

SDR und GNU-Radio

Teil 1: Software Defined Radio, Hardware

1.1 Analoger und digitaler Empfänger

1.2 Abtastung und Quantisierung

1.3 SDR-Prinzipien

1.4 Für GNU-Radio geeignete SDR-Geräte

1.5 RTL-Stick

1.6 ADALM-PLUTO

1.7 Ettus Research B200

1.8 SDR-Erweiterungen mit Standard-Baugruppen



SDR und GNU-Radio am 14. 04. 2021

Teil 2: GNU-Radio, Anwendungsbeispiele und Software-Voraussetzungen

2.1 GNU-Radio und GNU Radio Companion (GRC)

2.2 Beispiele in GRC

2.2.1 Audio-Oszillator

2.2.2 Spektrum-Analysator 70 MHz.... 6GHz

2.2.3 UKW-Radio mit Spektrum-Analysator

2.2.4 Filtersimulation

Live-Demo GNU-Radio

2.2.5 FM-TX

2.2.6 SSB-RX

2.2.7 KW-SSB-Transceiver nach W7FU

2.3 Betriebssysteme, Quelltext und Binaries, Programmiersprachen

2.4 Glossar



2.1 GNU-Radio und GNU-Radio-Companion

- GNU-Radio ist eine Open Source - Plattform
- GNU-Radio läuft unter LINUX, Windows und Mac OS
- GNU-Radio hat eine grafische Oberfläche GRC (GNU RADIO COMPANION). Es können eigene Blöcke in Python und C++ geschrieben werden
- Es gibt viele fertige Funktionsblöcke für alle Bereiche der digitalen Signalverarbeitung: Quellen, Senken, Modulatoren, Mischer, Demodulatoren, Generatoren, Filter....



2.1 GNU-Radio-Companion (GRC), die Oberfläche

The screenshot displays the GNU Radio Companion (GRC) interface. The main workspace, labeled "Arbeitsfläche", contains a flow graph with the following blocks: "Options" (ID: top_block, Generate Options: QT GUI), "Variable" (ID: samp_rate, Value: 32k), "Signal Source" (Sample Rate: 32k, Waveform: Cosine, Frequency: 1k, Amplitude: 1, Offset: 0), "QT GUI Sink" (FFT Size: 1.024k, Center Frequency (Hz): 0, Bandwidth (Hz): 32k, Update Rate: 10), and "QT GUI Time Sink" (Number of Points: 1.024k, Sample Rate: 32k, Autoscale: No). On the right, the "Bibliothek" (Library) pane lists various blocks, with "QT GUI Time Sink" selected. At the bottom, the "Terminal" pane shows the following output:

```
Executing: /usr/bin/python2 -u /home/berle/gnuradio/grcs/top_block.py
Gtk-Message: 16:33:46.357: Failed to load module "canberra-gtk-module"
fft_impl_fftw: /home/berle/gr_fftw_wisdom: Keine Berechtigung
>>> Done
Generating: /home/berle/gnuradio/grcs/top_block.py
>>> Warning: This flow graph may not have flow control: no audio or RF hardware blocks found. Add a Misc->Throttle block to your flow graph to avoid CPU congestion.
Executing: /usr/bin/python2 -u /home/berle/gnuradio/grcs/top_block.py
Gtk-Message: 16:34:56.448: Failed to load module "canberra-gtk-module"
fft_impl_fftw: /home/berle/gr_fftw_wisdom: Keine Berechtigung
```

Below the terminal, a "Variable" table is shown:

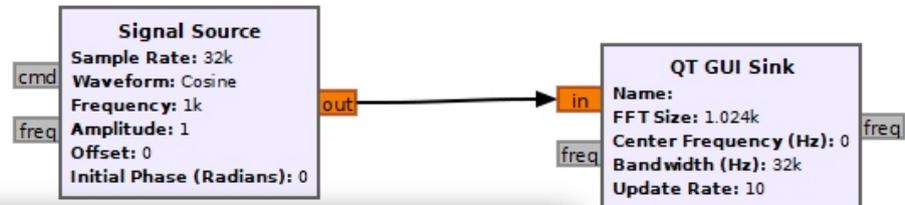
Id	Value
Imports	
Variables	
samp_rate	32000



