VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2018 Bc. Petr Svobodník

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV RADIOELEKTRONIKY
DEPARTMENT OF RADIO ELECTRONICS

ZPRACOVÁNÍ SIGNÁLU SDR PRO PŘENOSNOU MONITOROVACÍ STANICI
SDR SIGNAL PROCESSING FOR PORTABLE MONITORING STATION

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **Bc. Petr Svobodník**
AUTHOR

VEDOUCÍ PRÁCE — **doc. Ing. Jiří Šebesta, Ph.D.**
SUPERVISOR

BRNO 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní obor **Elektronika a sdělovací technika**
Ústav radioelektroniky

Student: Bc. Petr Svobodník ***ID:*** 154879
Ročník: 2 ***Akademický rok:*** 2017/18

NÁZEV TÉMATU:

Zpracování signálu SDR pro přenosnou monitorovací stanici

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte principy a techniky monitoringu rádiového spektra a seznamte se s metodami používanými Českým telekomunikačním úřadem. Proveďte analýzu a výběr vhodného monitorovacího přijímače na bázi SDR s open source řídicím software pro řešení přijímací a vyhodnocovací části přenosné monitorovací stanice pokrývající kmitočtový rozsah od 9 kHz do 6 GHz. Na vybraném SDR přijímači proveďte základní měření pro monitoring spektra a specifikujte požadavky na rozšíření open source řídicího programu na PC pro aplikace monitorování rádiového spektra (zobrazení 2D/3D vodopádu, definice thresholdů, měření obsazenosti pásma apod.). Seznamte se rovněž s rozhraním pro ovládání koaxiálního přepínače a rotátoru ve vysokofrekvenční jednotce monitoru. Proveďte koncepční rozbor doplňků do open source softwaru přijímače. Sestavte a odlaďte softwarové komponenty pro monitoring rádiového spektra. Proveďte praktická měření s realizovaným softwarem. Sestavte uživatelský manuál k realizovanému softwaru.

DOPORUČENÁ LITERATURA:

[1] ROHDE, U. L., WHITAKER, J. C., ZAHND, H. Communication Receivers. Principles and Design. 4/E. . New York: McGraw-Hill Education, 2017.

[2] Český telekomunikační úřad. Národní kmitočtová tabulka. [online]. 2017 [cit. 2017-09-01]. Dostupné z: http://www.spektrum.ctu.cz. Součást webové aplikace Využití rádiového spektra.

Termín zadání: 5.2.2018 ***Termín odevzdání:*** 17.5.2018

Vedoucí práce: doc. Ing. Jiří Šebesta, Ph.D.
Konzultant: Ing. Karel Holek (Český telekomunikační úřad)

prof. Ing. Tomáš Kratochvíl, Ph.D.
předseda oborové rady

ABSTRACT

Goal of this thesis is to develop portable monitoring station for radio spectrum monitoring and its controlling application for use by Czech Telecommunication Office. The station is based on Software Defined Radio (SDR) and capable of monitoring in range of 1 MHz - 6 GHz.

Developed application controls not only the SDR but also the external RF unit (including choice of receiving antenna, filter, optional amplification/attenuation and azimuth of antenna rotator). Measurement processed by computer and displayed graphically in form of spectrum diagram and waterfall diagram. Furthermore, the application will perform spectral measurement in compliance with requirements of International Telecommunication Union (ITU).

The application is also capable of recording into file and of analyzing historical data from previous measurement.

KEYWORDS

SDR, spectrum monitoring, radio spectrum, portable monitoring station, Qt

ABSTRAKT

Cílem této práce je vyvinout přenosnou monitorovací stanici a její ovládací aplikaci pro potřebu Českého Telekomunikačního Úřadu. Stanice bude realizována prostřednictvím softwarově definovaného rádia (SDR) a bude pracovat v pásmu kmitočtů 1 MHz - 6 GHz. Vyvíjená aplikace obsluhuje nejen SDR, ale i externí vysokofrekvenční jednotku (v ní ovládá volbu antény, filtru, zesilovač/atenuátor a úhel natočení antény prostřednictvím anténního rotátoru). Výsledky měření jsou zpracovávány počítačem a zobrazeny graficky ve formě grafu spektra a vodopádu. Dále aplikace provádí spektrální měření v souladu s požadavky Mezinárodní telekomunikační unie (ITU).

Aplikace umí ukládat měřená data do souboru a následně je znovu zobrazit a analyzovat.

KLÍČOVÁ SLOVA

SDR, monitoring spektra, přenosná monitorovací stanice, rádiové spektrum, Qt

SVOBODNÍK, Petr. *SDR Signal Processing for Portable Monitoring Station*. Brno, 2018, 60 p. Semestral Project. Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Radio Electronics. Advised by doc. Ing. Jiří Šebesta, Ph.D.

DECLARATION

I declare that I have written the semestral project titled "SDR Signal Processing for Portable Monitoring Station" independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the project and listed in the comprehensive bibliography at the end of the project.

As the author I furthermore declare that, with respect to the creation of this semestral project, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll., Section 2, Head VI, Part 4.

Brno ....

author's signature

ACKNOWLEDGEMENT

I'd like to thank to my supervisor, doc. Ing. Jiří Šebesta Ph.D. for supervising my work, consultation and patiency.

Brno . . . .

author's signature

CONTENTS

LIST OF FIGURES

INTRODUCTION

Electromagnetic spectrum is the whole range of frequencies of electromagnetic radiation extending from 0 Hz to theoretical infinitely high frequencies (the limit is the Planck's wavelength). In its range there are several kind of services:

- 3 Hz – 3 THz – radio spectrum
- 430 THz – 770 THz – visible light
- 30 PHz – 30 EHz – X-ray

For practical reasons, only limited part of the spectrum is usable for radio communication. This range is called radio spectrum and it covers the range of 9 kHz – 3 THz. The radio spectrum is a scarce resource and there is a need to manage its use to be efficient and equitable.

The first chapter goes through some basic notes about radio spectrum and a need for monitoring. Commercially available solutions are mentioned as well as reasons for developing our own solution. Subjects of measurement are introduced and problematic of their measurement is explained.

In the second chapter the reader is acquainted with the concept of Software Defined radio. Also, parameters and requirements for SDR are presented. Marketed SDR's are compared and outcome of the chapter is a choice of hardware used for the thesis.

In the third chapter there is presented inseparable part of this specific spectrum monitoring solution – Radio Frequency Unit. Despite this is out of scope of this thesis, there is a need for collaboration as the application developed in this thesis will be control its settings.

The fourth chapter introduces the controlling application developed in this thesis. First, there are listed requirements for the application and for computer, then IDE selection and external libraries. Single components of developed application and its execution are thoroughly described. In chapter's conclusion there is carried out demonstrational measurement using this application and results are discussed.

1 RADIO SPECTRUM AND NEED FOR ITS MONITORING

The main source of information for this chapter is Handbook on Spectrum Monitoring [1] and ITU Recommendations on Spectrum Monitoring [2][3][4][5]. The official literature defines spectrum management and measurement standard.

1.1 Spectrum management

It is overall set of administrative and technical procedures to ensure operation of radio communication devices without causing interference, to eliminate unauthorized use and thereby maximizing spectrum utilization efficiency. Some tasks of the management are listed below:

- Spectrum planning and allocation
- Rules, regulation and associated standards
- Inspection of radio installations
- Rules, regulations and associated standards
- Spectrum monitoring
- Licensing, assignment and billing

With rapid growth of using radio spectrum, law covering its usage are becoming more and more important. At national level this law guarantees accessible spectrum for both civilians and government with respect to Radio Regulations (RR) [6]. For promoting these tasks the spectrum manager is ensured by entitling law enforcement.

At country's level there are national radio spectrum management agencies. For the Czech Republic it is Czech Telecommunication Office (CTU). In world's scope this is responsibility of International Telecommunication Union (ITU) – specialized agency of United Nations with headquarters in Geneva, Switzerland. This agency coordinates shared global use of the radio spectrum and assists in development of worldwide technical standards.

1.2 Spectrum monitoring

Radio spectrum monitoring is very closely linked to spectrum management as it is the way to check on proper use of spectrum. In fact it is an inherent part of spectrum management as it allows to perform other activities mentioned in previous section.

In reality some devices might become an unintended source of radiation resulting in interference of other services and of course it serves as a detection of an illegal transmission. Furthermore, radio waves are not confined by man made borders.

Because of this, there is a need for international cooperation in border areas which is described in Article 16 of RR [6].

Spectrum monitoring allows to ensure uninterrupted services to customers paying for license. Another important task of monitoring is to verify coverage of service (for example TV and radio broadcasting) over the specified area.

1.3 Portable monitoring station

With ongoing development the working frequencies are increasing and so is the bandwidth. But, with higher frequency the propagation distance decreases and the measurements need to be performed more locally. At this point fixed measuring stations are not sufficient as it is needed to measure closer to their source.

1.4 Commercial monitoring solution

As the need for the monitoring is present for longer period of time, there are in the market complete solutions to this. One of them is from German company Rohde&Schwarz

Disadvantages

The main disadvantage of ready products are their closeness in both software and hardware way leading to small scalability in the future. If there was for example need for increasing frequency range above possibilities of given system, the whole receiver would have to be replaced with new one whereas in case of modular open-source solution it is possible to extend system's abilities by partial replacement and change of controlling software.

1.4.1 R&S Argus

It is integrated software solution for both management and monitoring which means that there is shared database allowing to perform requirements seamlessly. Measurement and analysis complies with ITU specifications and software has several guides for measurement which allow less experienced users to perform demanding tasks.

1.4.2 R&S PR100

Portable spectrum monitor working in range 9 kHz – 7.5 GHz equipped with 6,5"display. Capable of battery-supplied operation for approx. 4 hours. Measured data stored at built-in SD card. Operations like monitoring emissions, detecting interference

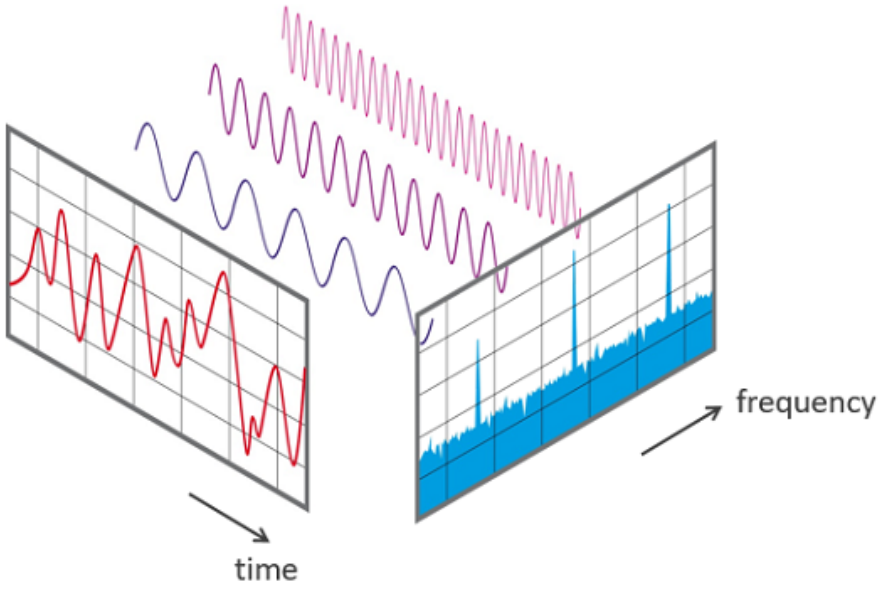


Fig. 1.1: Mix of signals in time and frequency domain [7]

or locating miniature transmitters are supported.

1.5 Methodology of measurement

1.5.1 FFT

Fast Fourier Transform is an effective algorithm for transforming signal from time to frequency domain. By factorizing the discrete Fourier transform into products of sparse factors, the complexity of calculation is reduced from $O(n^2)$ to $O(n \log n)$ where n is data size [8].

The number of frequency lines in the FFT spectrum is equal to the number of samples in the time record and growing number of samples for FFT calculation results in better frequency resolution according to eq. 1.1. But, with greater count of samples short signal might get drowned. Using less samples makes result more sensitive to signals which are short in time. The outcome is that the number of samples for FFT analysis is every time a compromise between time and frequency resolution.

$$\Delta f = \frac{f_s}{n} \tag{1.1}$$

1.5.2 Window functions

Fourier transform assumes whole number of signal periods in each calculation, which is in real world untrue as this condition would be fulfilled by only few signals. Not respecting this assumption leads to to spectrum leakage. The effects of spectral leakage can be lessened by reducing the discontinuities at the ends of the signal measurement time before FFT processing and hence avoiding time discontinuities. There are several types of window but in general it is valid that no window is perfect for every application as it is a compromise between frequency selectivity and amplitude accuracy.

Windows comparison

Parameters for specifying windows are resolution, dynamic range, sensitivity (ability to reveal sinusoids in presence of noise), usable bandwidth and side lobe roll-off rate. Shape of the window is defined by formula. Outside of the window the value is zero and those samples are not taken in consider at all.

