range

The dynamic range is limited to about 40dB. This is caused by the 8bit – A/D conversion and may be even lower in practical use while measuring white noise.

The measurement of DUTs with a gain > 30 ... 40dB is therefore not possible with this setup. Nevertheless CANFI is able to measure usual DUTs with 20dB gain without problems. Furthermore to get the best results, CANFI is containing a software based Automatic Gain Control (AGC) of all available tuner stages. This will keep the tuner gain at the highest possible value without getting the output clipped or non-linear. The AGC is checking the measured power and limits it to the following values:

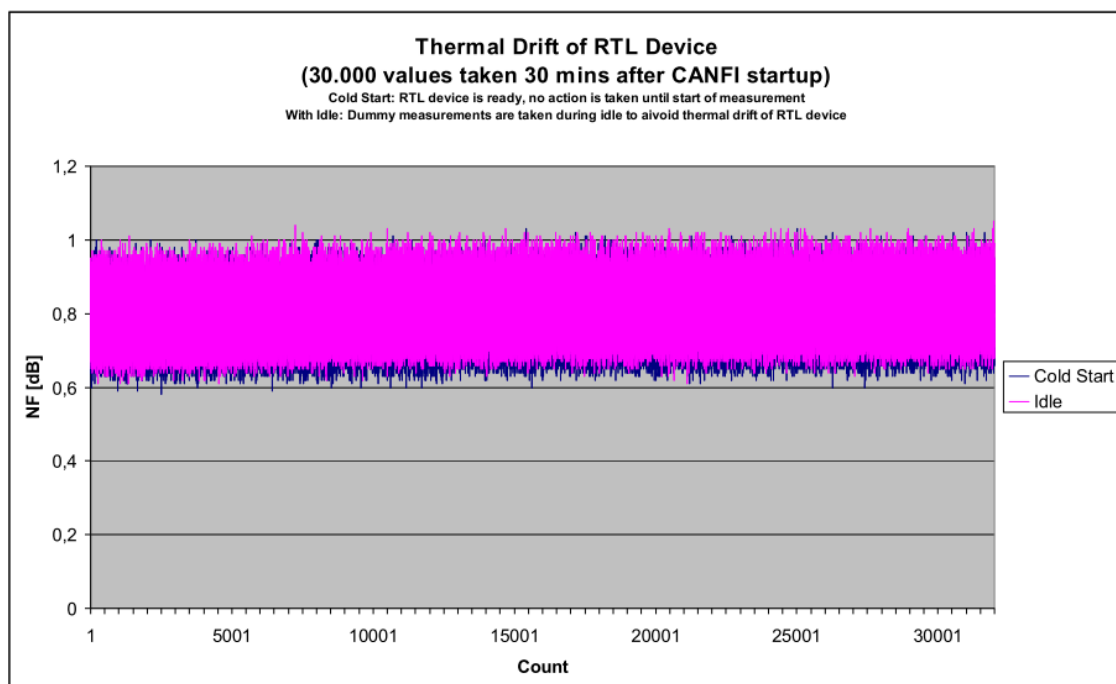
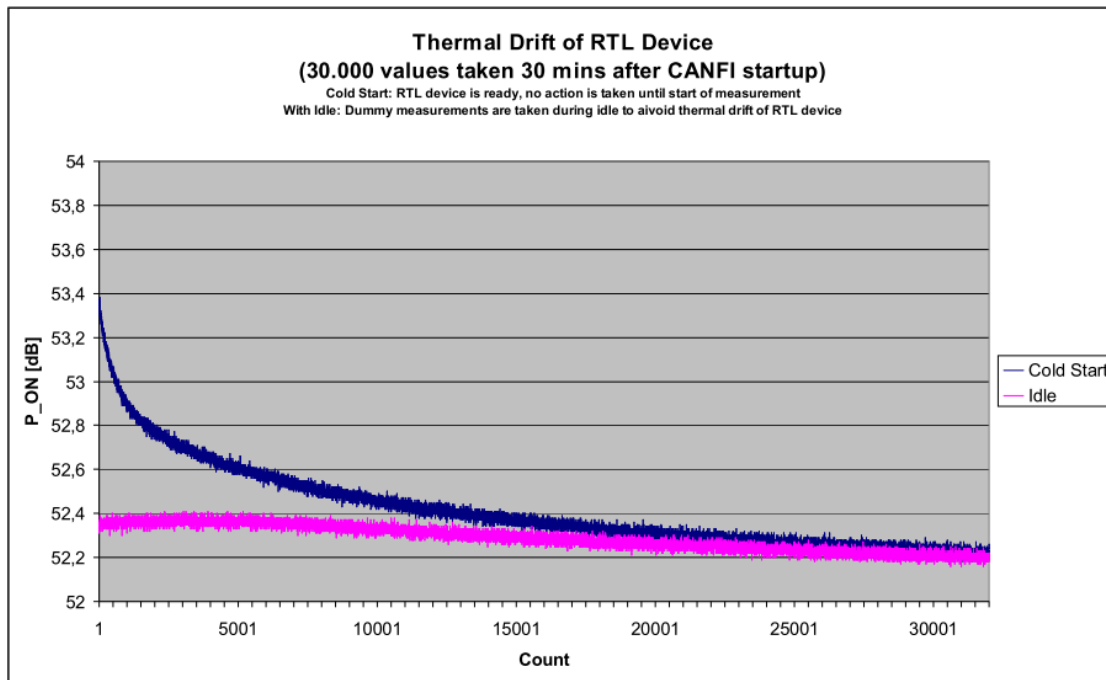
- E4000: 50dB
- R820T: 45dB

### 3.2.3 Thermal Drift

The power measurement seems to be not very stable with recent DVB-T sticks. This is most likely caused by the tuner chip rather than by the ADC inside the RTL2832U and does very much depend on the design of the DVB-T stick. It's a good idea to cool down the tuner chip with a copper clamp when opening the case of the stick anyway.