2.2 Beispiele in GRC

2.2.1 Audio-Oszillator

Variable
ID: samp_rate
Value: 32k



Properties: Signal Source

General Advanced Documentation

Output Type: float

Sample Rate: samp_rate

Waveform: Cosine

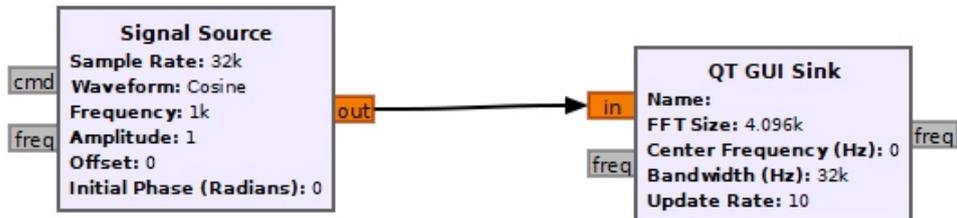
Frequency: 1000

Amplitude: 1

Offset: 0

Initial Phase (Radians): 0

OK Cancel Apply



Properties: QT GUI Sink

General Advanced Documentation

Type: float

Name: ""

FFT Size: 4096

Window Type: Blackman-harris

Center Frequency (Hz): 0

Bandwidth (Hz): samp_rate

Update Rate: 10

Show RF Freq: No

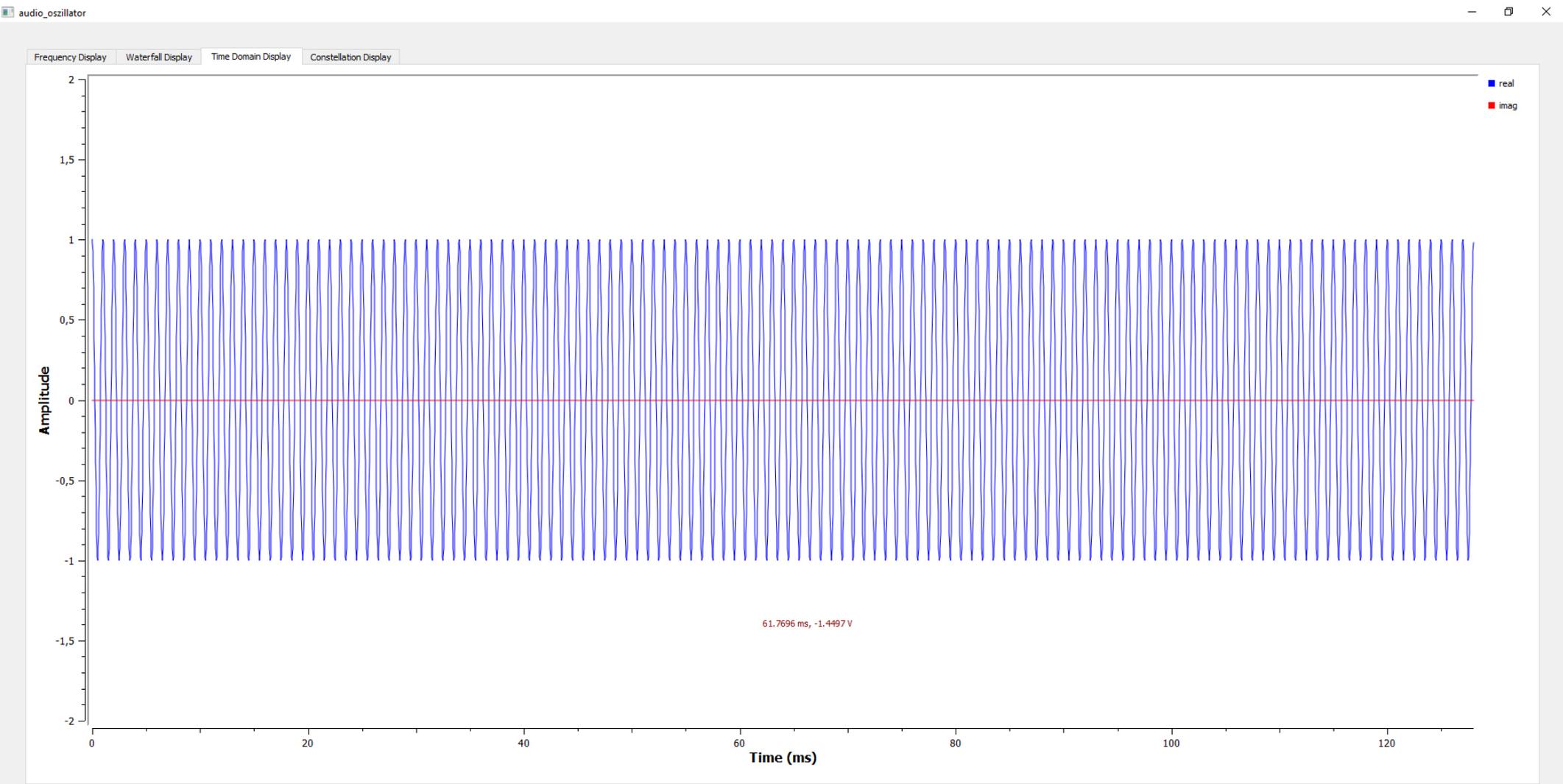
Plot Frequency: On

OK Cancel Apply

	Id	Value
dio or RF / graph to avoid	Imports	
	Variables	
	samp_rate	32000

on.exe -u D:-

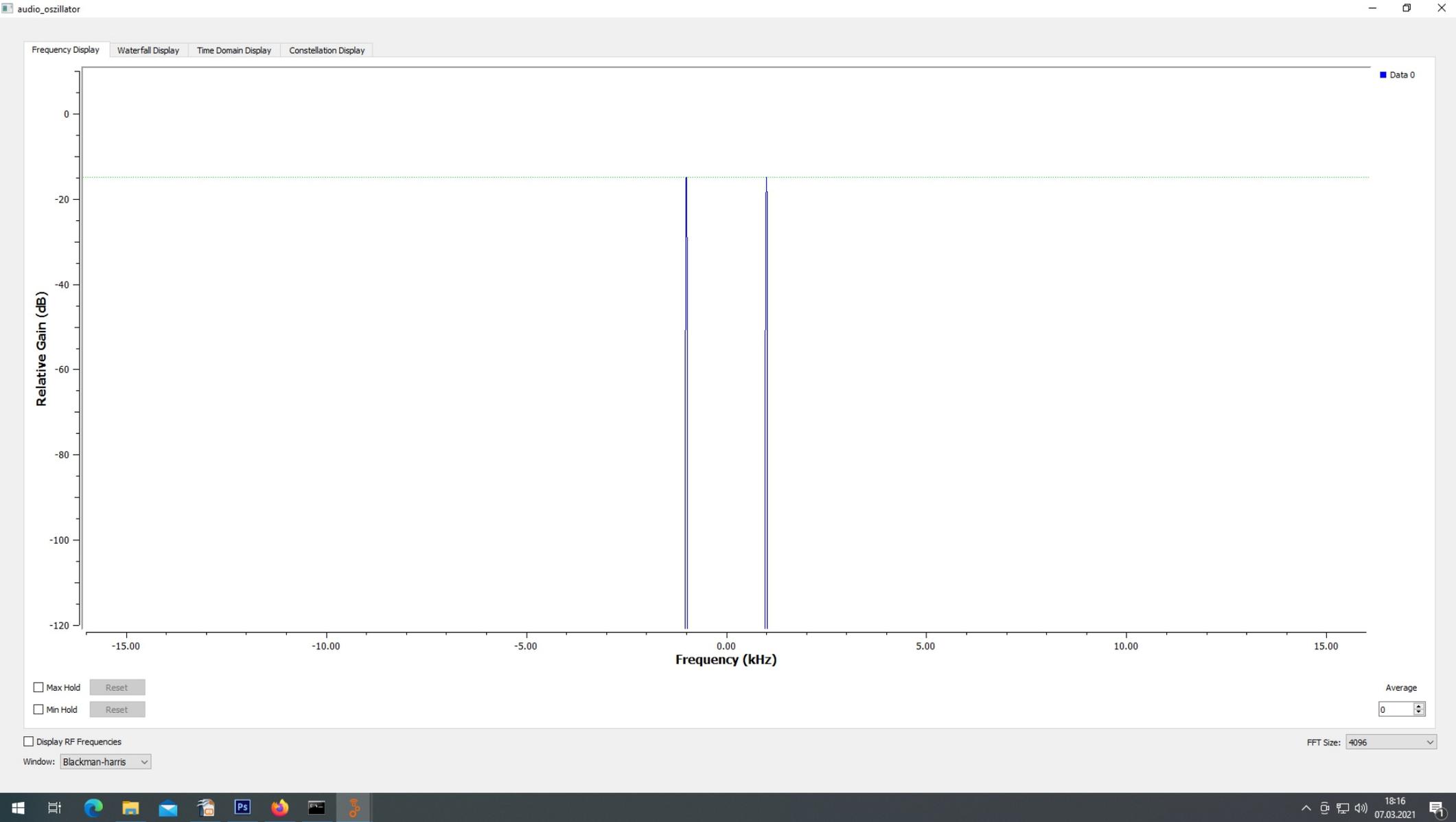


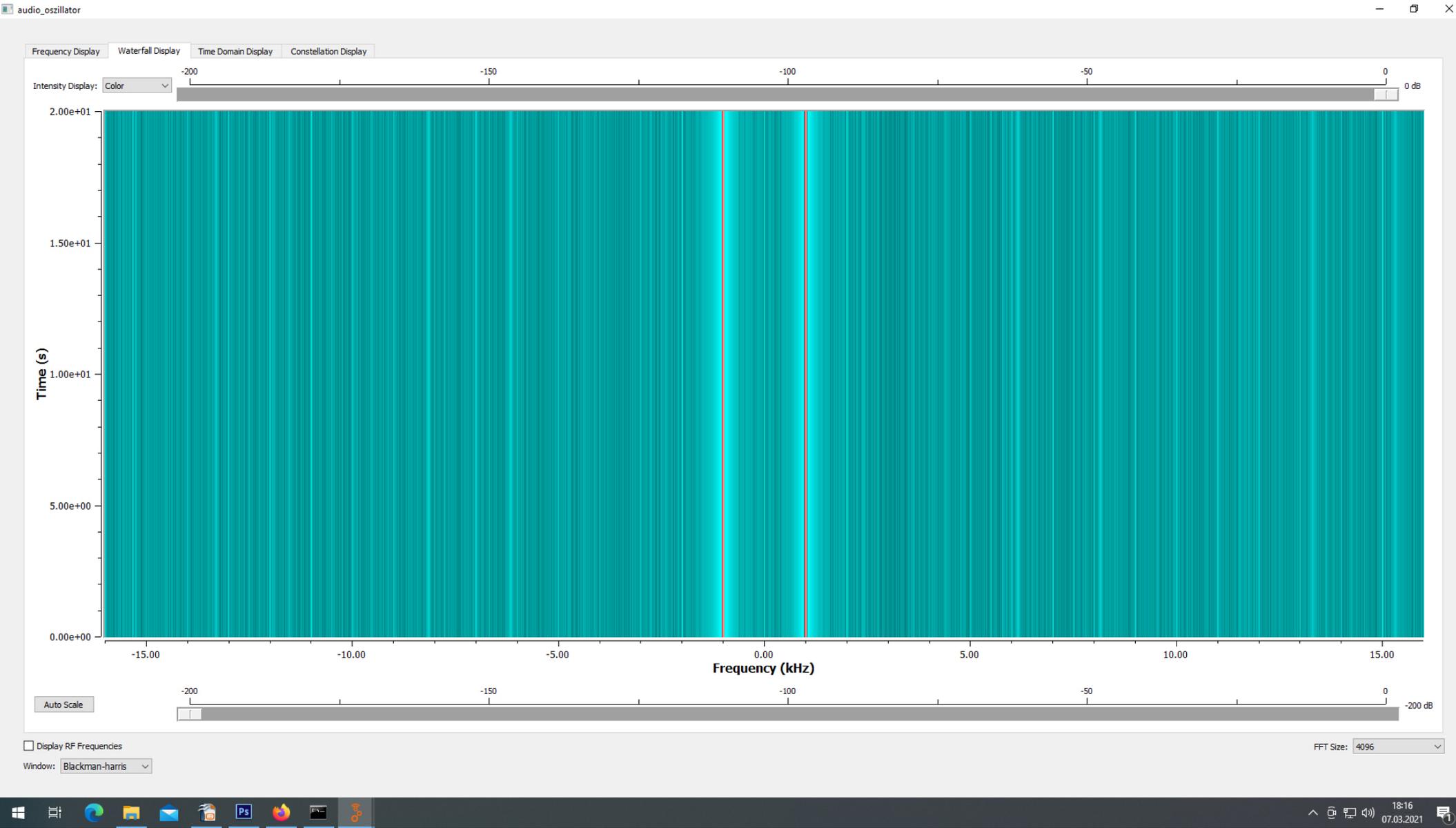