- Rectangular window – the simplest window, samples are either taken in consider in their original amplitude or not at all. Excellent resolution characteristics for sinusoids of comparable strength, but it is a poor choice for sinusoids of disparate amplitudes. Used for transient signals that lie completely within the time record.

$$w(n) = 1 \tag{1.2}$$

- Hann window – also know as raised cosine window or Hanning window. End points of the window are touching zero level resulting in removing time discontinuities.

$$w(n) = 0.5 * \sin^2(\frac{\pi n}{N-1}) \tag{1.3}$$

- Hamming window – optimized to minimize the maximum of nearest side lobe, giving it a height of about one-fifth that of the Hann window

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1}) \tag{1.4}$$

A good way to start is in most cases Han window as it has good frequency resolution and reduced spectral leakage. For most of applications these three windows should be sufficient and CTU didn't specified their requirements any closer.

1.5.3 Bandwidth occupation

Usage or channel occupancy data indicates how much of the time a particular frequency or band has had a signal present during a specified threshold. Measurements on a single frequency can be combined to show how usage varies during

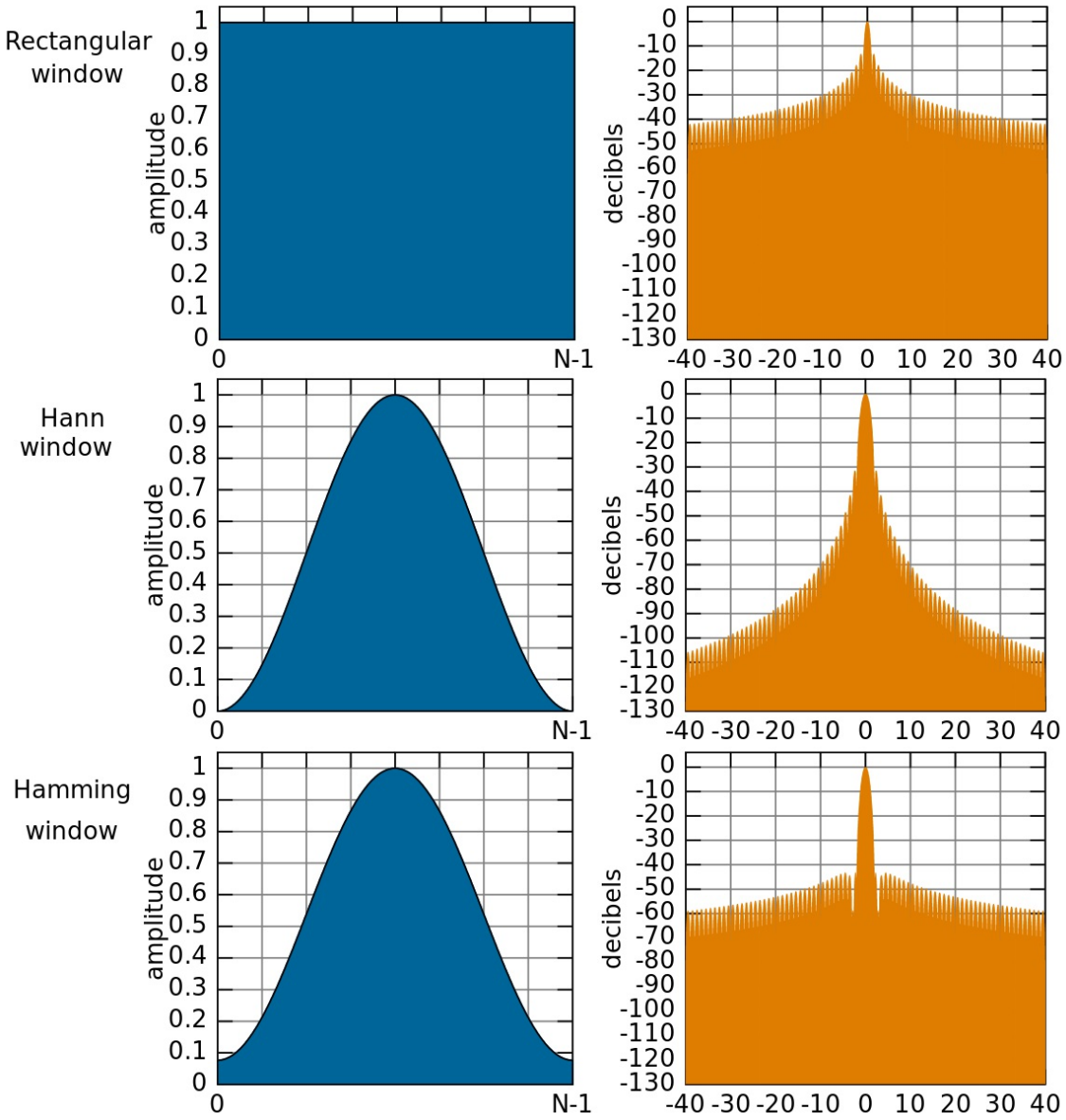


Fig. 1.2: Weighing window comparison – left column displays weighing factors of the window in time domain, the right one shows its frequency response

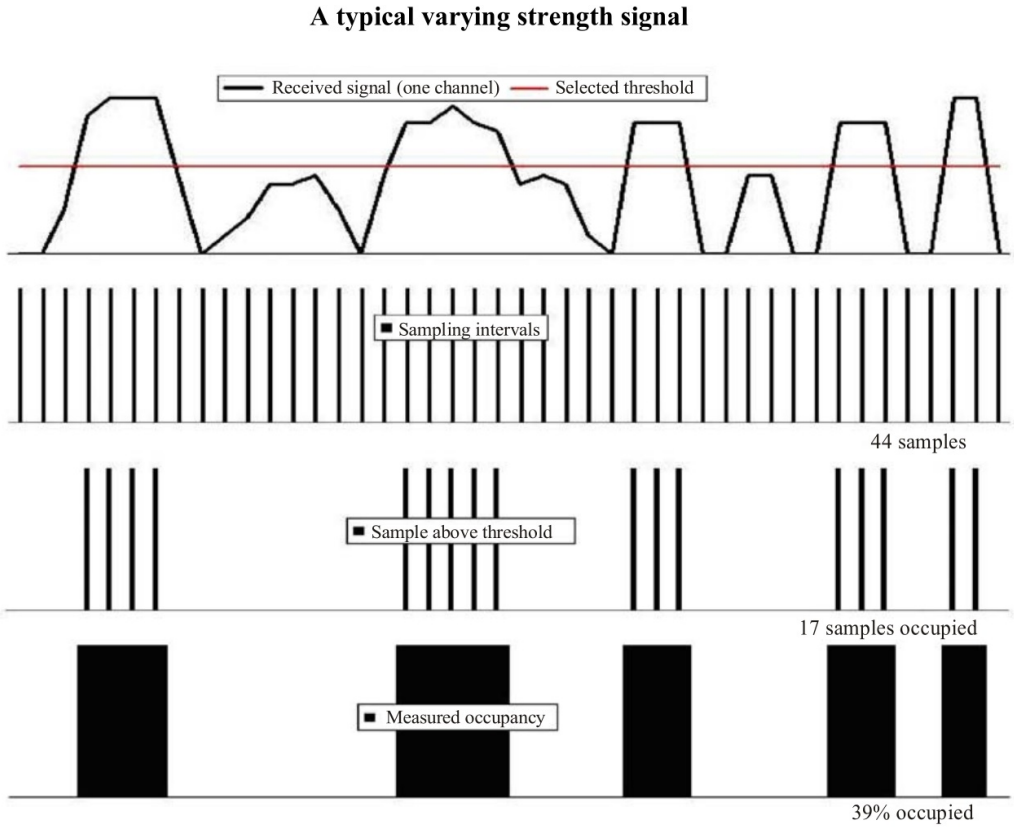


Fig. 1.3: Measurement of frequency occupancy for single channel/frequency, picture taken from [1]

longer period of time (a day or more). Integration time of a single measurement usually lasts 5 – 15 minutes.

Channel occupancy and band congestion information is a valuable tool for several spectrum management functions. This information can be used to identify vacant channels in a band, and can be used to prohibit adding more assignments to heavily used channels. Such data can also be used to prompt an investigation, either when signals are present on unassigned channels, according to the frequency management records, or when no usage is seen on frequencies with assignments. Changes with time-of-occupancy statistics, for the same band in the same geographical area, can reveal trends. Finally, this type of information can be used to help anticipate and plan for the allocation of additional bands when existing bands become too crowded.

Principle of measurement

A signal varying in strength is shown in Fig. 1.3 with a selected threshold as indicated. In this example 17 of the 44 sampling periods were found to be occupied leading to a 39 % occupancy.

1.5.4 Length of measurement

In order to get valuable results the length of measurement will be in order of days (up to months). More information about recommended quantity of measurement repetition for required level of probability confidence in [9].

2 THE CONCEPT OF SDR

This chapter introduce some brief summarization of history of Software Defined Radio (SDR), available software and hardware and SDR for this thesis will be selected.

2.1 Software radio

The original idea of SDR is Software Radio (SR) formulated by Joseph Mitola in 1992 [10]. The goal of this architecture is to replace analogue components with digital ones and to allow tuning in wide frequency range and reaching different requirements (modulation, bandwidth and so) solely by uploading different software configuration. This could be reached by AD (respectively DA) conversion right at antenna output.

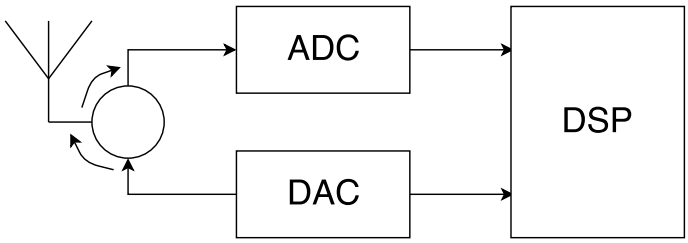


Fig. 2.1: Block scheme of ideal SR – schematic presumes having ADC with anti-aliasing filter and DAC with appropriate power amplifier.

Without an RF front end the entire band must be digitized [11]. Following Nyquist's criterion, the signal must be sampled with sampling frequency $f_s \geq 2 \cdot f_{max}$ (e.g., $2 \cdot 6 = 12\,\text{GHz}$). Using 12 bits sampling the data rate for real-time data processing would represent $12 \cdot 10^9 \cdot 12 = 18\,\text{GB/s}$, requiring great computational power. That might be manageable, but at very high costs.

Another problems are AD converters, because nowadays there is no technology capable of sampling at such high frequency. Plus, with higher resolution Heisenberg uncertainty principle might be violated [12].

2.2 Software Defined Radio

It is simplified architecture where in comparison with SR the signal is converted to Intermediate Frequency (IF) or baseband by analogue mixing with signal of Local Oscillator (LO). Then it is filtered, optionally amplified with LNA and sampled. These samples are then processed with on-board FPGA (Field Programmable Gate

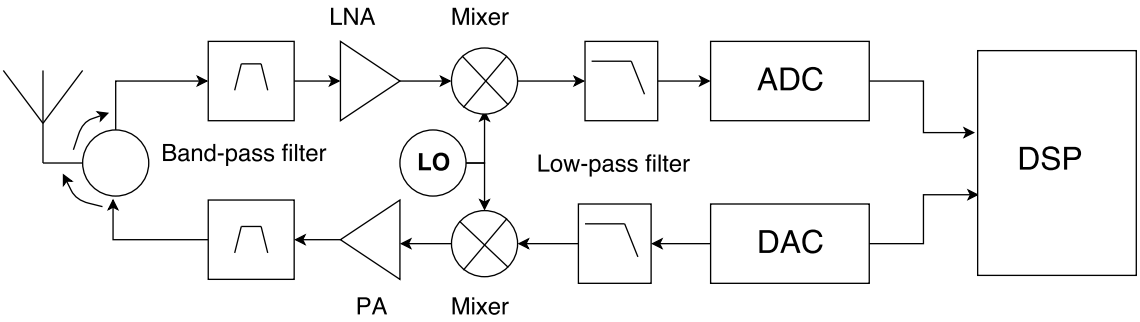


Fig. 2.2: Block scheme of general SDR. Upper branch represents signal reception, lower represents transmission

Array), ASIC (Application Specified Integrated Circuits), DSP (Digital Signal Processor) or in computer. Some radios with their parameters are outlined in 2.4.

2.2.1 Requirements for SDR

- Frequency range – is given by requirements of CTU – 9 kHz – 6 GHz
- Sample rate – shows the bandwidth that can be observed
- ADC Resolution – number of bits which represent each of samples
- Oscillator accuracy – deviation between set and actual frequency of LO
- Price – expressed in American dollars
- Open-source – for customizing and building own application

Frequency Range

Most of SDR's are not capable of working from 0 Hz due to presence of low-pass filters in their design. The upper limit is typically in order of low gigahertz.

Sample Rate

As SDR's are using conversion of signal to IF or to baseband, there is not anymore the need for sampling at such high frequency, but still it is valid that the more the better. Sampling rate influences the bandwidth that can be processed.

