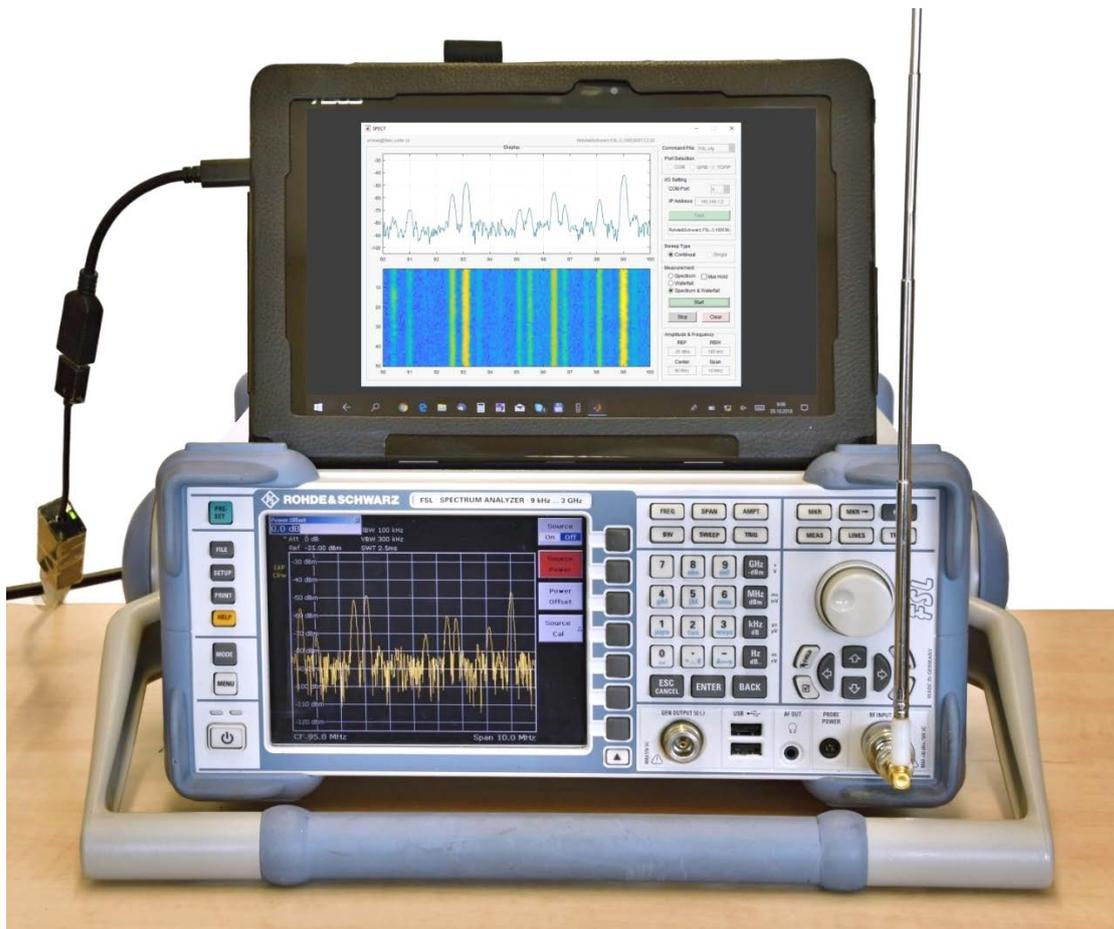


# Spectrum Viewer

Software for downloading and display of measured spectrum



**Ver. 1.3 Software Guide**

Aleš Prokeš, [prokes@feec.vutbr.cz](mailto:prokes@feec.vutbr.cz)

# Spectrum Viewer Software Guide

The Spectrum Viewer is a software for downloading of measured data from spectrum analyzers and their basic processing, saving, and displaying on a computer. It has been developed dominantly to display and store spectrum measured by older devices which are not able to save data into recent storage media and which have implemented some command-based communication protocol. However, the software can be used for any older or up to date devices equipped with either COM, GPIB or LAN port.

The application allows the basic operation such as spectrum display in classic and/or in waterfall form, saving the measured data and print screen in a few different formats, and setting a few fundamental parameters. The application was written in Matlab R2017b with installed Instrument Control Toolbox ver. 3.12 and tested on a few spectrum analyzers communicating via all three port types (Advantest R3131, HP8590, and R&S FSL). In the case of GPIB communication the Agilent USB/GPIB Interface 82357B with an Agilent driver was used.