Display RF Frequencies
Window: Blackman-harris
FFT Size: 4096

Windows taskbar showing icons for Start, Task View, Edge, File Explorer, Mail, Photos, PS, Firefox, and other applications. System tray shows volume, network, and date/time: 18:17, 07.03.2021.







Jetzt können wir simulieren:

- einen Funktionsgenerator (und das Signal im Audiobereich zu einer Soundkarte leiten)
- ein Oszilloskop
- einen Spektrumanalysator
- ein Wasserfall-Diplay



2.2.2 Spektrum-Analysator 70 MHz.... 6GHz mit Ettus B200

Options

ID: top_block

Generate Options: QT GUI

QT GUI Range

ID: tuning
Label: Frequency
Default Value: 100M
Start: 70M
Stop: 6G
Step: 10k

QT GUI Range

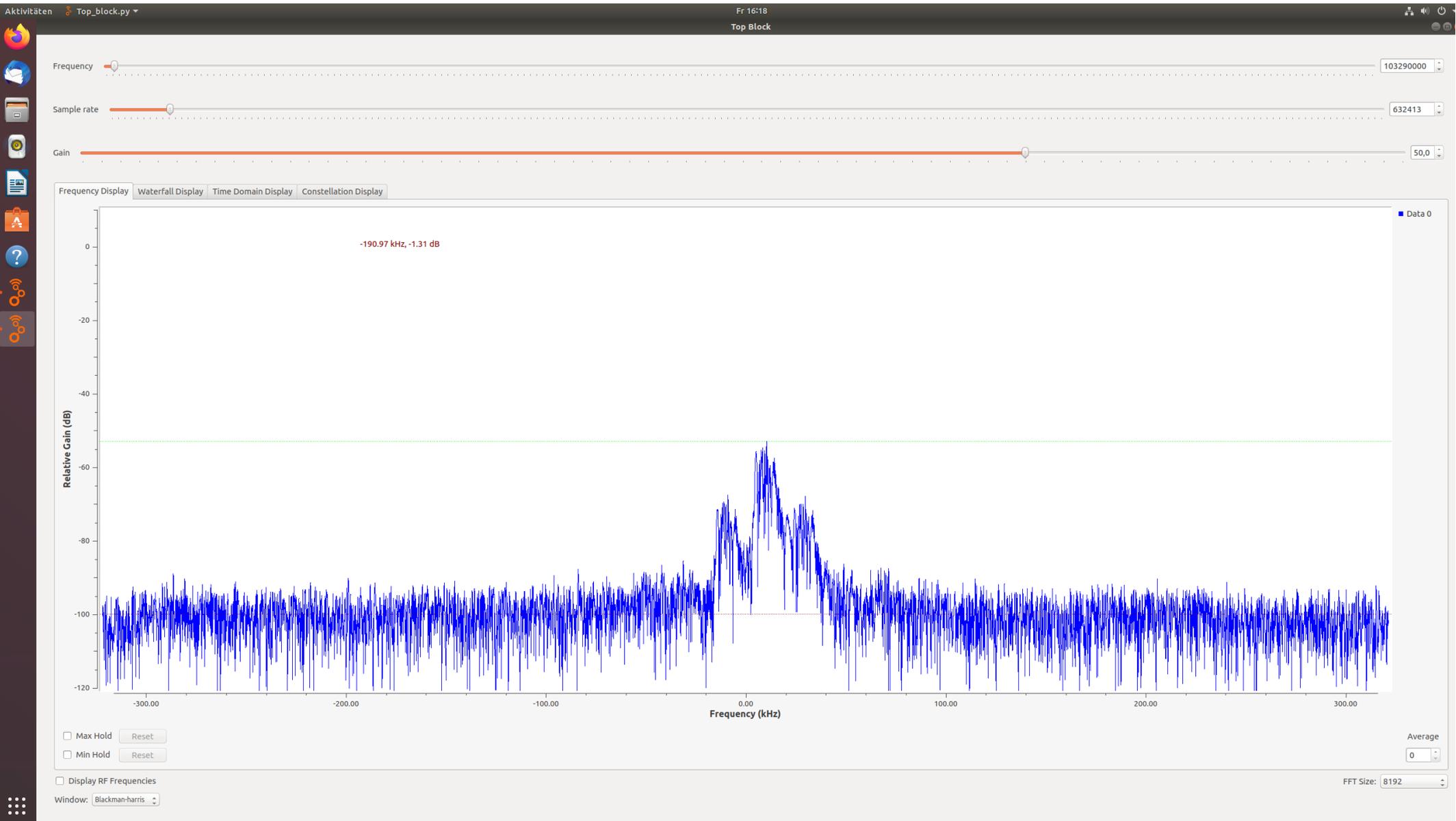
ID: sample_rate
Label: Sample rate
Default Value: 192k
Start: 192k
Stop: 10M
Step: 1