Dynamic Range

The captured spectrum contains the signal of interest and a multitude of other signals. Interfering signals can be much stronger than the signal of interest. The digitizer must have sufficient dynamic range to process both the strong and the

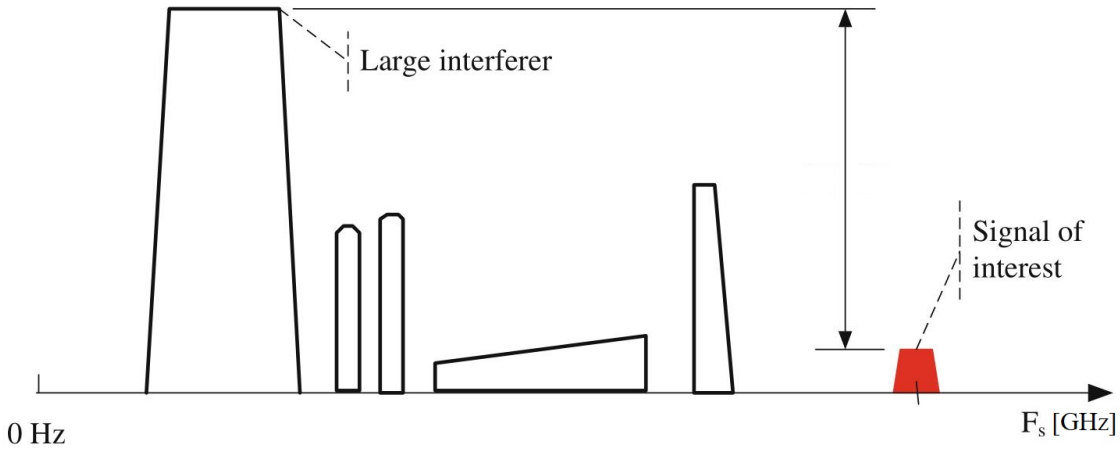


Fig. 2.3: AD Conversion in presence of strong interfering signal

weak signals as shown in Fig. 2.3. Dynamic range is proportional to Analog to digital conversion (ADC) resolution.

Linearity of ADC

The ADC must be very linear. Nonlinearity causes intermodulation between all the signals in the digitized band. Even a high order intermodulation component of a strong signal can swamp a much weaker signal. The most critical is the third order intermodulation nonlinear term.

2.2.2 Advantages of SDR

Versatility of use

An SDR can seamlessly communicate with multiple incompatible radios or act as a bridge between them. Interoperability was a primary reason for the US military's interest in, and funding of, SDR for the past 30 years.

Reducing obsolescence

As the function of SDR is configured by software, it can be easily upgrade to support the latest standards. For example, if mobile operators used SDR's, they would be able to use the latest generation of network without wasting the old hardware and save lot of money that way [13].

2.2.3 Cons of SDR

So far it looks like there are only good reasons to use SDR, but every technology has some drawbacks.

Cost

For volume production it is important to reach the lowest possible price. SDR is powerful tool, but for many application there is cheaper (and still functioning) solution.

Power Consumption

As SDR is universal and complex (contains LO, filters, AD/DA converters and so on), it will never be as efficient as single purpose devices.

Complexity

It takes more time and effort to develop and thoroughly test the software of the radio.

2.3 Controlling software

Software for controlling the radio is an essential part of the work with SDR. There are several criterion for choice. Firstly, there are proprietary solutions which work only with products of their producer and then, there are universal ones.

In the other category there are several open programs for (mostly amateur) work with SDR. Some of them are for complex work with radio and raw IQ signal data and then there is the other category which works with graphic interface. Let's have look at them to get some basic overview of the functions.

2.3.1 SDR# - SDR Sharp

Easy to use application which main controlling is done in graphical interface. The first large window displays FFT diagram of received signal and the other one plots the waterfall diagram. Program comes from producers of AirSpy radio 2.4.4 but it comes with some of widely spread radios pre-configured so the user can simply plug the radio in the computer and within few moments he's able to work.

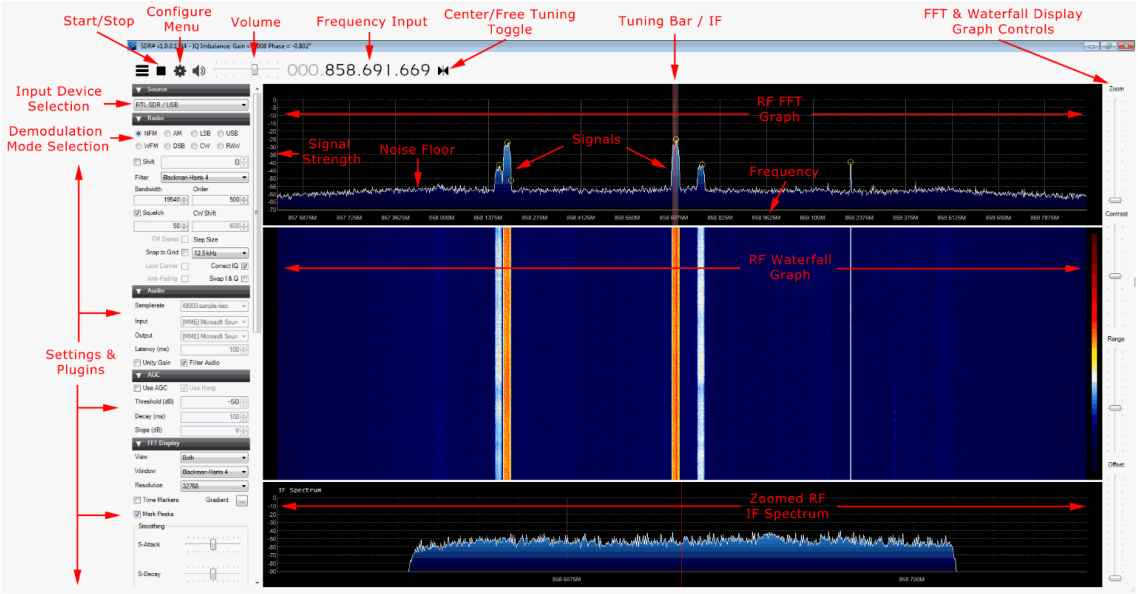


Fig. 2.4: SDR# User interface (source: www.rtl-sdr.com/tag/sdrsharp/)

2.3.2 SDR-Console

SDR-Radio.com is a Windows console program for SDR receivers and transceivers. Designed for the commercial, government, amateur radio and short-wave listener communities.

2.3.3 GNU Radio Companion - GNU RC

Multi-platform software, operations with signal are represented by graphic blocks which are placed on the canvas and interconnected with lines which represent the cables (with support of data types). This program allows very complex work with signals at low level, radio and supports transmission as well. Version for OS Windows exists, but the program is primarily for OS Linux.

The main difference between GNU RC and other programs is the workflow. User needs to choose the right sampling rates at each stage of project.

2.4 Overview of the SDR's

Below, there is an introduction of some of the radios available on the market with a focus on open-sourced ones.

2.4.1 HackRF One

- **Frequency range** – 1 MHz to 6 GHz

Fig. 2.5: The HackRF One radio

- **Sample rate** – 20 MSPS
- **ADC Resolution** – 8 bits
- **Oscillator accuracy** – +/-20 ppm
- **Max RX power** – -5 dBm
- **Dynamic range** – 48 dB
- **Price** – $299 USD
- **Open-source** – Yes

HackRF One represents an open-source hardware radio. On board there is ARM Cortex M4 microcontroller and a CPLD. It is receiver and transmitter in single device, but it only works in half duplex mode which means it is not possible to transmit and receive at the same time. As the reception is corner stone of this application, this feature does not represent any problem. It is equipped with standard SMA connector for connecting antenna. It also contains SMA clock input and output for synchronization. The connection with computer is made via USB 2.0 interface which is also a power for the radio. The HackRF One ships with an injection molded plastic enclosure.

2.4.2 LimeSDR

- **Frequency range** – 100 kHz to 3,8 GHz
- **Sample rate** – 61,44 MHz

- **ADC Resolution** – 12 bits
- **Oscillator accuracy** – +/-4 ppm
- **Price** – $299 USD
- **Open-source** – Yes

Another radio which is capable of transmission with power 0 - 20 dBm (depends on frequency). The interface for connection with computer is USB 3.0 (also PCIe version). It has an FPGA integrated – Altera Cyclone IV. This radio is outcome of crowdfunding campaign.

2.4.3 RTL-SDR – RTL2832U V3

- **Frequency range** – 24 – 1766 MHz
- **Sample rate** – 2,4 MHz
- **ADC Resolution** – 8 bits
- **Oscillator accuracy** – <1 ppm
- **Price** – $25 USD
- **Open-source** – No

Originally DVB-T tuner with customized driver to optimize characteristics. While it was never designed to be used as a general purpose SDR in the first place, its performance is still surprisingly good. Low price makes it ideal hardware for beginners with SDR. Host interface is USB.

2.4.4 Airspy R2

- **Frequency range** – 24 – 1800 MHz
- **Sample rate** – 10 MHz (9 MHz usable)
- **ADC Resolution** – 12
- **Oscillator accuracy** – +/-0,5 ppm
- **Price** – $169 USD
- **Open-source** – Yes

An open-source SDR with good dynamic range, capable of sampling 10 MHz bandwidth in spectrum from 24 MHz to 1,8 GHz. Compatible with all the standard softwares SDR#, SDR-Radio, HDSDR, GQRX and GNU Radio. The lower frequency limit can go down to DC when using their own up converter (will be mentioned in Ch. 2.5.2).

2.4.5 Signal Hound BB60C

- **Frequency range** – 9 kHz – 6 GHz
- **Sample rate** – 27 MHz

- **ADC Resolution** – 14 bits
- **Oscillator accuracy** – +/-1 ppm
- **Price** – $2879 USD
- **Open-source** – NO

Marketed as a real-time spectrum analyzer and RF recorder with sweep speed of 24 GHz/sec and dynamic range from –158 dBm to +10 dBm. The interface is USB 3.0 and it comes with own software.

2.4.6 SDRPlay RSP2

- **Frequency range** – 1 kHz – 2 GHz
- **Sample rate** – 10 MHz
- **ADC Resolution** – 12 bits
- **Oscillator accuracy** – +/-0,5 ppm
- **Price** – $169 USD
- **Open-source** – NO

Also exists in version marked as "pro"which has metal enclosure. The rest of parameters is the same. SDRplay uses its own SDRuno software which is able to monitor and record RF power.

2.4.7 USRP B200

- **Frequency range** – 70 kHz – 6 GHz
- **Sample rate** – 56 MHz
- **ADC Resolution** – 12 bits
- **Oscillator accuracy** – N/A
- **Price** – $675 USD
- **Open-source** – NO

The USRP comes with metal enclosure which prevents interference to affect the performance of the radio. Regarding source openness the answer is not that clear. The radio uses company's own USRP Hardware Driver (UHD) which works with both GNU RC and Matlab. Thanks to that and to its integrated Xilinx Spartan 6 FPGA on board the radio is popular in academic environment and many papers written are based on results of this radio.

2.5 Dealing with very wide frequency band

None of the radios is fulfilling the requirements for the frequency range. There are to possible attitudes to solving this:

- using SDR's with different ranges
- transforming part of the spectrum

both of these are explained in following sub-chapters.

2.5.1 Two SDR's covering range together

By smart choosing of two radios we can cover the whole requested range. Controlling application would then decide about the radio performing the measurement, but still there would be needed some manual action of connecting antenna to the proper radio or using some switch, which would insert some loss of the signal before processing. Also, there is a need for some overlapping to ensure that measurement for given bandwidth will proceed only at one of these radios.

For example the combination of HackRF One (1 MHz – 6 GHz) and SDRPlay RSP (1 kHz – 2 GHz) would ensure more than sufficient overlapping of the ranges.

Another problem that shows up is that radios will definitely differ from each other considering sensitivity, dynamic parameters and others, which might result in inequality of results when performing measurement with the same parameters at both radios.

2.5.2 Upconversion

The other way of reaching ability to measure in the lowest band is using upconverter. Such device mixes original signal with it's own local oscillator and that transforms the original spectrum to frequency $f_{new} = f_{LO} + f_{original}$. This signal can be further processed by one radio receiver. However, upconverters insert some conversion losses which need to be taken in account in power calculations for getting correct RF power levels.

Below, there are listed possible upconverters. Both of them require 5V DC supply (also possible from battery for mobility and less noise).

Ham it Up v1.3

This upconverter is based on the double balanced mixer architecture implemented by the ADE-1 mixer chip from Minicircuits under open source license. The local built-in oscillator generates signal of 125 MHz (<25 ppm) which is in sufficient distance from FM radio broadcast. As part of the design there is a bypass switch which reroutes the journey of the signal and unpopulated space on circuit board was used to integrate noise source circuit. Product comes without any enclosure. Insertion loss is 10 dB.

Fig. 2.6: Upconverter SpyVerter R2

The range declared by manufacturer is from 100 kHz, but it is capable of working at lower frequencies at the cost of higher distortion and higher conversion loss.

SpyVerter R2

Another upconverter available as a whole solution is SpyVerter. It uses architecture of H-mode mixer design. The major advantage over ADE-1 (used in Ham it Up) should be better IIP3 (the third intercept) performance which means that strong signals will not cause overloading issues in the SpyVerter, meaning less noise and spurious images. Frequency of LO is 120 MHz (0,5 ppm) and creators focused on reaching low phase noise. Metal enclosure is default part of distribution.