The drift should not be a big issue as far as the values of P\_ON are drifting the same manner as P\_OFF. As all calculated values are basing on the ratio P\_ON/P\_OFF, the drift should be

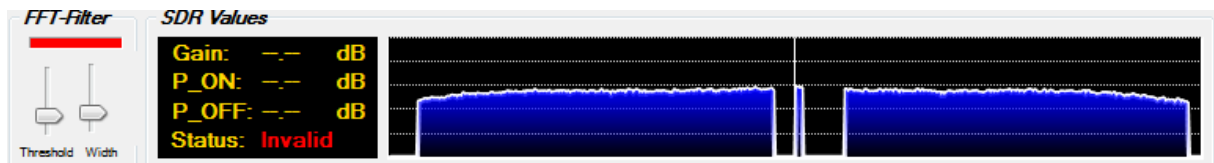
well compensated. Nevertheless an “Idle” measurement procedure was implemented to keep the device in measurement mode all the time.  
As an example, these are two diagrams showing the drifting measurement of noise power and noise figure comparing cold start and idle:



### 3.2.4 Birdies and other Irregularities

Depending on the tuner design several birdies and other irregularities like noise peaks may occur over the frequencies affecting the measured noise power values. Furthermore, the E4000 is producing a small DC component in the pass band. These effects are usually higher during NOISE\_OFF when the tuner is not on the edge of its dynamic range. As a result, measured P\_OFF is too high, the calculated Y-factor is too low, and, at the end, the calculated noise figure is too high.

CANFI provides a special FFT-filter algorithm to notch out discrete carriers in the pass band automatically. You can adjust the threshold and the width of the notches online. It will not work with broadband noise peaks.



The DC component is always removed while calculating the power from the measured values.

### 3.3 Operation Modes

CANFI can work in three basic operation modes. Depending on your setup you have to choose a direct mode or one of the converter modes. The different modes handle:

- the frequency setting of the RTL device
- the frequency from which the ENR value of the noise source is taken during calibration
- the frequency from which the ENR value of the noise source is taken during measurement

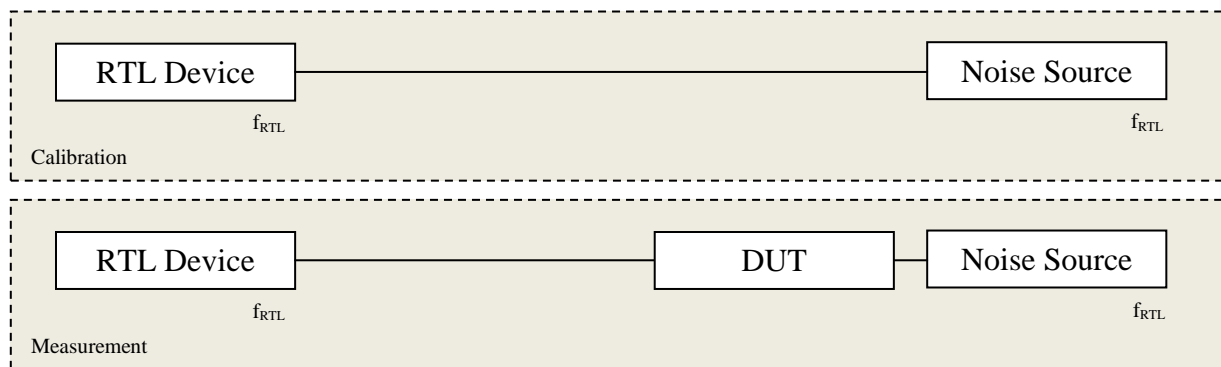
This is the most basic setting for the whole calibration and measurement process. See the following table for details about the frequency input usage:

Mode	Tuner RX Frequency from	ENR Value for Calibration from	ENR Value for Measurement from
A	Measure Device Frequency Input	Measure Device Frequency Input	Measure Device Frequency Input
B	Measure Device Frequency Input	Measure Device Frequency Input	DUT Frequency Input
C	Measure Device Frequency Input	DUT Frequency Input	DUT Frequency Input

#### 3.3.1 Mode A: Measuring of Amplifiers in Direct Mode

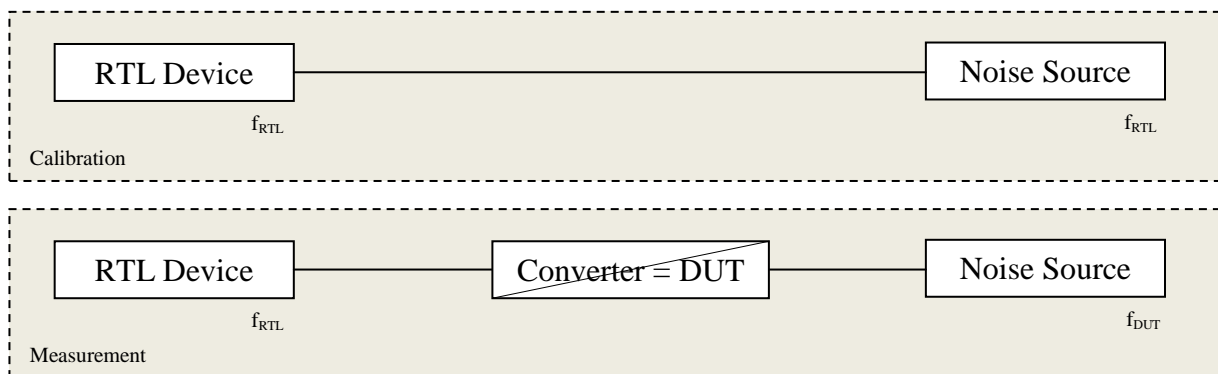
This is the most common mode. Use it whenever the RTL device can handle the frequency directly (without converter). Enter the frequency into the RTL device frequency input control.

This frequency value will be used for setting the tuner's receiving frequency and getting the ENR value of the noise source. The DUT frequency input is disabled.



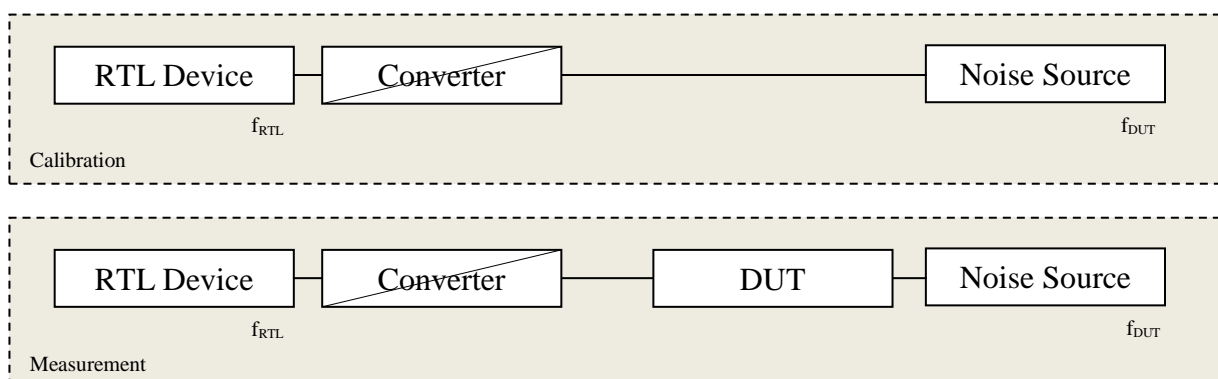
### 3.3.2 Mode B: Measuring of Converters in Converter Mode

Use this mode if you want to measure a converter. Enter the tuner's receiving frequency (which is the IF of your converter) into then RTL frequency input control. Enter the converter's receiving frequency into the DUT frequency input control.