In order to provide managing of any spectrum analyzer controlled by various command sets without need of the program modification a configuration file containing a set of device specific commands associated with the software variables has to be created (see below). The window of the Spectrum Viewer is shown in Fig. 1.

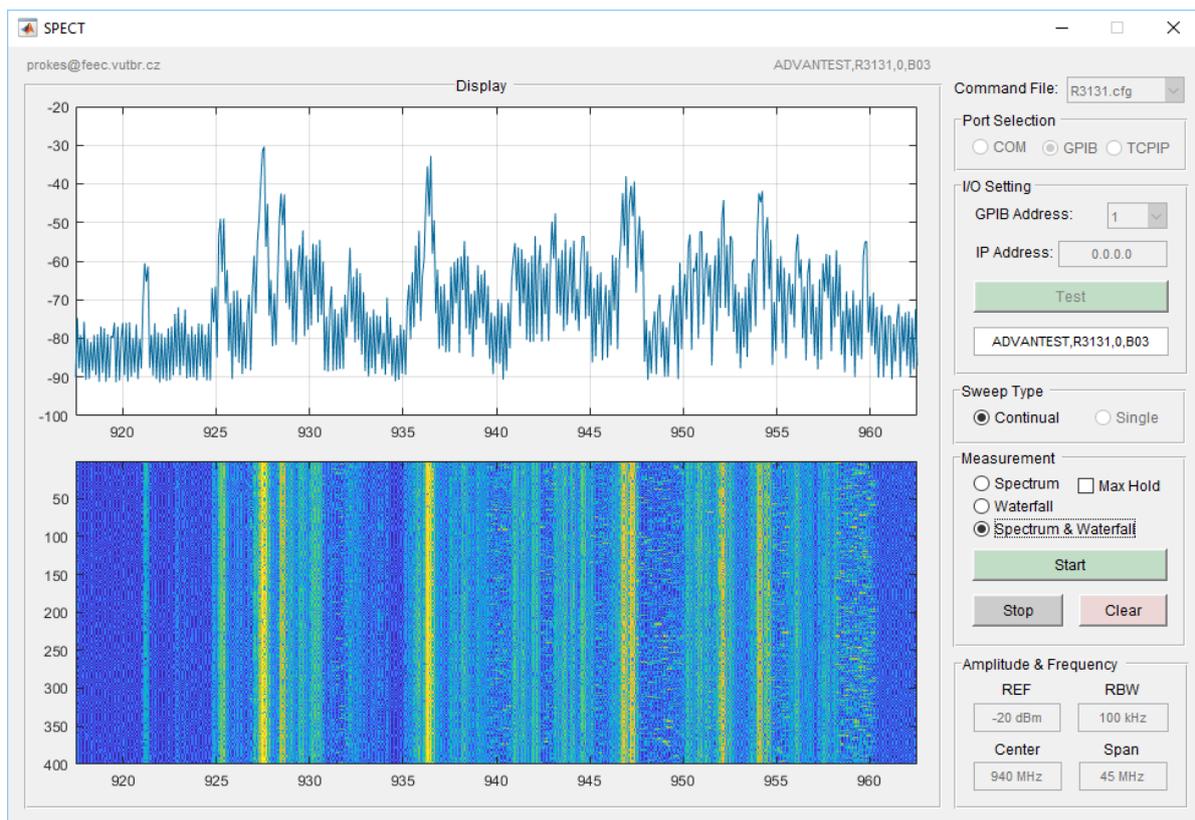


Fig. 1. Window of the Spectrum Viewer application working in Spectrum & Waterfall mode

The **Command File** pop-up menu allows to select proper configuration file (with *“cfg”* filename extension), which is necessary for configuration of the application. After selecting the configuration file, the **Port selection** block and the **I/O setting** block are preset according the configuration file content.

The communication port can be changed in the **Port Selection** block by setting an appropriate button. Basic communication options such as GPIB address, COM port, or IP address can be changed

in the **I/O Setting** block by selecting a proper value in the pop-up menu or by writing IP address in the text box. The **Test** button has to be preset before any measurement to check the connection and to establish the communication. When the device is found, its identifier appears in the text box below the button. In the other case the message *No connected device* is shown in the box and the Spectrum Viewer is subsequently switched into the demonstration mode which is indicated by the *Demo mode* message. Note that the demo mode is implemented due to the demonstration purpose and the application managing in this mode is (with some limitations) identical to the real measurement mode.

The continual or single measurement can be selected by one of the two **Sweep Type** buttons. The measurement rate in the *Continual* mode is limited by the speed of device response and by the transmission data rate. The measurement speed, of course, depends on the frequency span/resolution bandwidth ratio.