- Conversion loss: typically 5,2 dB
- RF input: 1 kHz – 60 MHz
- Max. RF power: 10 dBm

2.6 Hardware selection

From the devices listed above some set of devices needs to be chosen for further realization. Looking only at parameters, the USRP B200 seems to be the best choice as it is covering almost whole required frequency range with very high sampling rate and great resolution of samples. However, this SDR will not be used for the thesis after hearing bad references from thesis' supervisor.

Instead, taking in account his recommendation, the SDR used for this project will be the HackRF One. With its frequency range and source-openness the HackRF One should be suitable for the thesis. Measurement at frequency below 1 MHz will

be realized with usage of upconverter SpyVerter R2 as it is capable of working at required lower frequency.

3 RADIO FREQUENCY UNIT

This chapter introduces shortly RF unit which manages Antenna movement, processes signal reception and band filtration for measuring. However, this is not topic for this thesis as there is another part of this project [14].

3.1 Antenna Rotator

For controlling antenna's azimuth remotely there is used antenna rotator ARS-USB. This is an universal rotor interface, so any rotator – motor with a potentiometer for a voltage feedback – is supported [15]. That potentiometer serves as position sensor and from its resistance the actual azimuth of antenna is determined.

Fig. 3.1: The ARS-USB antenna rotator

This unit uses for communication with computer USB bus, which emulates an RS-232 serial link whilst connected. This rotator is distributed with custom software, but commands for controlling are published, so using custom application is not a problem. In modern OS drivers for USB Serial port are not required.

Instructions for the unit are pretty simple. There is a move to an azimuth command, move left/right command and request for actual position. For reading the resistance there is 10-bit ADC and this 10-bit value is returned to the computer as a reply to that request. This value needs to be converted to angle by simple mathematics.

ARS-USB supports both azimuth and elevation directing, however CTU's antenna holder is at this moment only capable of azimuth movement. Therefore, remote polarization control is not in scope of this thesis.

Parameters of the connection

- Baud rate: 9600 bps
- Data bits: 8
- Parity: None
- Stop bits: 1
- Flow Control: None

3.2 RF Unit

The RF unit consist of set of antennas, altogether covering the whole requested frequency band, respective set of filters and optional amplifier or attenuator. Desired signal path is built using coaxial switches. These are controlled by microcontroller which is connected via RS-232 to computer.

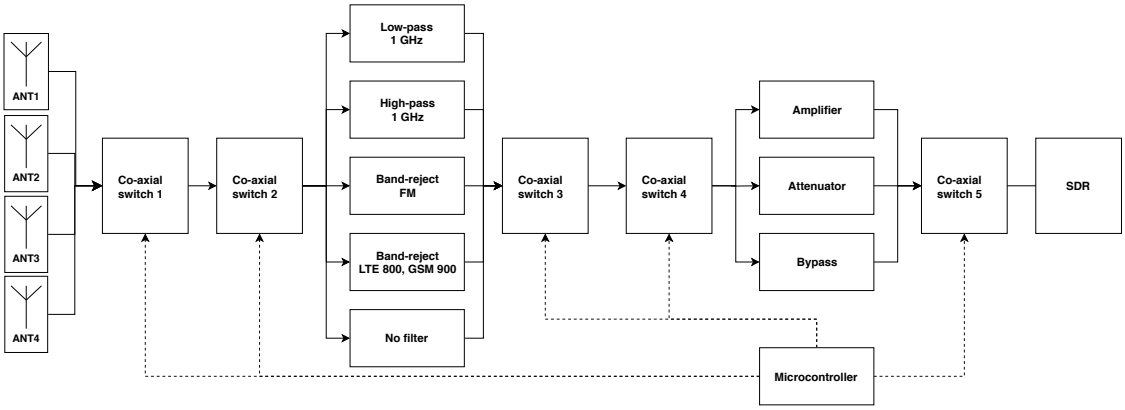


Fig. 3.2: Block scheme of RF unit

3.2.1 Antennas

Four antenna from Rohde&Schwarz were selected:

- HE010E - rod antenna working in range of 8.3 kHz to 100 MHz
- HK309 - extremely wide bandwidth 20 MHz to 1.3 GHz
- HL040E - log-periodic broadband antenna, range from 400 MHz to 6 GHz
- HL223 - virtually frequency-independent radiation pattern, 200 MHz to 1.3 GHz

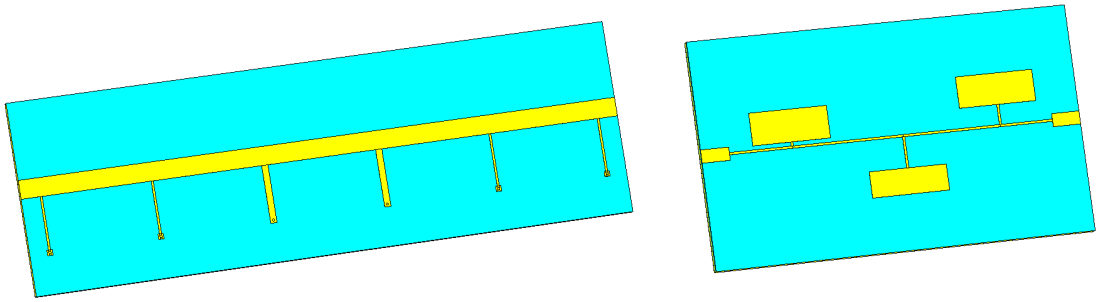


Fig. 3.3: High pass and Low pass filters

3.2.2 Filters

For suppressing particular bands there are four filters. Only one filter can be used at time.

- Low Pass 1 GHz – Custom microstrip filter of 7^{th} order
- High Pass 1 GHz – Optimum Distributed HP Filter of 6^{th} order
- Band Reject FM (87.5 - 108 MHz) – made of classic LC components
- Band Reject LTE 800, GSM 900 (791 – 862 MHz, 876 – 960 MHz) – made of LC components as well

3.2.3 RF power

At this stage the RF power is adjusted to suit the needs of measuring. There are basically three options:

- Amplifier – low-noise wideband Mini Circuits ZX60-V63+ amplifier with 21 dB gain is used
- Attenuator – DC – 6 GHz attenuator API Weinschel 3406-55. Value may be edited with 1 dB step, maximal attenuation 55 dB.
- Bypass – direct interconnection of coaxial switches 4 and 5

3.2.4 Coaxial switches

Basically there are two possible variants of switching. Either mechanical or electronic. In this case, the first variant is preferred and Teledyne CCR-38S14(6)0-T was chosen. This switch comes with even count of ports – selected variants have 4 or 6 of them. Insertion loss is stable and may be considered to be frequency independent.

4 APPLICATION

In this chapter the developed application and its components will be described. For each window, there will be a separate section. Following section are so comprehensive they may also serve as a manual of operation for operators.

4.1 Abilities of the application

Requirements from CTU as the ordering party are:

- Frequency range – 9 kHz – 6 GHz (as mentioned before)
- Measurement bandwidth – is given by HackRF One's sampling rate which is 2, 4, 8, 10 or 20 MHz.
- Length of measurement – may vary from minutes to months. Possibility of indefinite length of with data storage as the only limitation.
- Displaying charts – application will contain several plots with data measured:
  1. FFT diagram – displays a single value of FFT process at given time. On the vertical axis there frequency range and the vertical one there is RF power at referenced frequency.
  2. Waterfall diagram – also called spectrogram. Taking in results and assigning its power levels a color. Individual results are shown as line and then they are shifted down like in a shift register.
  3. Zero span diagram – a single frequency component's power plotted in dependency on time
- Bandwidth occupancy – this has already been explained in Chap. 1.5.3
- RF power level measurement – SDR doesn't guarantee any sensitivity and RF Unit has impact as well. This part of application will have to be tested and calibrated.
- Saving results – all the data measured will be stored for future analysis and displayed in diagrams after measurement finishes. For this purpose results of FFT calculation are saved.
- Controlling the RF unit – Parameters of the measurement need to be sent to the unit (more info in Chapter 3).

4.2 Requirements for host computer

When sampling with the highest available rate – 20 MHz at 8 bits per sample in both I a Q plane, the data rate flowing to a computer is $20 \cdot 10^6 \cdot 2 \cdot 8 = 40$ MB/s. The USB 2.0 bus, which is user for transferring data, is at this rate getting close

to its limit, but still it is sufficient. For this reason, it is not possible to run this application in a virtual machine as they don't allow full-speed access to the bus.

Not every processor is capable of processing this data flow and for smooth work, the CPU should reach at least 5000 points at CPU Mark. Further, the computer should be equipped with at least 4 GB of RAM.

Regarding Operation system used, the choice is 64-bit Ubuntu Linux as the libHackRF API library is designed for usage with UNIX systems. More about this library will be in Chap.4.4.1.

4.3 Development interface

At first there were thoughts of writing an application in GNU RC, but later on Qt was chosen for this purpose. Qt is C++, JavaScript and QML development framework for creating GUI applications for most of the usual platforms (Windows and Linux included). Qt is capable of producing applications for all major OS's, but sticking to Linux is recommended.

4.4 Additional Libraries

Many routines have already been programmed and it is pointless to develop them again. This project contains only open source solutions.

4.4.1 libHackRF API

For communication with HackRF One, there is a library from authors of the project. Published on GitHub [16] makes this library open-sourced as the rest of the radio. This library is written in C and includes all the commands for setting parameters of the radio and controlling its functions. Also, it allows user to configure chips (like MAX2837 Baseband IC, Si5351C clock generator IC or RFFC5071 mixer IC) which are part of the board or to update HackRF's firmware.

Even though the programming language C is platform independent, this library is tailored for UNIX systems as it uses POSIX threads. It is possible to make this library working in OS Windows, but every computer would have to be configured manually for using it.

4.4.2 FFTW

A free open-source C subroutine library for computing the discrete Fourier transform. This library manages to compute multi-dimensional transformation with ar-

bitrary length of input data. But, the highest efficiency is reached with the length of 2^N samples. Creators of the library declare it to be one of the fastest solutions for computing FFT. This library is portable to any platform as source codes are provided. More at [17].

4.4.3 QCustomPlot

Free C++ Qt library (distributed under GPL license) for displaying data in plots offering high performance for real-time visualization applications [18]. Many types of graphs are included and this library is used for all the plots in this program.

The general way of using this library is to create UI widget and then to promote it to class QCustomPlot (QCP). Then a graph is assigned to this widget.

QCPGraph

Simple graph for plotting scatter diagrams. Many parameters like type of line, data points, graph background are user-definable. This type of plot is used for displaying of frequency spectrum all over the program.

QCPColorMap

A 2D map which visualizes a third data dimension by using a color gradient. Even though this type of graph is initially meant to plot static data, in this thesis the QCPColorMap will be used as Waterfall Diagram (WD) and as band occupancy diagram.

Other types of graphs

The QCP library also contains other subclasses

- QCPCurve - similar to QCPGraph, but this graph may contain loops
- QCPBars - represents data series as bars
- QCPErrorBars - superposes lines representing error on existing graphs

4.5 Main Window

The main window is displayed after launching the application. All other dialogues are opened from this window by clicking a button. In the Fig. 4.1 there is an example of the main window with descriptions of its elements.

In the upper part of the window there is a spin box for selection of RX frequency and right next to it, the user can switch frequency units (Hz, kHz, MHz, GHz).

Frequency shown in mentioned spin box is converted to new units subsequently. The default choice of units is MHz.

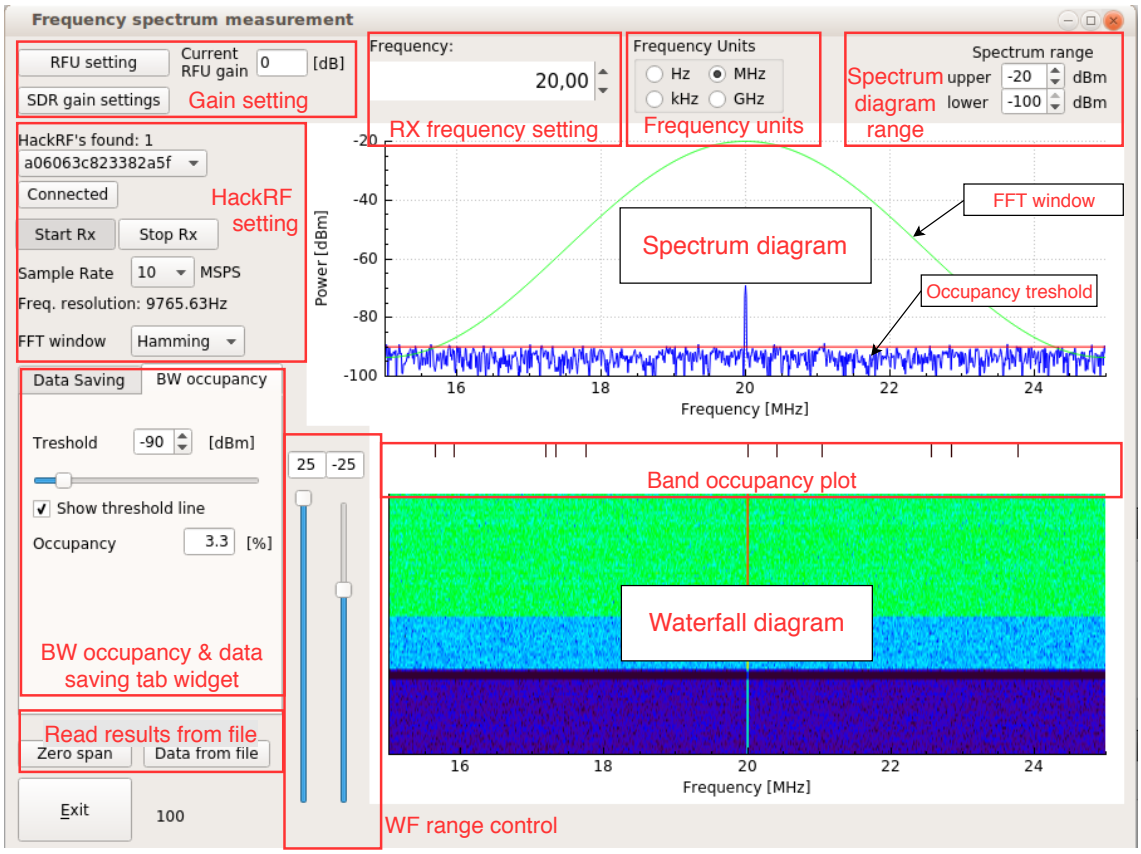


Fig. 4.1: Main window of the application

4.5.1 Spectrum Diagram

Majority of the area is taken by spectrum chart for plotting live results of FFT calculation. The length of FFT is 1024 samples as CTU wishes and the refresh rate is 500ms.