### 3.3.3 Mode C: Measuring of Amplifiers in Converter Mode

Use this mode when you are measuring an amplifier outside the receiving range of your RTL device with the help of a converter. Enter the tuner's receiving frequency (which is the IF of your converter) into then RTL frequency input control. Enter the converter's receiving frequency into the DUT frequency input control.



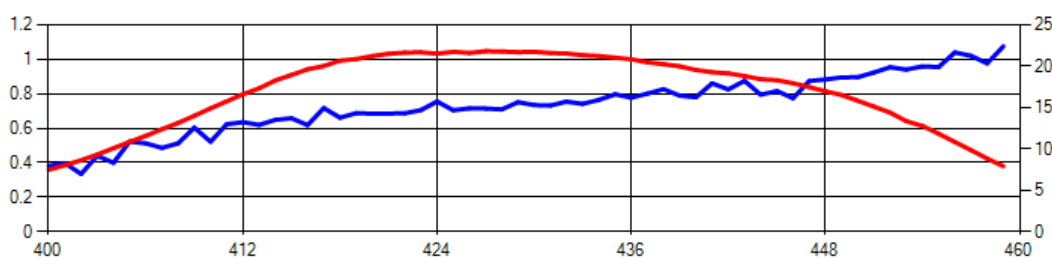
### 3.3.4 Sweep Mode

Furthermore, CANFI has an experimental "Sweep Mode" which allows measuring in either mode A, B or C to cover a frequency range. You can enter:

- a start frequency
- a stop frequency
- a frequency step width
- a sweep mode ("NONE", "SINGLE" or "CONTINUOS")

The software sweeps through the frequency range and plots the noise figure and gain (if calibrated before).

As an example, this is a sweep taken between 400MHz and 460MHz with a 70cm pre-amplifier as DUT.



*CAUTION: Depending on the frequency range and step width, a sweep can be a very time consuming process. Therefore it is not recommended to use AGC especially when calibrating.*

## 3.4 Hardware

The hardware prerequisites are the following:

Assuming an up-to-date PC hardware with Windows XP, Vista, 7, 8 or 8.1 is available the hardware prerequisites for the CANFI device itself are listed below:

- A DVB-T stick with an Elonics E4000 or a Rafael R820Txx tuner
- A USB to Serial adapter
- A switchable 5V to 28V converter (homebrew)
- A broadband low noise preamplifier (homebrew)

And optional:

- An USB switch to get a single cable solution
- An USB stick for software, drivers and documentation

The complete CANFI hardware fits into a standard euroboard case and is connected to the PC with one USB cable.

The PC must have a free USB 2.0 port or an USB 3.0 port in legacy mode. The current version of the DVB-T driver does not run on a native USB 3.0 port. Be sure to connect the CANFI device to an USB port with full power supply capabilities. The hardware will consume about 0.5A when noise source is on. You should check your own device to meet the USB standard. If not, use an external power supply for your USB hub.

### 3.5 DVB-T Stick

Up to now, CANFI supports two types of DVB-T sticks:

- Elonics E4000 tuner and
- Rafael R820T/R820T2 tuner

Both of them have the Realtek RTL2832U as an A/D-converter inside. This is the only combination for which drivers and software is available. The E4000 is marked end of life and hard to get but some of devices are still available in several online shops. The R820Txx is very popular and is part of the most DVB-T sticks sold at present. If you are going to buy a device read the hardware specification carefully. No other device or combination is supported yet. To increase noise immunity remove the plastic cover and antenna jack. Then solder it together with the preamplifier in a tinfoil box for shielding as shown in [1].

### 3.6 USB to Serial Adapter

The USB to serial adapter is used for controlling the noise source power supply by status line. The second status line can control a DUT switch optionally.

The use of a Silicon Labs CP2102 based adapter is strongly recommended. The USB to COM Bridge which comes with the driver software is mapping the CP2102 device to a fixed COM port number regardless of the connected USB port. A lot of them (sometimes with breakout cables) are listed in several online shops. The price is < 10€ / pc.

Other chipsets (including the very popular FTDI devices) will work but you will get different COM Port numbers for each USB port depending on how the Windows system enumerates them.

Use a status line of your choice (RTS or DCD) to control the 5V to 28V converter. It is recommended to solder it together with the converter in a tinfoil box as shown in [1].