The required spectrum plot can be selected in the **Measurement** block. We can choose between classic spectrum (*Spectrum*), waterfall (*Waterfall*) or both (*Spectrum & Waterfall*). When the **Max Hold** box is checked the maximum values in the spectrum are continuously stored and displayed. The **Start** button initiates the measurement in both the *Continual* and the *Single* mode and the **Stop** button ends or breaks the measurement in the *Continual* mode. During a measurement the data obtained in each sweep are saved in the matrix of the `Freq Points` × `Rec Length` size (see any “`cfg`” file), where the `Freq Points` and `Rec Length` denote the number of frequency values at which the spectrum is measured, and the number of sweeps stored in the computer memory respectively. The **Clear** button deletes all values in the matrix, and further if the **Stop** and **Clear** buttons are pressed successively, the new measurement session (allowing the new configuration file download and communication parameter changing) can be created.

When the **Stop** button is pressed, or *Single sweep* is selected, the **Amplitude & Frequency** parameters can be changed. We can set the reference amplitude value (*REF*), resolution bandwidth (*RBW*), center frequency (*Center*), and frequency span (*Span*). After writing a new value into the box click outside this box or press Enter on keyboard. If the syntax of inserted string is wrong (e.g. there is missing space between the number and the unit), the box color will change from white to red and you have to correct it.

The **Stop** button also changes the **Start** button to **Save & Plot** button. After stopping the measurement, we can plot and save the spectrum as Matlab “`fig`” file.

If you type the command *Output Format: FIG* the Matlab Figure window will open with only classic spectrum plot as shown in Fig. 2. When entering command *Output Format: FIGs* the same window will be open but with the four plots whose meaning is obvious from the subplots headings (see Fig. 4). In the intensity plot the sweeps (whose number is given by the variable `Rec Length`) are evaluated in the floating window and the more frequent values are plotted by the color corresponding to a higher level. The 3D plot can be rotated by a mouse.