Another usage of this diagram is for representation of FFT filter shape values (more in 4.5.5) and the threshold of band occupancy measurement (more in 4.5.6.

When HackRF receives data user can only change Y axis range using two spin boxes located above spectrum diagram. Lower and upper range do not overlap, so the situation of reverting graph is prevented. X axis range is set automatically based on set RX frequency and width of the band (span). But, after stopping of reception user can change the X axis range using mouse drag for shifting both ranges and scrolling mouse wheel above the area of the graph for zooming in the frequency domain. These range changes are synchronized with WD so they move and zoom together.

4.5.2 Waterfall Diagram

A QCPColorMap widget for showing the history of FFT calculation. The dimension of the canvas is 100x1024 data points which means there are 100 samples shown at once with the newest pushed to the top of the diagram, making the effect that the diagram is shifting down the screen. As mentioned in the previous section, when the reception is stopped, the user can manually zoom X axis. For Y axis there is not data range since there are only 100 samples

Regarding the ranges, there is third scale in WD which represents mapping colors to values of RF power. Left from the diagram there are two sliders for manipulating color assignment

1. QCPColorMap's data upper range changes the look of all the diagram
2. Adding slider's value to displayed data, so only new samples are affected

The scope of these sliders is only within WD which means that they only change the way how data are presented, not their actual values, so archiving data is not influenced.

4.5.3 RX Frequency Settings

In the top of the window, there is located a QDoubleSpinBox for configuring RX frequency. There are several ways of editing widget's value:

- Writing the value in on keyboard
- Mouse scrolling over the widget
- Clicking the up and down arrow buttons
- While the widget is focused, it reacts to keyboard Up and Down, respectively Page Up and Page down keys

Changes of value are accepted automatically, no confirmation is required. As the name of the widget tells, not only integer values are accepted. However, the precision is limited to two decimals for keeping the widget neat. When entering more decimal positions, the number is truncated. When the new value is accepted, this change is projected in both the diagram's X axis range, still respecting sample rate set. On the other hand, this range in WD applies only to samples recorded after this change, so the user needs to keep this in mind. This behavior is standard for programs mentioned in Chap. 2.3, hence there is no warning about this fact.

Ranges of values always respect possibilities of the radio, so user cannot set frequency which violates this rule. If the sample rate is set to 10 MHz, the spin box has lower limit 6 MHz and so on, which results in preventing of mirror images around zero frequency.

Next to this spinbox there is a set of radio buttons which define frequency units. User may choose Hz, kHz, MHz and GHz. The value in the spin box which sets RX

frequency is changed in a respective way (here the precision of 2 decimal number might be limit when using GHz). Also labels in both plots change right away.

4.5.4 HackRF Connection and Controls

When the app is launching, the list of connected HackRF radios is created and their serial numbers are listed in the combo box. If there is no radio found, a message is shown to the user, see Fig. 4.2 b). If the user clicks the Connect button, an error dialog is shown (fig. c)). This dialog is also shown in the case when there is already another instance of the app running, since the access to the HackRF is exclusive.

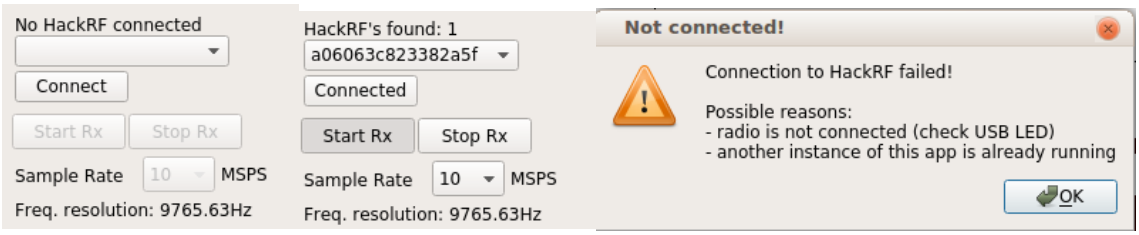


Fig. 4.2: a) situation after launching the app with no HackRF connected (or detected)
b) When successfully started reception, the Start RX button is checked
c) Error message in case radio was detected, but the connection failed. Troubleshoot advice are provided.

When the connection is a success, buttons for control of reception are enabled and Connect button changes its text to 'Connected'. Once the user presses the Start RX, the reception and data processing and plotting begin. To make reception more obvious, the button is set to be checkable.

Allowed values of sample rate are listed in a combo box. Selection doesn't need to be confirmed and change is applied immediately. New sample rate also influences X axis range of both plots which are updated and become evident with next graph replot, which occurs every 500 ms when reception is enabled.

With sample rate is tied frequency resolution of FFT analysis. This value is shown right under the combo box for selecting sample rate. Default bandwidth is 10 MHz.

4.5.5 FFT Filter

The purpose of FFT filter has already been explained in 1.5.2. All three of window shapes are present in the application and user can change them freely by only selecting desired type in the combo box and as was mentioned before, the shape of

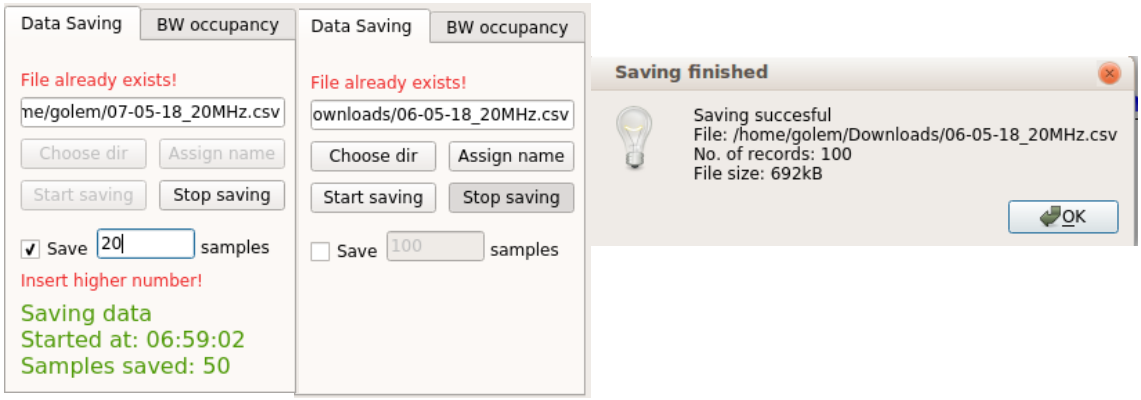


Fig. 4.3: Data saving options, the dialog from last picture shows after finishing

FFT filter is shown in spectrum diagram. Samples are bounded to the secondary y axis of the plot pane with fixed range, so scaling of spectrum diagram does not affect the look of this plot.

4.5.6 Bandwidth Occupancy

For reasons of lucidity, BW occupancy has been placed into tab widget together with data saving. User defines threshold either by entering the value in the spin box or by pulling the slider and once the value is crossed, the usage percentage is calculated. Samples crossing the threshold are also shown in a bar between Spectrum and Waterfall diagram (see 4.1) which is actually again QCPColorMap with dimensions 1x1024 pixels. The frequency at which RF power crossed the threshold is then presented as a dark pixel (respectively stripe) at the respective position of the graph. Measurement of occupancy is done over the bandwidth which is equal to sample rate of the radio, therefore this graph, neither calculated percentage do not change when the user is zooming or changing range in the spectrum or waterfall diagram.

In order to make threshold decision clearer, there is an option of displaying line representing threshold in spectrum diagram. User just needs to check the button which is located below the threshold slider.

4.5.7 Data Saving

Saving measured data is in the same tab widget as bandwidth occupancy. In the upper part there is a line edit widget containing the filepath which is by default leading to Home directory. User can change this folder by clicking Choose Dir button which opens a standard dialog for selection of directory. The new path is then set into this line edit widget and user can either define file name manually (suffix .csv is checked and pertinently added) or there is Assign name button which generates

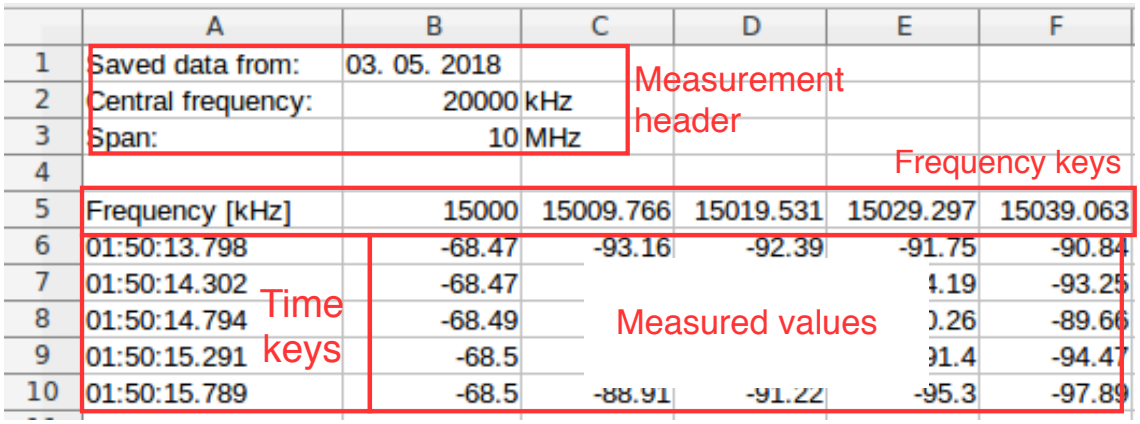


Fig. 4.4: An example of saved file format

name based on the formula dd-MM-yy_freq+MHz.csv, ie. 18-04-18_142MHz.csv for measurement from April 18th, 2018 at frequency 142 MHz. Even when the name is assigned through the button, user can still change it.

In case there already exists, there is shown the warning (see middle screenshot in Fig. 4.3) informing the user about this fact. In case user presses the Start saving button, a warning about overwriting existing file is shown and the user can still abort this process.

Now everything is ready for saving. Clicking on the Start saving button disables the possibility of changing RX frequency, sample rate and file name till Stop saving button is pressed. Also, a text label information about the start of saving and count of samples saved is shown. In case the application is closed during saving, the file is closed correctly and no data is lost.

Regarding the format of the data saving see Fig. 4.4. First, there is a header containing the date of measurement, central RX frequency and bandwidth. Then, on row 5 there are listed frequency keys of FFT calculation in separate columns and from row 6 on there are data alone (in [dBm]) with time of that specific sample in the first column. The saved values of FFT take in consideration RF unit gain and HackRF's built-in VGA and LNA amplifiers. Data is saved in comma separated values data type (.csv) where the actual separator is tabulator - '\t'.

A single file may contain data from only one calendar day. The first reason is reasonable handling with data files (considering loading in latter applications) and the other reason is data security in case of any failure. If data saving is running over night, at midnight new file with the automatically generated name is created.

There are 1024 values saved in the file in every sampling which occurs after approx. 500 ms, therefore the size of the file is negligible, even though data is saved as plain text. One hour of measurement generates around 7200 samples which results in around 50 MB of data. Since the maximal length of record into one file is 24 hours,

the size of the file should not exceed 1,2 GB.

Length of measurement is indefinite in default. In case the user wants saving to be closed automatically, the program offers an option to do so by saving defined count of samples. After checking respecting checkbox the line edit is set active and waits for the value entered. The number needs to be higher than the count of samples which are already saved and application is checking for that. If this condition is not met, a warning message shows up and saving keeps going.

This planned stopping of saving may be used also for duration estimation as there are two samples per second. Therefore time of data saving is approximately half the count of samples (in seconds).