### 3.7 Switchable 5V to 28V converter

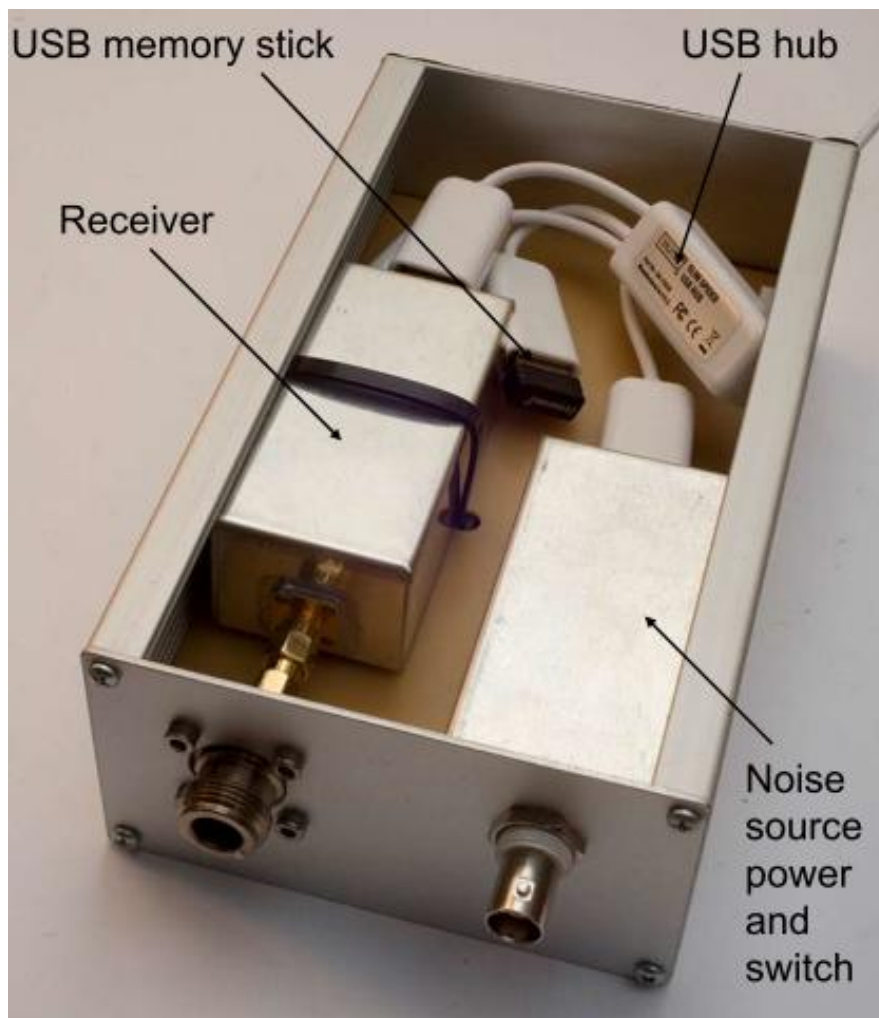
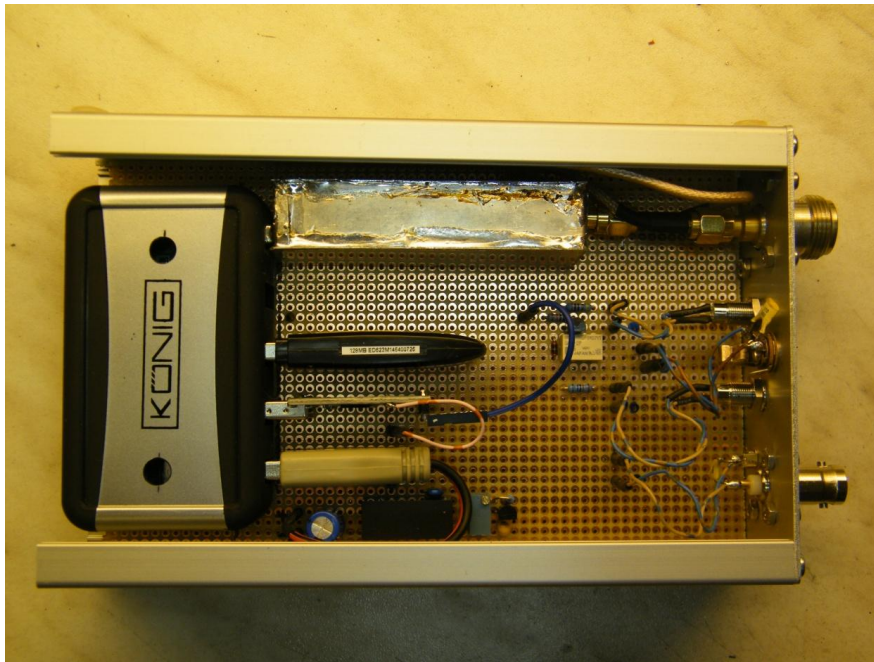
To meet the industrial standard of noise source power supply a switchable 28V DC source is necessary. This can be done easily by using the converting the 5V USB power supply. Be sure that the converter is well shielded to avoid negative influence on the noise measurement. The switched output should be a sharp edged rectangle and should follow the command input with almost no delay. The output voltage should be well stabilized at 28V (+/- 0.1V). Best result can be achieved using a 5V to 30V+ step-up converter and a discrete LM3xx stabilization circuit. Switching can be done with a MOSFET or a modern high side switch.

Be aware from any soft start behavior as this will screw up your measurement!

Several proven designs are available, see [1].



### 3.8 Device Examples





## 4 Software Manual

The CANFI software should run on any PC with Windows XP, Vista, 7, 8 or 8.1. Microsoft's .NET4.0 package is required; it should be preinstalled on all newer versions of Windows. If not, go to the Microsoft homepage and download the needed installation files.

### 4.1 Installation Instructions

#### 4.1.1 DVB-T Stick Driver Installation

To use the DVB-T stick for other than TV applications you must install the Zadig driver package first. This is basically the same procedure as using the stick for SDR software like SDR#. This package is not provided along with the CANFI software. Follow the steps below:

1. Download the suitable version from [2]
2. Install the driver following the instructions on [2]
3. Open the Windows Device Manager. You should see there a new device named "RTL2832U device" or "Bulk-In, Interface"

A quick start guide with step-by-step instructions made for SDR# installation is available from [3].

#### 4.1.2 USB to Serial Driver Installation

If the USB to Serial converter is not recognized by Windows itself use the driver software provided with the device itself. If you are using a CP2102 device you can download and install the VCP driver for your system from the Silicon Labs homepage [4]. Looking into the Windows Device Manager you should see a new COM port on your system.