QT GUI Range

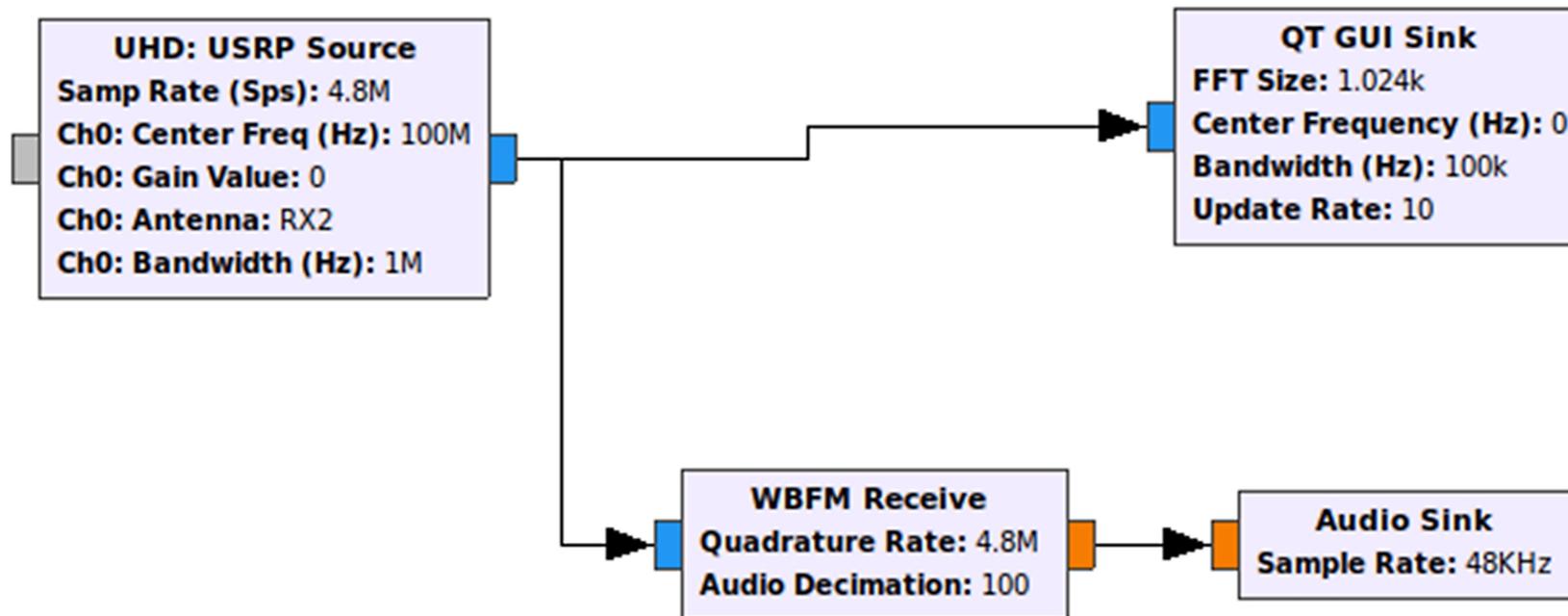
ID: gain
Label: Gain
Default Value: 50
Start: 0
Stop: 70
Step: 1