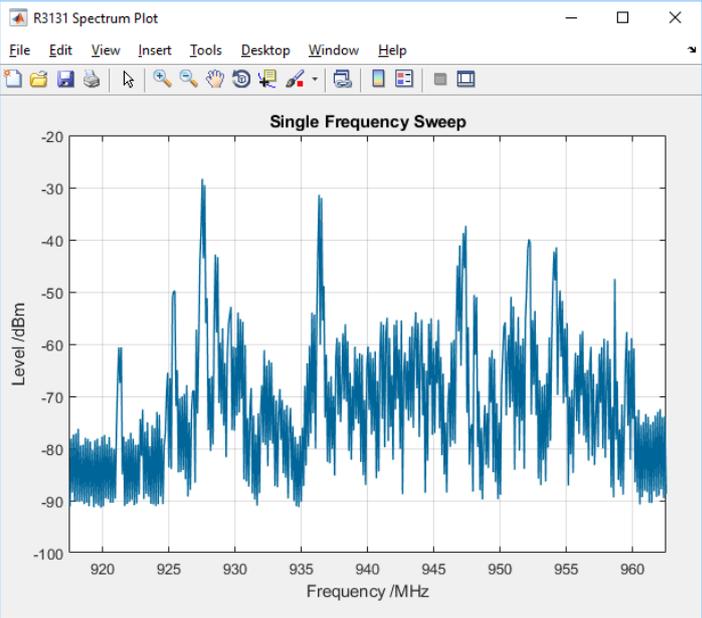


Fig. 2. Example of a FIG output

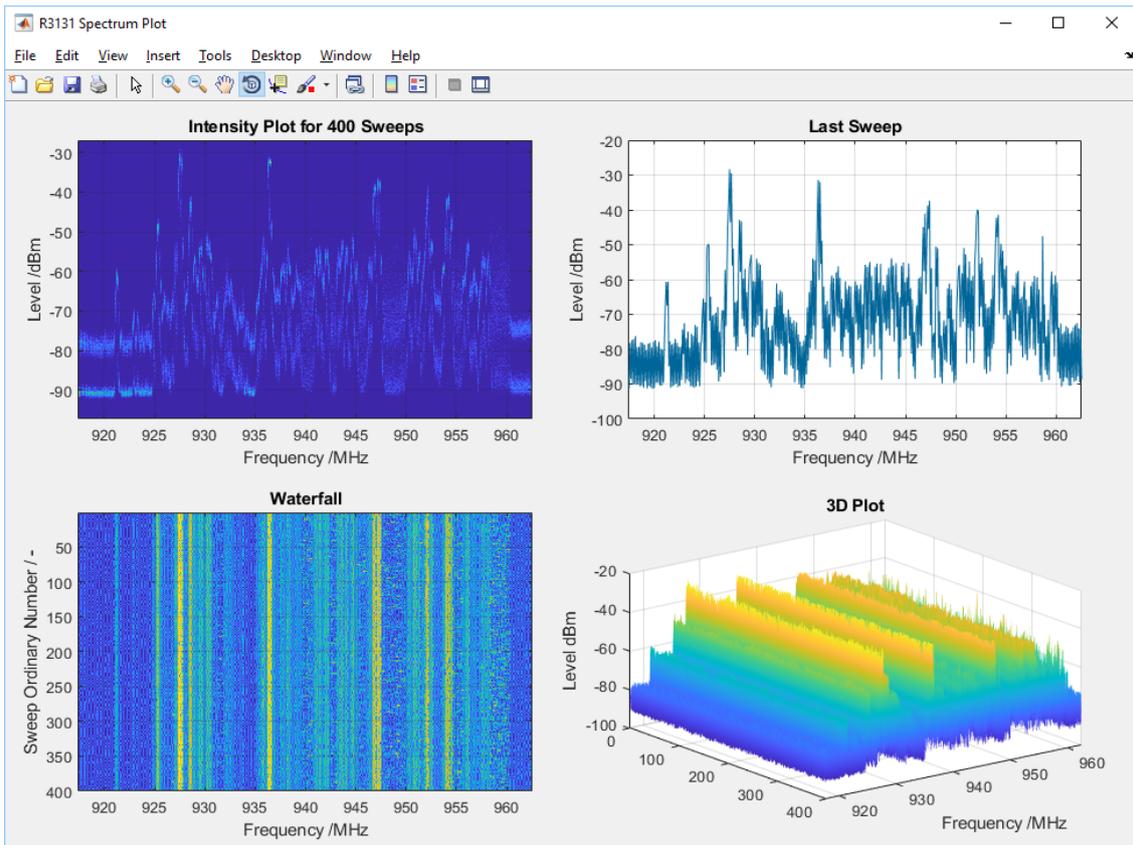


Fig. 3. Example of a FIGS output

We can also save the matrix of measured data in CSV or XLS format by adding the corresponding abbreviation after the *Output Format:* command as in the previous case. Note that the spreadsheet Excel accepts only 255 rows i.e. `Rec Length` should be lower or equal to 255 (if it is higher you have to limit the number of rows in the open file). If you open XLS file, set the surface plot and assign the first row of the file to the horizontal axis you can obtain the spectrum like in Fig. 4. Note that you can change the color scheme and the figure style.