#### 4.1.3 CANFI Installation Procedure

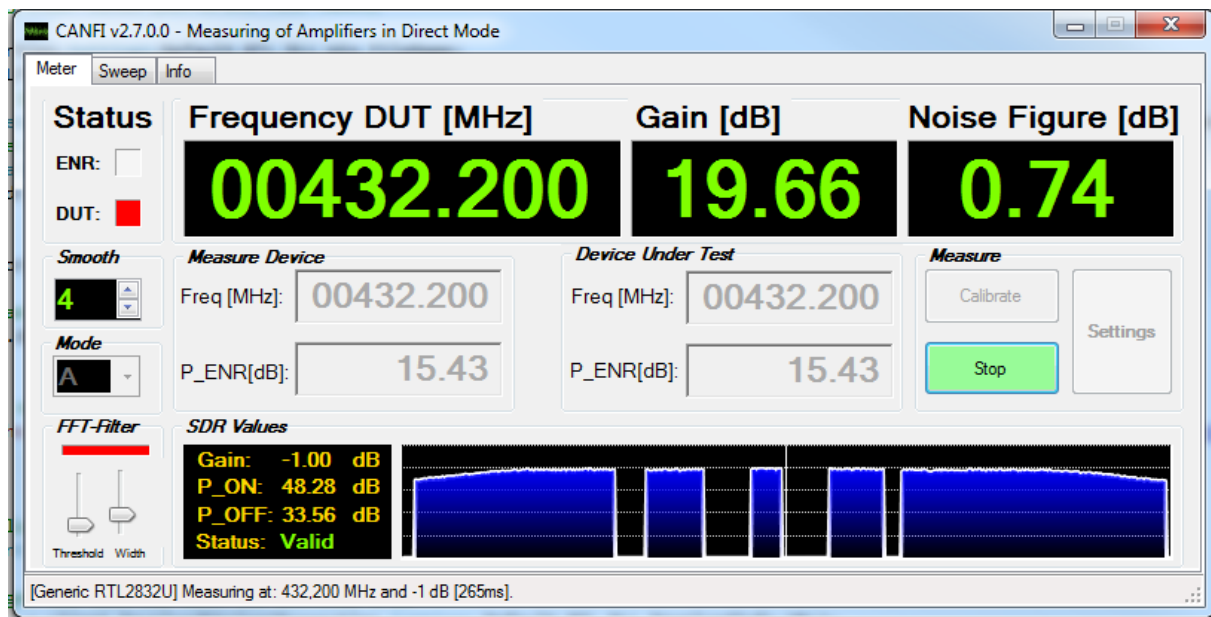
There is no special installation procedure for the CANFI software itself. Just follow the steps below:

1. Download the latest version of CANFI software from the website
2. Unzip it into a directory of your choice including all subdirectories.  
Be sure to have write permissions for that directory.  
Please do not use "C:\Program Files\..." or similar as there are restrictions given by the Windows system.
3. Run CANFI.EXE
4. Go to "Settings" and select the DVB-T device and COM-Port

*Caution: After selecting a DVB-T stick for the very first time it may happen that the initialization of the device does not work properly. You will see a red error message after closing the "Settings" window. A restart of CANFI is necessary in that case.*

## 4.2 Operating Instructions

### 4.2.1 CANFI Meter Window



#### Title Bar

Is showing the CANFI software version and measuring mode.

#### Tabs

Is showing the several available tab pages for:

- Meter
- Sweep
- Info

CANFI is always starting with the Meter tab selected.

#### ENR LED

Flashes red when noise source status line is on. A 28V DC output is generated to supply the external noise source.

#### DUT LED

Flashes red when DUT switch status line is on (optional, additional hardware required).

#### Frequency DUT Display

You can see the input frequency in MHz of the Device under Test (DUT) you have chosen. This is usually the same frequency as chosen in the DUT frequency setting control below.

#### Gain Display

Is showing the calculated gain of the DUT in dB. A gain value is only available if a calibration was made before measurement.

#### Noise Figure

Is showing the noise figure of the DUT in dB.

#### Smooth

Is showing the current smoothing level of the displayed values. The value n stands for a floating average of  $2^n$  measurements.

#### Mode

Is showing the current operation mode.

#### Measure Device Frequency Input

Enter the Measure Device Frequency in MHz here. You can use numeric keys or cursor keys for altering the value.

#### Measure Device ENR Input

Enter the ENR value of the noise source for the given frequency here manually. This input is only available if the ENR values are not taken from a calibration file automatically.

#### Device under Test Frequency Input

Enter the DUT Frequency in MHz here. You can use numeric keys or cursor keys for altering the value. This input is only enabled in Mode B and C.

#### Device under Test ENR Input

Enter the ENR value of the noise source for the given frequency here manually. This input is only available if the ENR values are not taken from a calibration file automatically.

#### “Calibrate” Button

Press the button to start/stop a calibration process. If a calibration process is running the button will change its capture to “Stop”. Button will change its color to orange when a calibration was successfully finished. Button is disabled when a measurement is running.

#### “Measure” Button

Press the button to start/stop a measurement. If a measurement is running the button will change its capture to “Stop”. Button is disabled when a calibration process is running.

#### “Settings” Button

Press the button to see the Settings Window (described below). Button is disabled during calibration and measurement.

### FFT Filter

Is showing the FFT-Filter settings and state. A small red bar above the two controls appears red when FFT-Filter is on. You can switch it on/off while clicking on the "LED". The two controls are for changing the filter threshold and filter width. Use it to notch out birdies from the spectrum.

### SDR Values

Is showing basic SDR values of the measuring device, as they are:

Gain:	current tuner gain
P_ON:	calculated RF power while noise source is ON
P_OFF:	calculated RF power while noise source is OFF
Status:	Status of measurement (valid/invalid)

### Spectrum Display

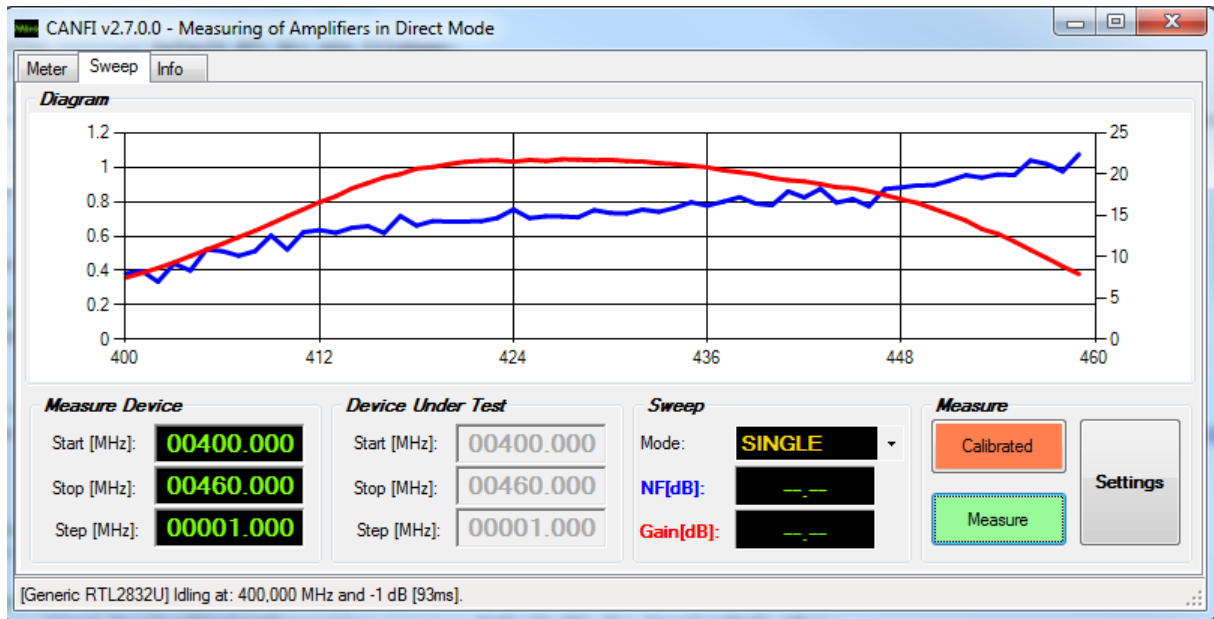
Shows the current spectrum of the measurement for information. As we are receiving white noise only the spectrum should be nearly flat over the whole range. Lower values at the edges are caused by sampling and filtering inside the ADC. A small peak at the center of the spectrum shows some DC parts of the signal caused by the tuner receiver design. Any other peak shows external carriers which are unwanted and may cause false measurements. Use the FFT-Filter to notch them out. You will see the result immediately as gaps will occur on the spectrum display.