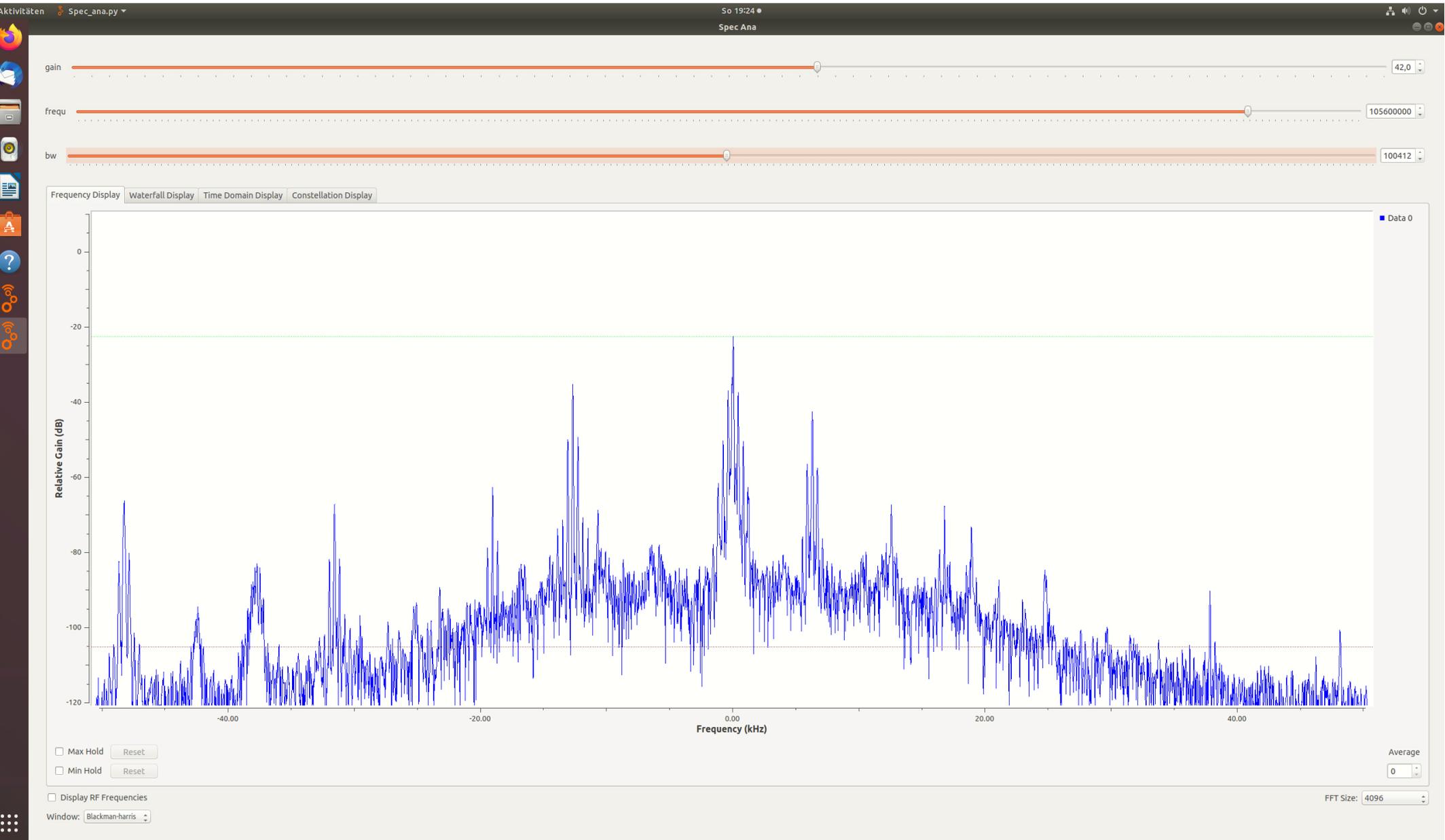