When saving successfully finishes, a confirmation dialog is shown. Once again there is printed full file path, count of samples recorded and size of the file in kB. Then the file path field is cleared and in case of need for saving again, user needs to set new file name the same way like he did before.

4.5.8 Gain settings

This area contains two buttons opening new dialog (more will be said later) and also a line edit widget in read-only mode which shows 0 dB in default. As the label says, this is the total gain of RF Unit. After configuring the unit, the value will periodically update (when reception is active). This gain *does* have impact on Spectrum diagram and on values which are saved in file.

4.6 RF Unit Settings

This window is opened by clicking the "RFU Setting"button in Main Window. There are three parts of the window

- Frequency line edit widget - RX frequency of HackRF in Main Window. In case of tuning the new value is refreshed by clicking the "Reload frequency"button
- Antenna rotator controller - for antenna directing
- RF Unit controller - for choosing antenna, filters and changing signal's power level

After launch of dialog are both controllers disabled except for combo box with list of available serial ports and Connect button in each of them. This list of ports is generated by operating system at window's opening.

4.6.1 Antenna Rotator controller

Antenna rotator was already discussed in 3.1, this section deals with how controllers of the Antenna rotator were designed.

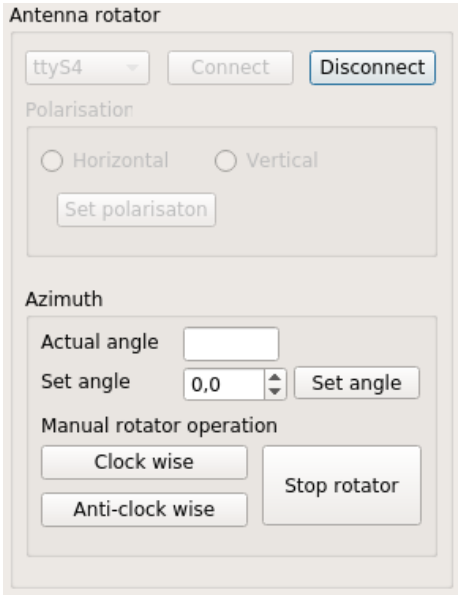


Fig. 4.5: Preview of Antenna rotator controller while communication is opened

The upper part of the controller's section is prepared for changing polarization. This is just dummy interface as polarization change is not required.

In the lower part of the screenshot there is a section dedicated to azimuth control. The line edit widget in the first row shows actual position of antenna read out from the unit. On the second line there is a spin box in case user knows the angle together with the set button. Accepted value of angle is -359–+359 degrees.

There is more buttons bellow. Those serve for manual rotation which lasts as long as the user holds the button. In other words, as the user releases the mouse button, antenna movement stops. Current value of antenna's azimuth is every 200 ms requested and displayed in line edit widget. In case of any failure, there is also the Stop button present. It works for both ways of movement initiation.

This rotator unit is capable of controlling antenna in both azimuth and elevation (or for polarization), but as the antenna stand is not ready yet, only azimuth movement is subject of this thesis. However, the interface is already prepared for changing polarization remotely and user interface for doing so is disabled.

4.6.2 RF Unit controller

This subwindow calculates parameters of RF unit (described in Chap. 3)and sets its coaxial switches via serial port. Individual sections will be discussed in following

lines

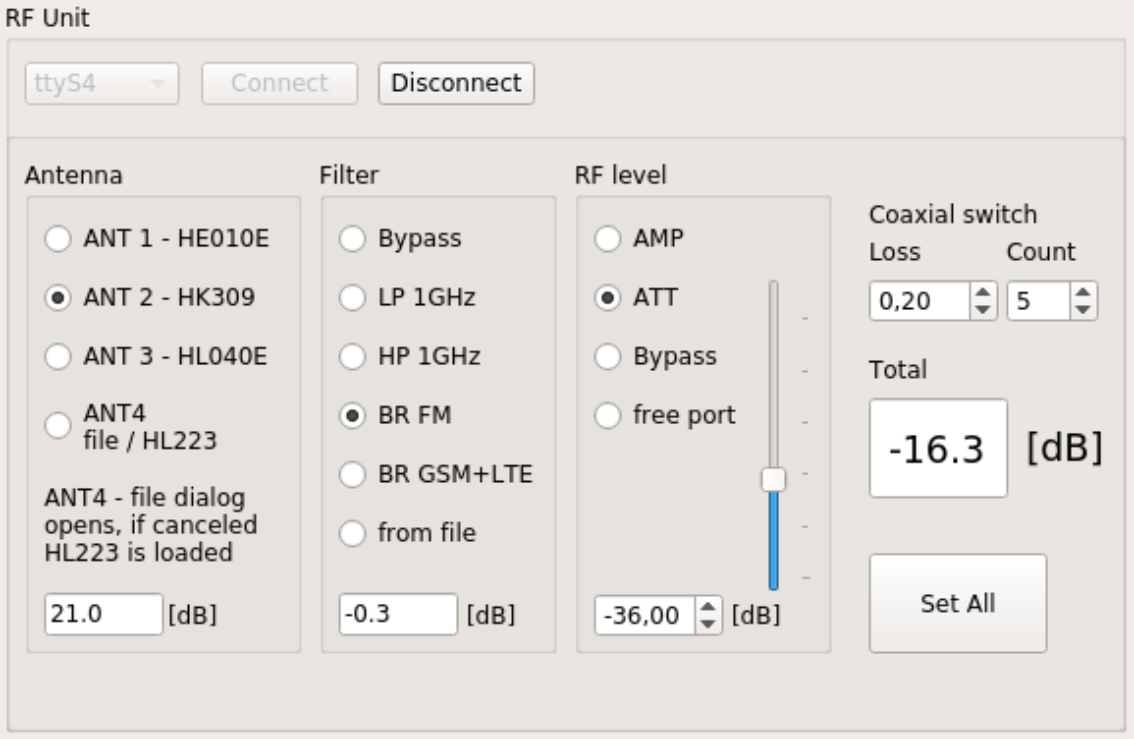


Fig. 4.6: In default there is no radio button checked

Antenna

Four antennas will be used for covering the whole frequency range. Their antenna factors were measured with certain frequency step. If actual frequency is somewhere between these steps, an approximation of AF's value is calculated using linear interpolation from known values and displayed in the line edit widget. Antennas' parameters are defined for only certain band, and the user is responsible for choosing the right one.

For keeping some versatility of this interface, the ANT4 port may be used with anoother antenna, which parameters are loaded by clicking the last radio button. File selection dialog will pop up and data will be assigned. If the user cancels this dialog, values of antenna HL223 are loaded.

Considering the format of data, there may be used any plain text format with two columns of data separated by a comma (;). The first column bears frequency in Hertz and the other one gain in decibels.

Filter

In this step, user has 6 options to select as mentioned in 3. Those are:

- Low pass 1 GHz
- High pass 1 GHz
- Band Reject FM Band
- Band Reject GSM+LTE
- Bypass
- Free port - load parameters from the the file

Parameters of filters are again defined in files and their actual gain at HackRF's RX frequency is interpolated the same way like antenna factor.

RF power level

In this case, possibilities are:

- Amplifier - gain is frequency dependent and its value is interpolated from measured values
- Attenuator - value is user-controlled with step 1dB in range 0-55dB, slider for setting value is present. Default value is -55 dB to prevent damages
- Bypass
- Free port - load parameter from file

4.6.3 Coaxial switches

Power loss in the coaxial switch is considered to be constant all over the range of measurement. User can define both loss and count of these switches. Default values are 0,2 dB and 5 switches

Total gain&Set button

The sum of all previous gain (respectively losses) appears in large digits. By clicking the "Set All"button command to the microcontroller is sent. If all mentioned options aren't defined, the user is notified and no changes proceed.

4.6.4 Connection via serial port

Both antenna rotator and unit communicate via serial link. During this window's launch system searches for available ports and lists them into combo box (there are two boxes, but both display the same list actually). There's no other way of finding the right port than testing it out since devices connected have no identifiers.

After selecting the port and pressing connect button system attempts to open up the communication. In the picture below there is shown dialog window signalizing the result of the connecting attempt. In the unsuccessful case, there is an error generated by the operating system. The failure from screenshot occurred when I

was trying to open already opened port. The whole list of possible errors is listed in [19].

Fig. 4.7: Example of success and fail during opening serial port

For having the possibility to gain access to the port in Linux user needs to promote his account to obtain higher privileges. This procedure is well described in [20].

To close the port user can use the Disconnect button or just to close the window, as the port close command is part of this window's destructor.

4.7 SDR Gain Setting

This window is launched with the second button in the upper left corner of Main Window and sets additional parameters of HackRF One radio. These are not that important and probably won't be used as there will be connected Antenna unit with definitely better filters and amplifier.

Both sliders are respecting allowed step size of particular IC's and after reopening this window the last value of gains is loaded (talking of the same instance of the program). Another parameter configurable from this window is 3.3 V antenna power. Its state corresponds with the checkbox. The last parameter is an integrated filter

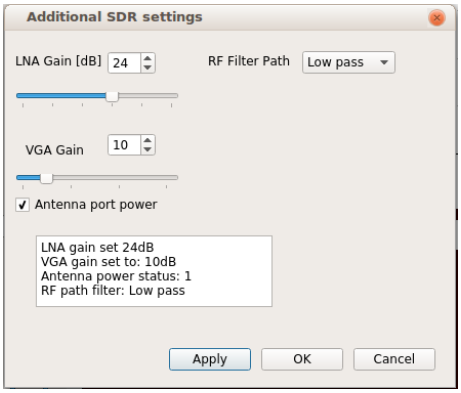


Fig. 4.8: Preview of SDR Gain Setting window

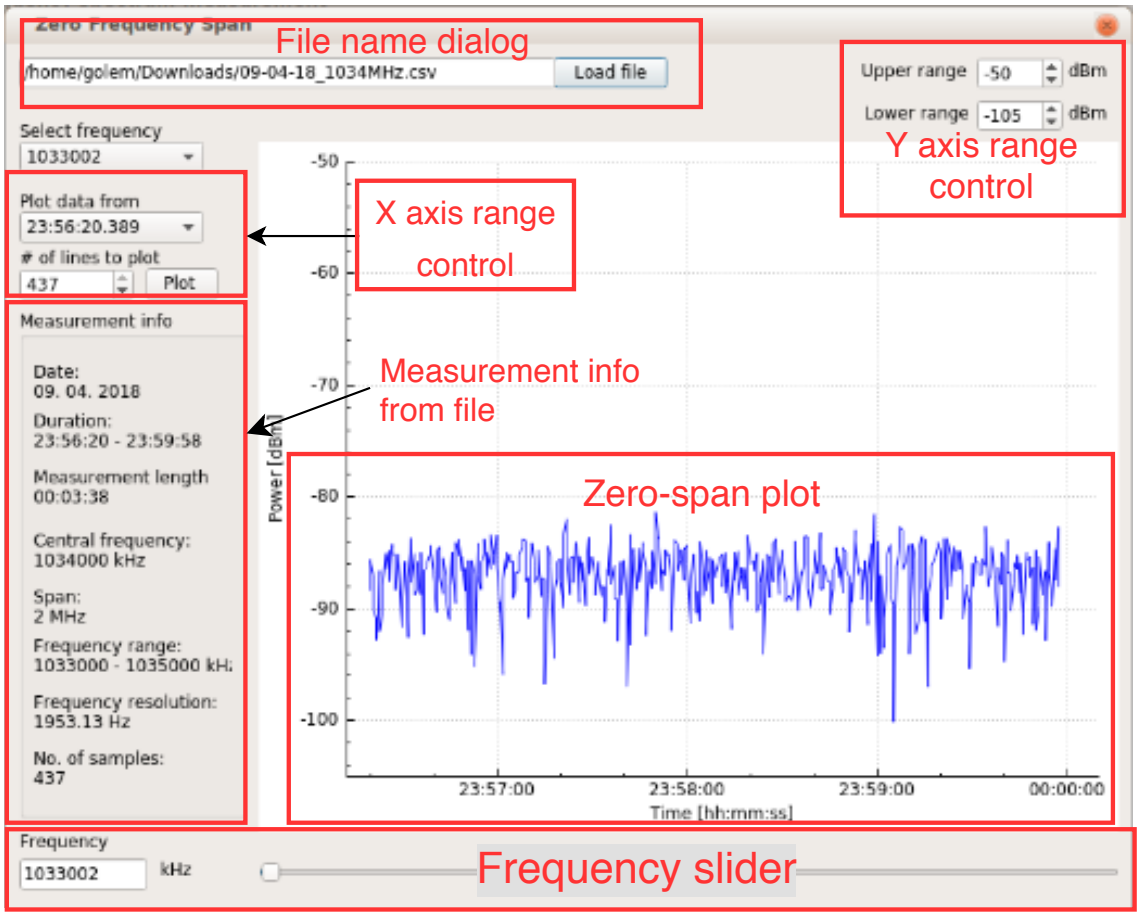


Fig. 4.9: Description of SDR Settings window

with three options: High-pass, Low-pass and Bypass. State of these parameters is after confirmation shown in lower part of the window

4.8 Zero Span

The purpose of this window is to plot RF power at the specific frequency over time. Data shown are historic – taken from file with the structure shown in 4.4. It's like a single row was taken. The main element of this window is QCPGraph surrounded by controllers

After launching the window there is no data loaded. These have to be loaded from a file. By clicking the Load button, a standard file dialog pops-up and the user selects the file. Then parameters of measurement are loaded. The most important are date and time of measurement and list of frequencies. Other measurement facts are for example duration, frequency step between single values and count of samples in that particular file.