*CAUTION: The use of the spectrum display in real time will consume a lot of CPU performance!*

### Status Bar

Is showing basic status information on the left and errors on the right (in red).

## 4.2.2 CANFI Sweep Window



### Diagram

Is showing the values for noise figure (Blue curve, left axis) and, when calibrated, the gain (Red curve, right axis) over the frequency.

### Measure Device

Is showing the basic sweep input for the measure device as they are:

- the start frequency for sweep
- the stop frequency for sweep
- the frequency step width for sweep

### Device Under Test

Is showing the sweep values calculated from the measure device. In mode B and C the start frequency for sweep can be entered manually. The other values are calculated.

### Sweep

Set the sweep mode here. You can choose between:

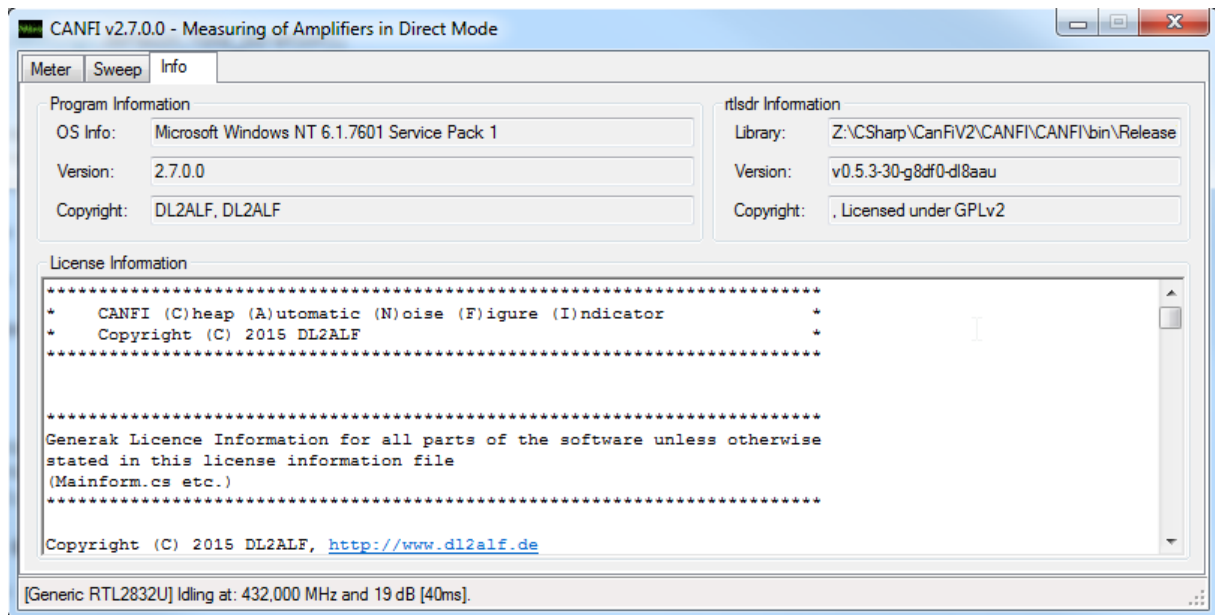
- none
- single sweep
- continuous sweep

While sweeping, the current values for noise figure and gain are shown.

### Measure

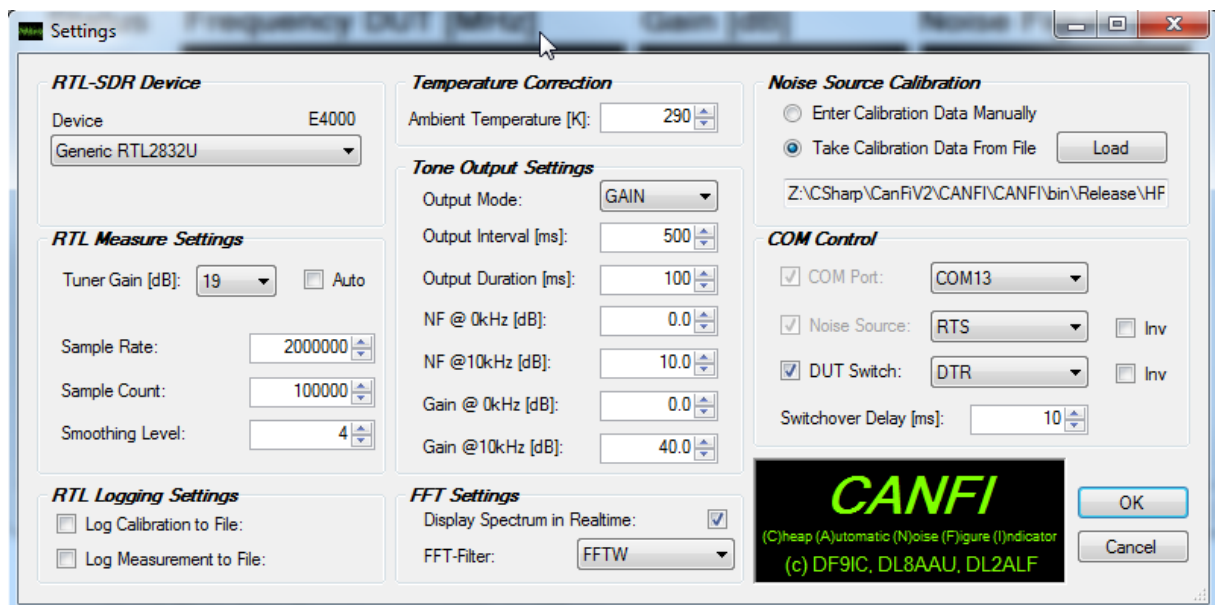
Buttons for calibration, measurement and settings.