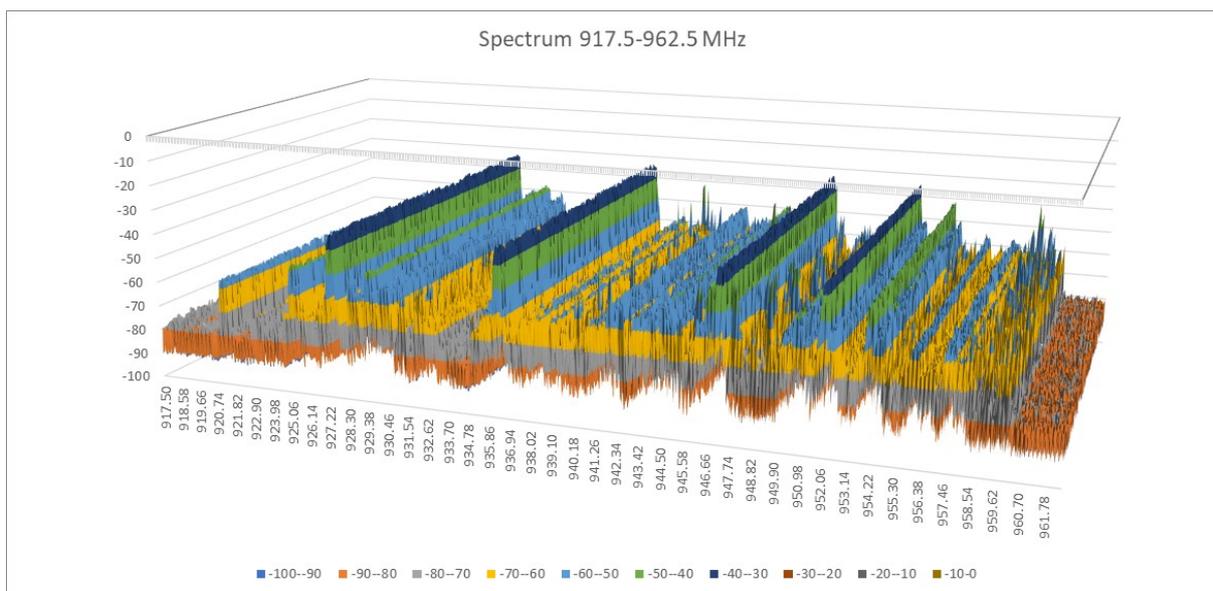


Fig. 4. Example of Excel graph obtained from an XLS output

Another output data format is a binary one. It can be chosen by writing *Output Format: BIN*. The data matrix is stored in a binary format with units and frequency points, which can be used for further processing. One of available application which offers the spectrum plot in similar formats like in the case of FIGs choice is **Spectrum Plotter**. The brief Spectrum Plotter Guide you can find bellow. It is also possible to read the binary file using Matlab script given in Apendix 1. The script is included also in downloadable SpectPlot\_files.zip file (see below).

The application also enables to save a print screen of entire Spectrum Viewer window (similar to the one in Fig. 1) in the TIFF, JPEG, or PDF format. It can be done by writing corresponding TIF, JPG, or PDF abbreviation after the *Output Format:* command (e.g. *Output Format: JPG*). After pressing the **Save & Plot** button the Spectrum Viewer prompts you to select the directory and save the file. The file extension will be automatically added.

As mentioned above, any spectrum analyzer can be managed by the Spectrum Viewer when a relevant configuration file is available. This file contains among other the device specific command sets for reading or configuring the device. You can create another configuration file by creating a copy of some existing file and by editing its content. It is recommended to use short file name derived from the device name. If the **Command File** pop-up menu is activated, the Spectrum Viewer shows all the available files with “cfg” extension, but only if they are located in the same directory as the “exe” file.

### Structure of configuration file

1. The first line contains an optional analyzer name (e.g. `Device: R3131A`)
2. The next block of parameters configures the communication between the computer and the spectrum analyzer. The full command consists in any cases from variable name, colon, and parameter(s). *To find correct device specific commands see a device programming manual!*

`Port Type:` defines the communication port (COM, GPIB or Ethernet) for the device. It can be set (as well as other parameters bellow) later in the **Port Selection** box but to prevent from many settings after the Spectrum Viewer launching, the port can be predefined in this item. The possible parameters are `COM`, `GPIB`, or `TCPIP`.

`Com ID:` in the case the GPIB port is chosen, the number predefines the GPIB address, however, if the COM port is chosen, it predefines the COM port number. If the device is connected via Ethernet cable, write any integer number here (e.g. 1).

`IP Addr:` sets the device IP address (the same address as the one stored in your analyzer). If you do not use the LAN connection, write e.g. `0.0.0.0`.

`Com Obj:` possible choice is `GPIB` or `VISA`. Some devices work with a GPIB object, while some prefers a VISA object.

`Vendor:` parameter for GPIB configuration (see manual of GPIB interface). For example, the Agilent USB/GPIB Interface 82357B requires assignment `Vendor: agilent`.

`GPIB Board No:` another parameter for GPIB configuration which corresponds with the index of the GPIB controller installed in the computer.

`Byte Order:` the two possible parameters are `bigEndian` or `littleEndian` (see the device manual). Note that when you debug a configuration file, the wrong spectrum plot is often caused by incorrect choice of this parameter.

`COM Baud Rate:` predefines the baud rate used for communication via COM port (add e.g. 9600 and set the same value in the device).

`Get Id:` asks for device identification string. In case of successful reading, it is shown in the text box bellow the **Start** button. Very frequent format is `*IDN?`.

`Init Str:` you can write here the sequence of device specific commands separated by semicolon for any initial configuration of the Analyzer. For example the `Init Str: FORM`

`REAL,32;DET APE; SWE:POIN 313` defines a numerical format of transmitted data, detector type, and number of frequency pints for the R&S FSQ analyzer.

3. The following variables starts with `Get` or `Set`. They are followed by a device specific command for reading or writing of some frequency or amplitude parameters, units or measured data. The `Get` parameter is usually terminated with a question mark.

`StartF`: start frequency of measurement (`Get StartF: FA?` causes that `FA?` is used always when the application asks the R3131 device for the start frequency, while `StartF: FA` causes that whenever you will want to set the start frequency, the application will send `FA` command to the device),

`StopF`: stop frequency of measurement,

`Span`: frequency span,

`CentrF`: central frequency,

`BW`: resolution bandwidth,

`Lev`: amplitude level,

`Unit`: command for getting or setting units.