Frequency to plot may be selected two ways. The first is a combo box with all

available frequencies and the other one is a slider on the bottom of this window. By dragging the slider user lists through available frequencies from the mentioned combo box. Actually, these controls are bounded to each other and their values are synced.

Y axis range

Resolution of power domain, or in other words Y axis range, is user-definable. In the upper right corner of the window, there are situated two spin boxes. Values of those have impact on respective range of Y axis. Graph replot happens automatically and user doesn't need to confirm this change.

X axis range

Concerning horizontal axis, there are more possibilities. Firstly, the user can define which data are plotted by selecting the start time and then the number of samples to be plotted from that point. To apply this change user needs to click the plot button.

When the graph is plotted, the user can use mouse wheel zoom and mouse drag range change the same way as in spectrum diagram in the Main Window.

4.9 Load Data from File

Clicking "Data from file button"launches the last dialog which is dedicated to browsing data from the previous measurement.

4.9.1 File input

After clicking the Open File button, a file dialog appears a user is prompted to choose one. The application doesn't check on correct format of the file, it relies on user's correct choice. As there is no requirement for compatibility, the only format suitable for plotting is the one produced by this application. This format has already been described in Fig. 4.3.

4.9.2 Measurement info

This infobox displays to the user some basic information about parameters of measurement which might be determined before reading data alone. Regarding time there is date, start&stop and duration of measurement.

From spectral aspect there is an information about central frequency, span (eventually frequency range) and frequency resolution.

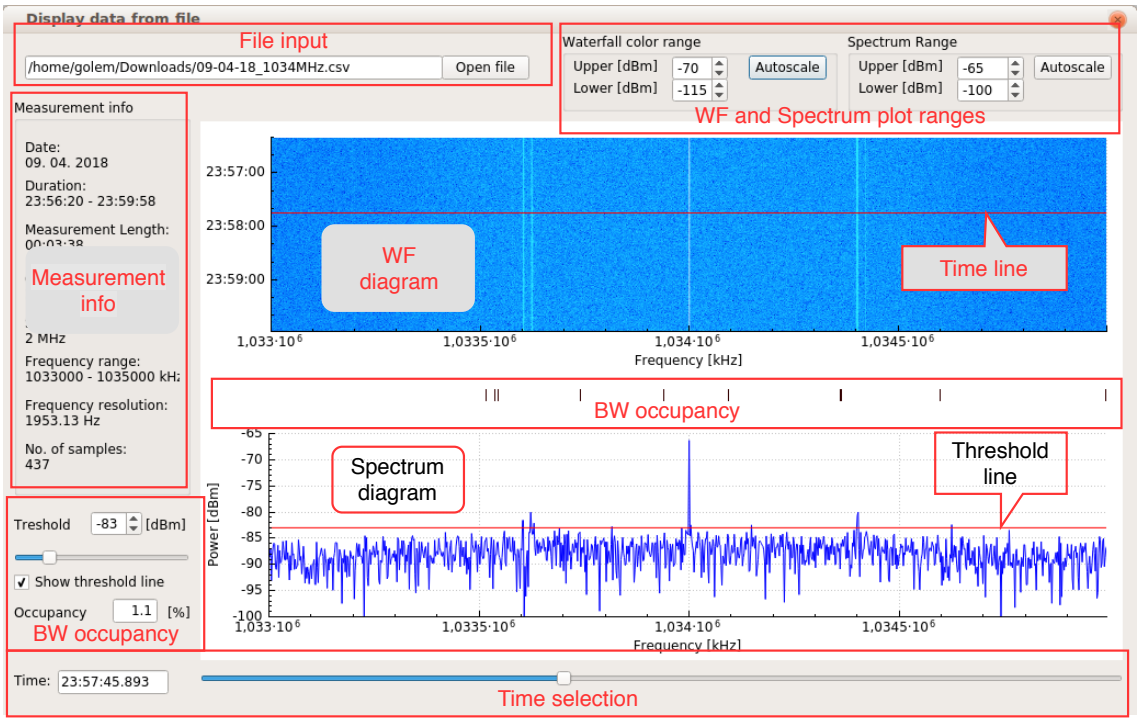


Fig. 4.10: Dialog for browsing data from file

4.9.3 Waterfall Diagram

Waterfall diagram is capable of displaying longer period of data as it showing three parameters at once. Two of them are bounded to axis - time and frequency and the third parameter (power) is represented by color. As historical data are plotted, mentioned features make this kind of graph great as primary overview of the acquired data.

Ranging the plot

After loading the file with results the whole measurement is displayed. This might be the whole day as well. To explore this data in detail, mouse interactions are allowed in this graph. On the contrary with the graph in main window, here the zoom and Range dragging is allowed in the vertical plane – in time.

To change the color range there are two spinboxes above this graph. Their value represent ranges of the color mapping. Changing their values therefore affects only color display of the map and ranges of both axes remain untouched.

Next to this pair of range controllers, there is a button Autoscale which sets color range of the map to cover all values present in the WD. Its secondary action is rescaling of Y axis to show all data.

QCPColorMap limitation

QCPColorMap has a limitation of maximal count of samples shown - 32767 data points in both directions. As there may be up to $24 \cdot 3600 \cdot 2 \approx 173000$ samples in single file, there exists a need to take care of that. In case where this condition would be void, only every n-th sample (where $n = (Tot\%32767) + 1$ – Tot is the total count of samples) is displayed in the graph.

4.9.4 Spectrum Diagram

Most of the functionality is the same as in Main Window, but there are some differences.

Ranging the plot

Here, the spectrum diagram's and WD's range are not tied together as zooming in the frequency domain is not allowed. User can drag and zoom in the plot with mouse and in certain range, these new values are set in the spinboxes above graphs. Also for this plot there is Autoscale button present.

Time selection

To plot certain time moment in Spectrum diagram, another feature not used anywhere else is implemented. All times of samples are loaded into internal vector variable and slider's value represents the index in that variable. Then, values at the respective line in the file are plotted into Spectrum diagram. To make this selection visual, therefore more comfortable, a red line was added into the waterfall diagram to show selected time moment. However, the data to plot in spectrum diagram is still selected only by the slider.

4.9.5 Bandwidth occupancy

is implemented the same way like in Main Window, see 4.5.6.

4.10 Source codes

All sources codes including commits of the progress are available at `https://github.com/petrsvobodnik/DP`.

4.11 Running the application

Application has been compiled using static linking to the libraries. However, it presumes presence of some standard libraries which should be present in Ubuntu distribution. Namely it is libusb-1.0 which might eventually be installed the standard way using command `sudo apt-get install libusb-1.0-0-dev`.

To launch the application user needs to enter application's folder in Terminal and then enter command `./hack_connect.sh` (without space after slash). The other way is to allow running executable text files in system settings and then, clicking the hackconnect.sh icon runs the application.

4.12 Calibration of the radio

For providing representative results of the measurement the SDR needs to be calibrated to match actual and displayed value of RF power level. For obtaining this offset value, the HackRF radio was connected to laboratory generator Agilent N9310A, and at frequency 100 MHz was set distinguishable output RF power. The difference of value displayed in spectrum diagram and actual value equals to the offset which was added to the application internally.

However, this calibration is tied to this specific radio (which was provided to me by CTU and it is their property) as the reception sensitivity is in some range of values.

4.13 Known issues

There are some bugs discovered during testing of the application that haven't been resolved:

- HackRF needs to be connected to computer before launching the application. The radio is not detected if it's plugged in during application's run.
- Only one item may be load during session – when attempting to open file after another one had been already opened, the data is not interpreted correctly. Application has to be shut down and relaunched
- GUI widgets do not adapt to window size – size of windows are now set to be fixed.
- Length of measurement information – this field sometimes stays blank as the conversion from QString data type fails
- DC offset – in the middle of FFT calculation (consequently in all the graphs) there is a narrow line at the central frequency which is caused by IQ imbalance

Tab. 4.1: Dependency of HackRF's indicated power level [dBm] versus distance from central frequency for each sampling rate

Sample rate	-50 %	-40 %	-30 %	-20 %	-10 %	10 %	20 %	30 %	40 %	50 %
2 MHz	-20	-20.1	-21	-20.3	-20.2	-20	-19.8	-21.7	-20.3	-20.2
4 MHz	-27	-22.8	-20.4	-21.8	-20	-20.7	-21	-21.4	-23.2	-27
8 MHz	-26	-23.3	-21.5	-22.1	-21.7	-22.9	-22.6	-21.7	-24.2	-26.4
10 MHz	-30.3	-24.2	-22.6	-23.3	-21.4	-22.5	-22.4	-22.3	-24.5	-30.2
20 MHz	-21	-21.5	-19.5	-20.4	-20	-20	-20	-20.6	-21	-25

in SDR receiver. However, this is more considered to be feature of SDR than a bug.

4.14 Demonstrational measurement

To demonstrate application's functionality, a measurement was performed. As the CTU's monitoring car isn't ready yet, the used source of signal is RF signal generator Agilent N9310A.

In order to test behaviour across measured range, frequency sweep was used. Output power was set to -20 dBm, central frequency 85 MHz with span consecutively set to all possible values – 2, 4, 8, 10 and 20 MHz. Application was set to identical central frequency and sample rate resulting in sweeping across the whole displayed range.

Non-linearity of RX level

Small preview of the measurement at sample rate 10 MHz is shown in figure 4.11. In the picture there is noticeable how HackRF's sensitivity of reception is dependent while source's power level is constant.

In the table 4.1 there are listed values of indicated power level at constant generator's RF output power. For each sample rate there are 10 data values covering measured range. Values in first row express percentage of sample rate from the central frequency.

Data show that values close to the ranges of the measurement evince significant error from the actual value. Surprisingly, this error is smallest at the highest sample rate. The way to cope with this error is to use for measuring only 60 – 80 % of central span.

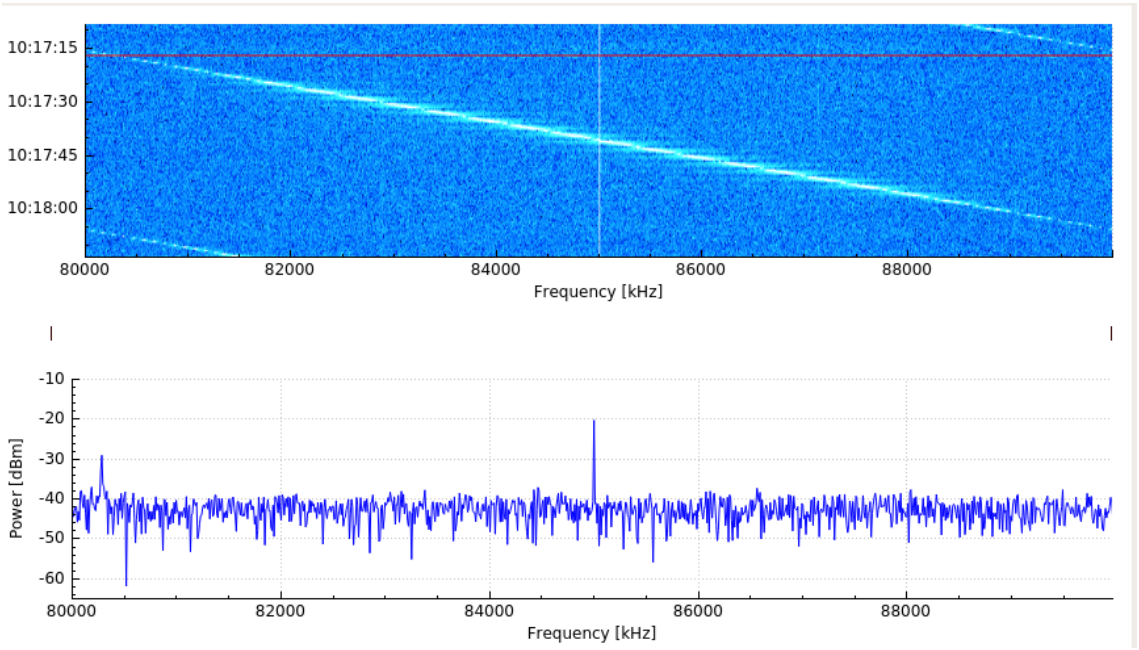


Fig. 4.11: Screenshot from the test measurement

Mirror images

Another problem I came across during this measurement is high susceptibility to shadow mirror images, especially when using lower sample rates. These images are almost as strong in amplitude as original signal as seen in 4.12. However, there exists pretty simple solution. Turning on VGA amplifier suppresses mentioned images efficiently. When its gain is set to 10 dB, images disappear completely as may be seen in the lower part of WD in the same figure.

Screenshot from the measurement in bigger scale is in Appendix A, files with saved data are present on the CD attached with this thesis. In each of file there is first measurement in default settings and after finishing frequency sweep the VGA gain was set to 10 dB. These files are compatible with presented application.