### 4.2.3 CANFI Info Window



Is showing basic version information of the CANFI main program and of the rtlsdr.dll. In a text box the content of the LICENSE file is shown.

### 4.2.4 CANFI Settings Window



#### RTL SDR Device

Select your DVT-T stick here. If more than one stick is present on your system be sure to select the right one! A text above the drop down list is showing the tuner type of the selected device.

#### RTL Measure Settings

Make some basic settings for RTL measurement here:

<i>Tuner Gain:</i>	<i>select a fixed tuner gain manually or tick "Auto" for Automatic Gain Control</i>
<i>Sample Rate:</i>	<i>Sample rate of the RTL device in bit/sec (default: 2.000.000 = 2MHz bandwidth)</i>
<i>Sample Count:</i>	<i>Count of samples taken in a single shot (default: 100.000)</i>
<i>Smoothing Level:</i>	<i>Initial level of smoothing (see above)</i>

### RTL Logging Settings

Switch on/off logging to file here for:

- Calibration values
- Measured values

If enabled, log files in CSV-Format are written in the program's main directory. See appendix for file format.

### Temperature Correction

Used to correct the result while measuring under different ambient temperatures. Default is 290K = 17°C.

### Tone Output Settings

Define the tone output settings here. You can choose a tone output either:

- no output
- tone frequency according to  $NF_{DUT}$
- tone frequency according to  $G_{DUT}$

If enabled, a series of beeps is generated. You can set the output interval and duration. To adjust the tone frequency to your needs you can set the lower (0kHz) and upper (10kHz) bounds of noise figure and gain.

### FFT Settings

Enable/Disable the real time spectrum display here. Showing a FFT in real time needs a lot of CPU performance. Switch it off in case of small PC's or embedded solutions. Furthermore, you can change the FFT procedure here via drop down list. Currently two procedures are available:

- A simple C# implementation (© Chris Lomont, see <http://www.lomont.org> )
- A very fast DLL in native C (FFTW, see <http://www.fftw.org> )

The use of FFTW is recommended. Change it only if you are experiencing troubles.

### Noise Source Calibration

Change your ENR input of your (calibrated) noise source here. You can either:

- Input the data manually

- Use a calibration file to get the ENR values automatically according to frequency

For automatic mode you have to select a suitable calibration file. The file structure is described in the Appendix.

### COM Control

Select your COM port to control the noise source supply (mandatory) and DUT switch (optional). You can choose between the RTS and DTR status line. Furthermore you can invert the signal if your hardware is requiring a negative logic.

A switchover delay can be set to guarantee that the switchover between noise source ON and OFF is finished before start sampling.



## 5 References

- [1] CANFI presentation by DF9IC at the Martlesham Microwave Roundtable 2014  
([http://www.df9ic.de/doc/2014/mmrt\\_2014/mmrt14\\_CANFI.pdf](http://www.df9ic.de/doc/2014/mmrt_2014/mmrt14_CANFI.pdf))
- [2] Homepage of the Zadig project  
(<http://zadig.akeo.ie/>)
- [3] Quick Start Guide from RTL-SDR Blog  
(<http://www.rtl-sdr.com/rtl-sdr-quick-start-guide/>)
- [4] Silicon Labs CP2102 VCP driver download page  
(<http://www.silabs.com/products/mcu/pages/usbtouartbridgevcpdrivers.aspx>)
- [5] Y-Factor on Wikipedia  
(<http://en.wikipedia.org/wiki/Y-factor>)
- [6] NoiseTemperature on Wikipedia  
([https://en.wikipedia.org/wiki/Noise\\_temperature](https://en.wikipedia.org/wiki/Noise_temperature))
- [7] Y-Factor Measurement fundamentals from Agilent  
(<https://web.stanford.edu/class/ee133/appnotes/5952-8255E.pdf>)
- [8] The Friis formulas for noise  
([https://en.wikipedia.org/wiki/Friis\\_formulas\\_for\\_noise](https://en.wikipedia.org/wiki/Friis_formulas_for_noise))
- [9] Noise power on Wikipedia  
([https://en.wikipedia.org/wiki/Noise\\_power](https://en.wikipedia.org/wiki/Noise_power))
- [10] Signal-to-noise-ratio on Wikipedia  
([https://en.wikipedia.org/wiki/Signal-to-noise\\_ratio](https://en.wikipedia.org/wiki/Signal-to-noise_ratio))

## 6 Appendix

### Noise Source Calibration File Format

The noise source calibration file contains the calibration data of a given noise source to simplify the measurement process. Having such a file; CANFI will take the ENR values for a specific frequency out of it rather than requiring a user input.

The file contains a number of frequency/ENR value pairs delimited with semicolon. Each line represents one value pair. The frequency values must be sorted ascending. Missing frequency values are interpolated. If you are using a commercial noise source simply type off the values from the calibration label. Comments are possible in a line starting with “//”.

The file itself is a simple text file which can have any file name or extension for identification, e.g. “HP346A\_DL2ALF.cal”. It can be located in any directory. Some examples are provided along with the program.

Frequency[GHz]:	Frequency in GHz, English notation (use dot as decimal point)
ENR[dB]:	ENR value in dB, English notation (use dot as decimal point)

#### Example:

```
// CanFi calibration file (c) 2014 by DL2ALF
// Syntax: Frequency[GHz]; ENR[dB]
// Example: 10.0; 15.53
```

```
0.01; 5.37
0.1; 5.47
1.0; 5.35
2.0; 5.19
3.0; 5.10
4.0; 5.08
5.0; 5.14
6.0; 5.21
7.0; 5.30
8.0; 5.42
9.0; 5.62
10.0; 5.62
11.0; 5.60
12.0; 5.60
13.0; 5.53
14.0; 5.53
15.0; 5.53
16.0; 5.55
17.0; 5.58
18.0; 5.63
```

### Tuner Gain Settings File Format

The modified rtlSDR.dll allows us to set tuner chip gain on a “per stage” basis. This helps us to use the best trade-off between sensitivity, noise, dynamic range and linearity. There is a gain settings file for each tuner type called either E4000.tun (E4000.tun.default) or R820T.tun (R820T.tun.default). Each line in the file is representing one “overall gain setting” with

individual gain settings for each available tuner stage. The available stages and gain settings per stage are listed in the comment lines of each file.