4. The following section defines the frequency and amplitude unit (Hz, kHz dBm, V, W etc.) representation. For example, by setting `Unit MHz: MZ` causes that in case the start frequency setting will be requested (see above) the `FA 100MZ` will be generated.

5. `Get Tr A`: request for single trace data transfer.

6. While most of devices have only fixed amplitude range 10 dB/div, several analyzers have user-definable ranges (1 db/div, 2 dB/div, 5 dB/div, ...). For such devices we can use following variables:

`Get dBDiv`: request to obtain the scale dB/div used on the device screen.

`Get SwpTime`: Some analyzers cannot be querying trace data, while they are sweeping. In such case write the command for request of the sweep time. The software will wait until sweep is finished and then will ask for new trace.

`Get Scale`: list of possible codes related to scales in dB/div that can be obtained from device and used for measurement (`Get` in this case is not a request for the device).

`Log RangedB`: list of corresponding amplitude ranges in dB (obtained from the device manual) displayed on the device screen.

7. The information about the amplitude displayed on the screen is often transformed into a specific integer range defined by:

`Min Level`: minimum amplitude of displayed spectrum.

`Max Level`: maximum amplitude of displayed spectrum.

For example, each point level displayed on the R3131 device screen is expressed by an integer from 512 (`Min Level`) to 3392 (`Max Level`). The corresponding value in dB can then be calculated from known scale (e.g. in dB/Div) and instant level.

8. Measured frequency points (given by device and in some cases selected by user) and number of sweeps stored in the computer memory (given by user) are defined by variables:

`Freq Points`: informs the application how many frequency points is transmitted during a single sweep. Spectrum analyzers use either fixed or user selectable number of frequency points. For example `Init Str: SWE:POIN 313` asks the R&S FSQ device for setting 313 measurements during single sweep and `Freq Points:313` has to inform the application about it. When number of points is fixed (no selectable) only `Freq Points:` has to be defined. In the case the number of points is selectable, recommended value is between 300 and 1000.

Rec Length: number of sweeps stored in the computer memory. It defines also the number of rows in waterfall spectrum format.

9. Last group defines color scheme for spectrum plots:

Color map: defines Matlab color scheme (e.g. parula, jet, hsv, spring, copper, summer, etc.) for the waterfall spectrum plot. See Matlab manual at <https://it.mathworks.com/help/matlab/ref/colormap.html> or list of *Color scheme* in the Spectrum plotter (see below).

Plot Color: defines RGB components (from 0 to 1 each) for the classic spectrum plot (e.g. Plot Color: 0 0.4 0.6).

10. If the device does not support a command N/A expression has to be used. Empty lines are ignored.

### Spectrum Plotter software guide

The Spectrum Plotter is a simple application for plot of data stored by the Spectrum Viewer in a \*.bin file. The window of the Spectrum Plotter is shown in Fig. 5. You can select the color of plots in the *Color Scheme* pop-up menus. The upper pop-up menu defines color scheme for the *Intensity Plot*, the *Waterfall*, and the *3D Plot*. The lower pop-up menu defines color of the *Last Sweep*. The *Plot Selection* box defines which type of plot will be displayed in plot window. Only one, two, or four pictures can be plotted in single plot window. A binary file for the plot can be chosen by pressing the **Select File** button. The selected file name then appears above the button. The spectrum plot is activated by pressing the **Plot** button. An example of the *Last Sweep* and *Waterfall* plots is shown in Fig 6 where the *copper* color scheme is selected.