Conclusion

Measured band needs to be within offset range with acceptable error of sensitivity. Trying to correct these values would lead to distortion of measurement. For getting valid data, operator performing measurement using this station needs to be aware of this fact and shouldn't take values close to side borders of the graph seriously.

Also, attention should be paid to mirror images, these devalue measurement a lot. Method of getting rid of them was mentioned.

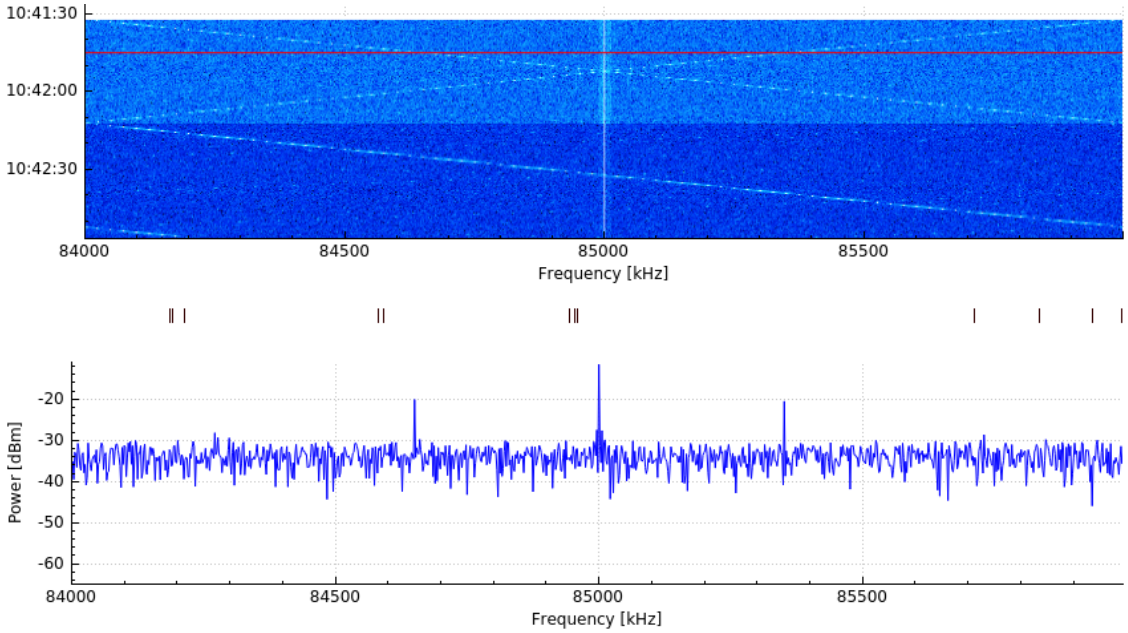


Fig. 4.12: Mirror images at low sample rates are visible in both spectrum and waterfall diagram. As frequency sweep was performed, in WD there is apparent image resembling letter X, which is the result of mirroring.
In the lower part of the image the frequency sweep keeps running, but VGA gain was set to 10 dB.

5 CONCLUSION

5.1 Main task

The task of this thesis was to find a suitable existing SDR for project of Portable Monitoring Station and to develop an application that would be capable of processing of acquired data, displaying them in several types of plot, proceeding basic spectral measurement and saving results into file.

First, the reader was introduced into problematics of spectrum monitoring and the need for monitoring was explained. Then, methods of spectrum measurement were described with further references to ITU's official guides and recommendations. In following chapter available SDR's were presented and suitable radio was selected.

Regarding the application, first communication with HackRF had to be established and to acquire measured data. After solving some compatibility issues with computer (Sony laptop doesn't support Linux properly), the connection was successful and the data stream started to flow right into FFT. Results of calculation take in consider RF Unit's gain before the data is displayed in all the graphs. Measured data is saved in custom data format. For re-opening historical data there are implemented methods with different representation of data to comply with CTU's requirements.

Functionality of all the application's windows was described in the last chapter. In that chapter was laid stress on being comprehensive, so it serves as manual for operators of the application. Then, instructions for executing the application were provided, including some troubleshoot links.

Demonstration measurement was performed, evaluated and issues were discussed.

Functionality of the complete Portable Monitoring Station couldn't be tested as the rest of the system is not ready yet by the time of handing this thesis in.

5.2 Non-realized requirements

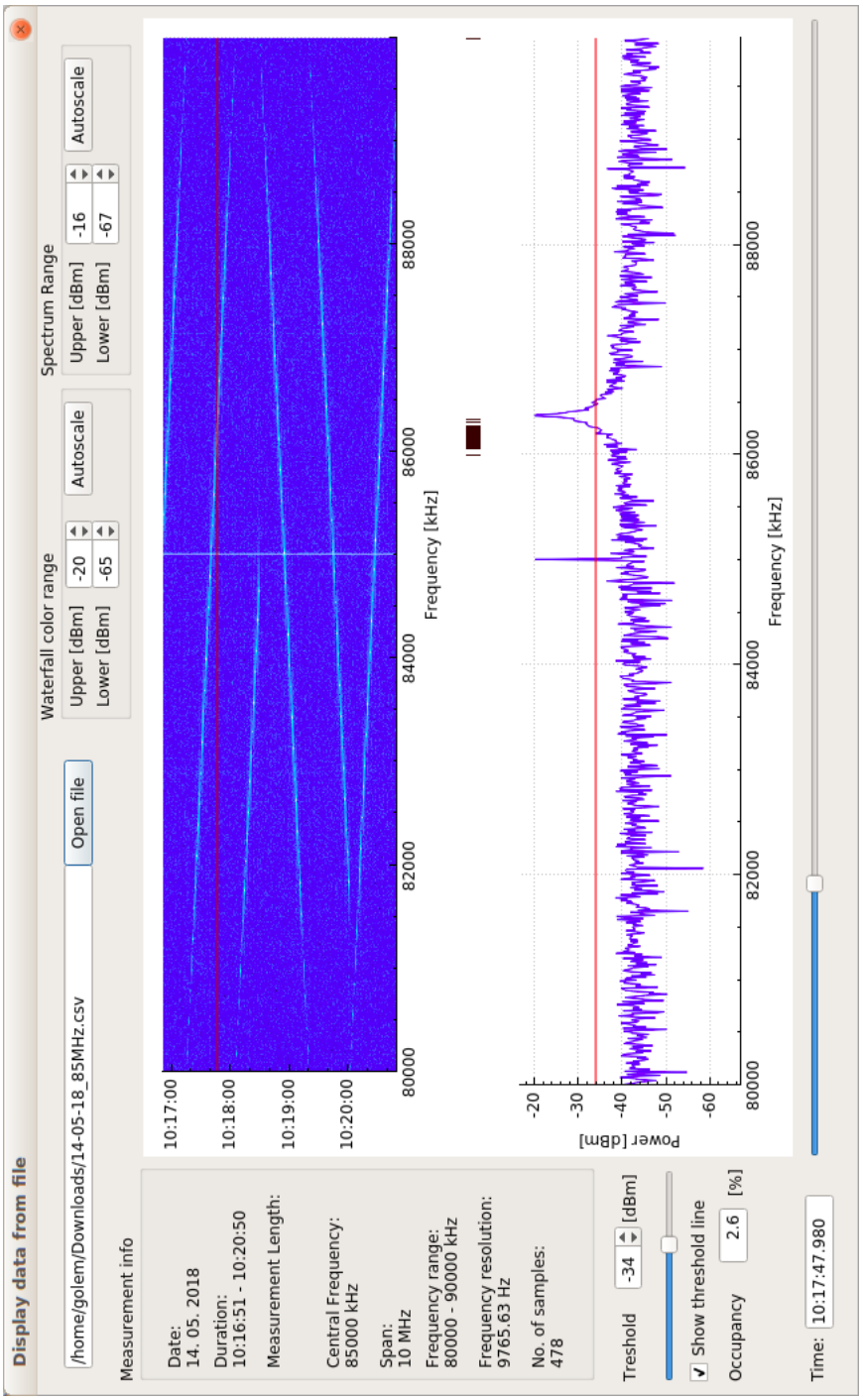
In the original assignment there is a requirement that haven't been met – the lower frequency range. This part of assignment haven't been fulfilled and the upconverter hasn't been used at all. Therefore the working range of the application is now given by possibilities of used SDR – 1 MHz – 6 GHz.

5.3 Further possible progress

This application could be further expanded to accommodate change of polarization (once the antenna holder is ready) and for the ability of using upconverter to cover the whole frequency band. Another request from ordering party is a standalone application for controlling the RF Unit which would be OS Windows compatible. To reach this, some minor changes are required and then there is a need to deploy the application again for different OS.

LIST OF APPENDICES

A RESULTS OF DEMONSTRATION MEASUREMENT



B ATTACHED CD

List of files on the CD attached with this thesis:

- DP_xsvobo98.pdf – the main file with diploma thesis
- hack_connect – folder with all the sources codes
- redist – folder with compiled application
- demo_meas – folder with demonstrational measurements. For each sample rate there is a file in separate folder with respective name

The CD is glued to the back cover of the thesis

BIBLIOGRAPHY

[1] ITU. *Handbook on Spectrum Monitoring*. 1. Geneva, Switzerland: ITU, Radiocommunication Bureau, 2011. ISBN 9789261188818.

[2] ITU. *Recommendation ITU-R SM.1050-2: Tasks of a monitoring service*. 2004

[3] ITU. *Recommendation ITU-R SM.1139: International monitoring system*. 1995

[4] ITU. *Recommendation ITU-R SM.1537-1: Automation and integration of spectrum monitoring systems with automated spectrum management*. 2013

[5] ITU. *Recommendation ITU-R SM.2039: Spectrum monitoring evolution*. 2013

[6] ITU. *Radio Regulations*. Edition 2016. Genova, 2016. ISBN 978-9261199975.

[7] Fast Fourier transform. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-12-02]. Available from: `www.en.wikipedia.org/wiki/Fast_Fourier_transform`

[8] FFT Fast Fourier Transform. *Nti-audio.com* [online]. [cit. 2017-12-02]. Available from: www.nti-audio.com/en/functions/fast-fourier-transform-fft.aspx

[9] ITU. *Recommendation ITU-R SM.1880-2: Spectrum occupancy measurements and evaluation*. 2017

[10] MITOLA, Josef. Software radios: Survey, critical evaluation and future directions. *IEEE Aerospace and Electronic Systems Magazine*. IEEE, 1993, **8**(4), 25–36.

[11] GRAYVER, Eugene. *Implementing software defined radio*. 1. New York: Springer, 2012. ISBN 14-419-9332-0.

[12] WAH, Benjamin W. *Wiley encyclopedia of computer science and engineering*. 1. Hoboken, N.J.: John Wiley, c2009. ISBN 978-0471383932.

[13] KENINGTON, Peter B. *RF and baseband techniques for software defined radio*. 1. Boston: Artech House, c2005. Artech House mobile communications series. ISBN 15-805-3793-6.

[14] ROKOS, L. *Vysokofrekvenční jednotka pro přenosnou monitorovací stanici*. Brno: Brno University of Technology, Faculty of electronics and communication, Department of Radioelectronics, 2017. Semestral project. Advisor: doc. Ing. Jiří Šebesta, Ph.D.

[15] ARS-USB *ea4tx.com* [online]. [cit. 2018-05-11]. Available from: https://ea4tx.com/en/products-page/ars-usb/

[16] LibHackRF API: Michael Ossmann. *Github.com* [online]. [cit. 2017-12-11]. Available from: https://github.com/mossmann/hackrf/wiki/libHackRF-API

[17] FFTW [online]. [cit. 2017-12-11]. Available from: www.fftw.org

[18] Qt Plotting Widget QCustomPlot *qcustomplot.com*, [online]. [cit. 2018-05-11]. Available from: http://www.qcustomplot.com/

[19] Qt Documentation - QSerialPort Class *doc.qt.io* [online]. [cit. 2018-05-11]. Available from: http://doc.qt.io/qt-5/qserialport.html#SerialPortError-enum

[20] Fix serial port permission denied errors on Linux *websistent.com* [online]. [cit. 2018-05-11]. Available from: https://websistent.com/fix-serial-port-permission-denied-errors-linux/

LIST OF SYMBOLS, PHYSICAL CONSTANTS AND ABBREVIATIONS

AD	Analog to digital
ADC	Analog to digital conversion
ASIC	Application Specific Integrated Circuit
CTU	Czech Telecommunication Office
DA	Digital to analog
DAC	Digital to analog conversion
DVB-T	Digital Video Broadcast – Terrestrial
DSP	Digital Signal Processing
FFT	Fast Fourier Transform
FPGA	Field Programmable Gate Array
f_s	sampling frequency
GUI	Graphical User Interface
IF	Intermediate Frequency
ITU	International Telecommunication Union
LNA	Low Noise Amplifier
LO	Local Oscillator
LTE	Long Term Evolution
QCP	QCustomPlot
RF	Radio Frequency
RR	Radio Regulations
RX	Receive
SDR	Software Defined Radio
SR	Software Radio
TX	Transmit
WD	Waterfall Diagram