To adjust the behavior to your needs you adjust the gain settings per stages or remove lines to reduce the amount of different gains. Be careful as the result can be an invalid gain settings file! The “overall gain” is calculated and displayed by CANFI when loading the file.

#### Example

```
// CANFI tuner gain settings file (c) 2015 by DL2ALF
// Tuner type: E4000
// Tuner stages: 8
// All stages and possible gains are listed below.
// Description of the following lines: // Stage name: gain_0[1/10dB] ... gain_n[1/10dB]
//
// Stage name / Index      0      1      2      3      4      5      6      7      8      9     10     11     12
// LNA:                    -50    -25      0     25     50     75    100    125    150    175    200    250    300
// Mixer:                   40    120
// IF1:                     -30     60
// IF2:                      0     30     60     90
// IF3:                      0     30     60     90
// IF4:                      0     10     20     30
// IF5:                     30     60     90    120    150
// IF6:                     30     60     90    120    150
//
// To define individual gain settings append one line per gain below.
// Syntax: gain[0]; ...;gain[n-1] where n is the number of stages (listed above)
// Example (3 stages): xxx;xxx;xxx
// Each line must contain exactly one gain per stage, e.g. 3 gains for 3 stages.
// Each gain must be valid for the according stage.
//
// CANFI defaults (from DF9IC)
//
300;120;60;60;30;0;120;120
300;120;60;60;30;0;90;120
300;120;60;60;30;0;90;90
300;120;60;60;30;0;60;90
300;120;60;60;30;0;60;60
300;120;60;60;30;0;30;60
300;120;60;60;30;0;30;30
300;40;60;60;30;0;30;60
300;40;60;60;30;0;30;30
200;40;60;60;30;0;30;60
200;40;60;60;30;0;30;30
150;40;60;60;30;0;30;60
150;40;60;60;30;0;30;30
100;40;60;60;30;0;30;60
100;40;60;60;30;0;30;30
50;40;60;60;30;0;30;60
50;40;60;60;30;0;30;30
```

## **Calibration Log File Format**

The calibration log file contains the results of the calibration process. To simplify any further use the file format is “Comma Separated Values” (\*.CSV) with local language settings. The file can be imported easily into a spreadsheet of your choice. The file name is “Calibration.csv” and it is located in the program’s main directory.

Invalid[Bool]:	Indicates whether the calibration point is valid/invalid
Frequency[MHz]:	Frequency of calibration point in MHz
TunerGain[dB]:	Tuner gain of the DVB-T stick in dB
P_ON[dB]:	Measured power in dB with noise source ON
P_OFF[dB]:	Measured power in dB with noise source OFF
Y[Units]:	Calculated Y-Factor in units
NF[dB]:	Calculated noise figure of the measure device in dB
ENR[dB]:	ENR value of the noise source in dB

#### Example

```
Invalid[Bool];Frequency[MHz];TunerGain[dB];P_ON[dB];P_OFF[dB];Y[Units];NF[dB];ENR[dB]
False;432,200;-11,00;17,1900136613234;16,7257753782202;1,11281719667634;14,9023387490271;5,4260917891536
False;432,200;-8,00;19,3326373185518;18,6615893164597;1,16709121549766;13,1965556062684;5,4260917891536
False;432,200;-6,00;18,6063571854913;17,3972932388765;1,32101088040186;10,3608942620392;5,4260917891536
False;432,200;-3,00;21,1133235991673;19,5174534561804;1,4440659052148;8,9516174901009;5,4260917891536
False;432,200;-1,00;21,2200492824543;18,8597991622034;1,72196774424098;6,84091384131153;5,4260917891536
False;432,200;2,00;24,0016947754519;21,065432545577;1,96619335034734;5,5754513470855;5,4260917891536
False;432,200;4,00;25,2341133638452;21,3571341853694;2,44173156699411;3,837247713928;5,4260917891536
False;432,200;7,00;28,2174794233332;24,3685811832121;2,42599456744854;3,88491307909764;5,4260917891536
False;432,200;9,00;29,7277385962907;25,219394278448;2,82380323732364;2,81631196652133;5,4260917891536
False;432,200;12,00;32,5141894715265;28,2168169788984;2,6899069019029;3,14746399215023;5,4260917891536
False;432,200;16,00;35,6105278117653;30,7183643505071;3,08472423948931;2,23560562861499;5,4260917891536
False;432,200;19,00;38,8611538463855;33,6720616152125;3,30300493917237;1,80314309534279;5,4260917891536
False;432,200;22,00;41,6245674147879;37,0939102492448;2,8383484890465;2,78181336433356;5,4260917891536
False;432,200;25,00;44,4457530098658;39,5056201549887;3,11898499539836;2,16481297447022;5,4260917891536
False;432,200;28,00;47,1859072234307;42,5137644688567;2,93233966805929;2,56525709528516;5,4260917891536
False;432,200;31,00;49,8188337602026;45,137502216386;2,93855047135994;2,55132066299616;5,4260917891536
```

## Measurement Log File Format

The measurement log file contains the values taken in an ongoing measurement process. To simplify any further use the file format is “Comma Separated Values” (\*.CSV) with local language settings. The file can be imported easily into a spreadsheet of your choice. The file name is “Values.csv” and it is located in the program’s main directory.

The content is cleared each time a measurement is starting. Be sure to copy it to another location if you want to keep the values.

UTC:	Timestamp in UTC
P_ON[dB]:	Measured power in dB with noise source ON
P_OFF[dB]:	Measured power in dB with noise source OFF
Gain[dB]:	Calculated gain of the DUT in dB (if calibrated before)
NF[dB]:	Calculated noise figure of the DUT in dB
TunerGain[dB]:	Tuner gain of the DVB-T stick in dB
f[MHz]:	Frequency of the DUT in MHz
ENR[dB]:	ENR value of the noise source in dB

### Example

UTC	P_ON[db]	P_OFF[db]	Gain[dB]	NF[db]	TunerGain[db]	f[MHz]	ENR[dB]
57:25,5	52,42	37,72	19,62	0,89	25	432,2	15,45
57:25,6	52,42	37,48	19,80	0,65	25	432,2	15,45
57:25,7	52,42	37,48	19,60	0,65	25	432,2	15,45
57:25,7	52,42	37,71	19,74	0,89	25	432,2	15,45
57:25,8	52,39	37,71	19,81	0,92	25	432,2	15,45