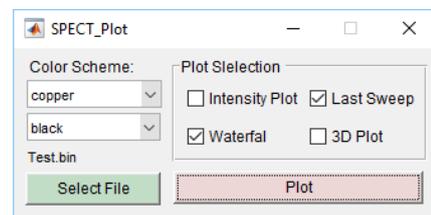


Fig. 5. Spectrum Viewer window

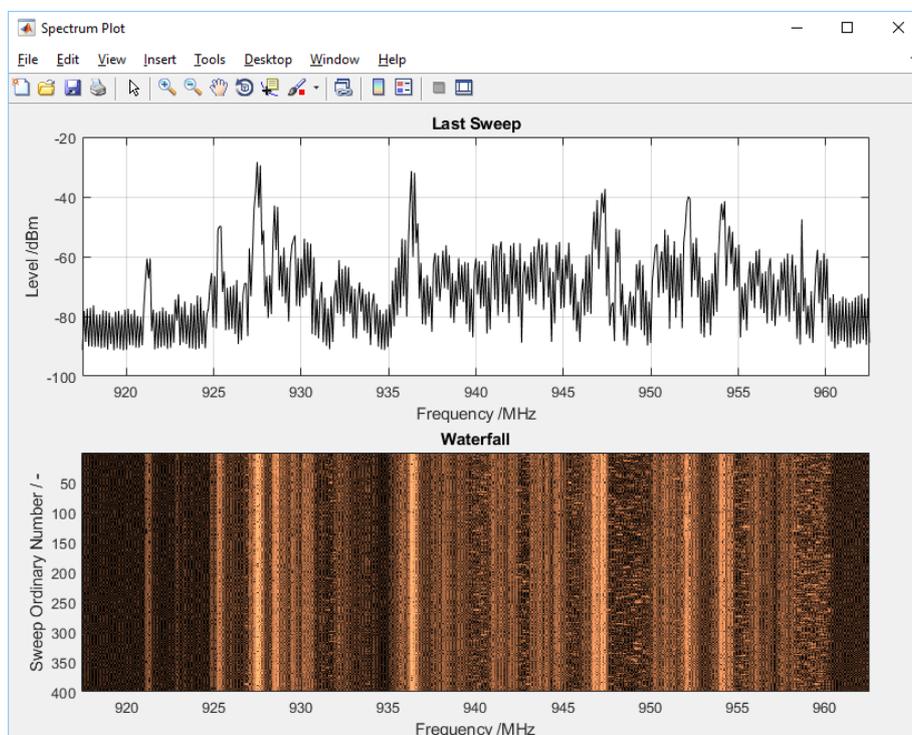


Fig. 6. Example of Spectrum Plotter output

## Installation of applications in Windows and creating connection

All the application files were generated for 64-bit PC only. All files for installation are available at [http://www.prokesfoto.cz/?page\\_id=7200](http://www.prokesfoto.cz/?page_id=7200). Depending on the software installed in your computer and on your preferences, you can install the application in two ways. Follow the instructions below:

1. If you have installed Matlab R2017 package, the **Spectrum Wiewer** can be launched using the redistribution files compressed in SpectView\_files.zip. In this case, follow the instructions in the file readme.txt). Otherwise, you can install the Spectrum Viewer application by executing the SpectView\_web.exe file compressed in the SpectView\_web.zip which runs without MATLAB but requires the MATLAB Runtime engine support. MATLAB Runtime is a standalone set of dynamic libraries that can be used to run compiled MATLAB applications on computers that do not have installed MATLAB. Finally, download examples of configuration files zipped in ConfigFiles.zip
2. Unpack the SpectView\_files.zip file into a previously created folder or install the application using the SpectView\_web.exe file. During the installation you will be prompted to select an installation directory. The Runtime engine will be downloaded automatically during the installation. Finally, unpack the ConfigFiles.zip which contains the support for Advantest R3131, HP8590L, R&S FSL, etc. devices and copy all the files into the folder containing the SPECTRUM\_VIEWER.exe.
3. The application can be run using the unzipped or installed SPECTRUM\_VIEWER.exe file (the installed file is usually located in c:\Program Files\...\SPECTRUM\_VIEWER\application\ directory). After installation it can also be executed from the Windows *Start* menu. The application launched from the *Start* menu will not probably find any configuration file. In such case go to SPECTRUM\_VIEWER shortcut location (right-click on the application icon in the *Start* menu) then go to *More* and *Open file location*. Right-click on the SPECTRUM\_VIEWER shortcut in the file location directory, select *Properties* and in the *Shortcut* folder copy the path except \SPECTRUM\_VIEWER.exe from *Target* box into *Start in* box as shown in Fig. 7.

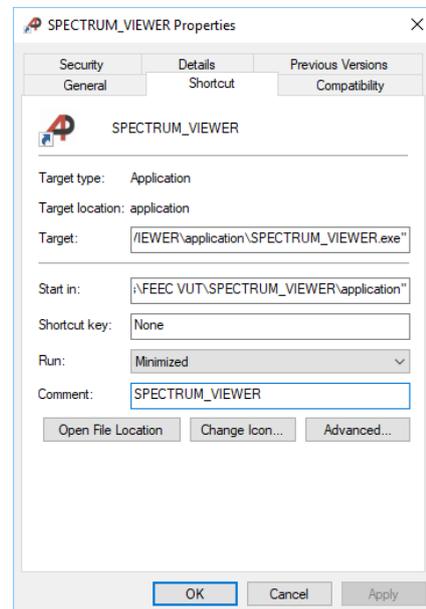


Fig. 7. Spectrum viewer properties.

4. Create the device connection with a computer:
  - When a COM port is to be used, connect the device with a RS232 cable (or USB/RS232 converter) to a computer and if necessary check the COM number in *Device Manager -> Ports (COM & LPT)*.
  - If the device should be connected via GPIB interface an appropriate hardware such as a USB/GPIB converter, has to be used. In such case install the driver, connect the device, and check the connection. For example, in the case of Agilent USB/GPIB 82357B converter install the Keysight IO Libraries Suite. The application was tested with the release 02.2016 packed in the Keysight\_Instrument\_Control\_Bundle\_2.2016.exe. It includes the driver, IO Libraries Suite 17.2, Command Expert 1.5, Connection Expert, and BenchVue 3.0 applications. After installation, the Keysight *Connection Expert -> Instruments* can be used to check the connection.
  - Moreover, you can connect device via an Ethernet cable. The proper communication setting depends on your network configuration. However, in any cases you need to know the device IP address. The most typical is direct connection computer to device performed by the rj45 patch cord. In this case set up fixed addresses both on the device and on your computer. Setting an IP address on the device depends on the TCP/IP setup implementation (see manual). For the TCP/IP protocol configuration in your computer with Win 10 open *Network and Internet settings -> Network and Sharing Center ->*

*Change Adapter Settings* -> double click on the used *Ethernet connection* -> *Properties* -> *Internet Protocol Version 4 (TCP/IPv4)*, chose *Use the following IP address*; in the box write the same IP address as stored in the device but with different last number (e.g. set the IP address in the device 192.168.1.2 and in the computer 192.168.1.1) and finally, write 255.255.255.0 in *Subnet mask* box and confirm by the OK button.

5. In all the above cases check and edit the corresponding configuration file so that all communication parameters agree with the device settings.
6. In the case of **Spectrum Plotter** installation, you have the same two choices as described in point 1. Depending on your preference download SpectPlot\_files.zip or SpectPlot\_web.zip.
7. Unpack the files into a previously created folder or install them in the same way as in point 2.

### Measurement procedure – step by step

1. Connect the device to computer via COM, GPIB or Ethernet cable.
2. Launch the Spectrum Viewer (see above).
3. Set the device related configuration file and check the **Port Selection** and **I/O Setting** boxes.
4. Press the **Test** button, wait a few seconds, and check whether the device returns its identifier.
5. If so, select the **Sweep Type** and **Measurement** choices. In other case check the connection between the device and computer and check the communication parameters stored in the device. If you edit configuration file, save it and go back to item 3.
6. Press the **Start** button. The spectrum plot will start. You can change the **Measurement** setting during the continual measurement as well.
7. If you want to change parameters in the **Amplitude & Frequency** box, press first the **Stop** button and then edit them.
8. Finally, if you want to save data or application window print-screen, write suitable command in the box bellow the **Save & Plot** button (read the instruction above) and press the button.

### Plotting procedure

1. Launch the Spectrum Plotter (see above).
2. Open a \*.bin file by the **Select File** button, select the colors and the plot type(s) and press the **Plot** button.

### Appendix 1 – Matlab script for reading \*.bin files

In this paragraph the function for reading binary files generated by the Spectrum viewer (implemented in the Spectrum Plotter) is presented. The variable `PathName` in the function call defines the path and file name (e.g. 'd:\R3131\FM1.bin'). Variable `Data` contains the measured amplitudes saved in the matrix described above. An appended last row of the `matrix` contains vector of frequency values which can be obtained for example by entering `FreqVec = Data(:, end)`. The frequency and amplitude units and the number of frequency points are stored in the variable `Units` and variable `FPoints` respectively (note that you can also obtain the number of frequency points by writing `Size(Data)`).

```
function [Data, Units, FPoints] = BinReadAll(PathName)
% -----
% BIN file Opening -----
fileID = fopen(PathName, 'r');

% Read Frequency Points -----
fseek(fileID, -4, 'eof'); % Last float position - FreqPts
FPoints = fread(fileID, 1, 'float'); % Read Frequency points count

% Read Data -----
fseek(fileID, 0, 'bof'); % Begin of file position
Data = fread(fileID, [FPoints, Inf], 'float'); % Read Spectrum data
RowCount = length(Data(FPoints, :)); % Read rows count
```

```
Data(:, RowsCount) = []; % Delete last row with units

% Read Units -----
fseek(fileID, (FPoints*(RowsCount - 1)*4), 'bof'); % Last row position
Units = char(fread(fileID, 9, 'char')); % Read units from last row
end
```