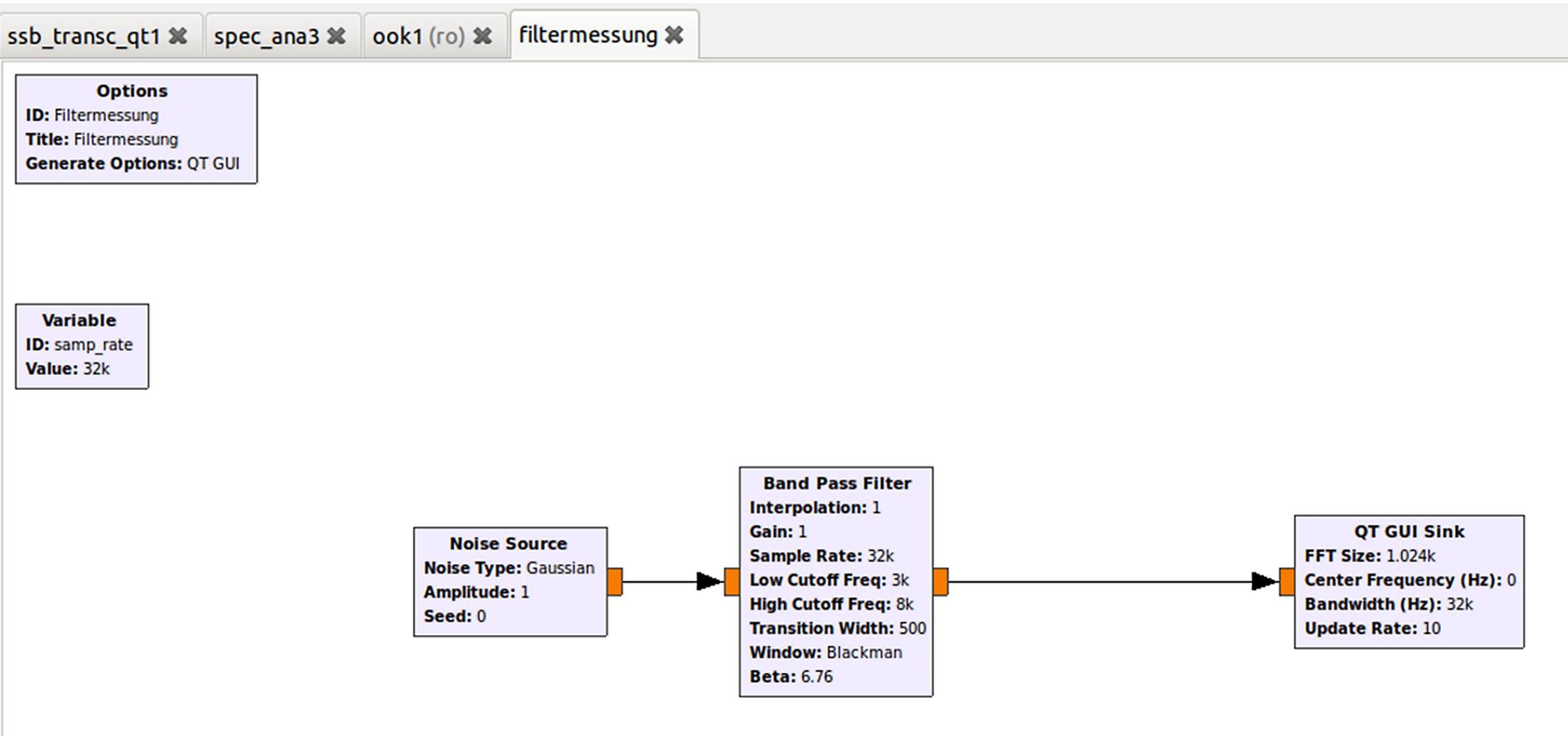


2.2.3 UKW-Radio mit Spektrum-Analysator

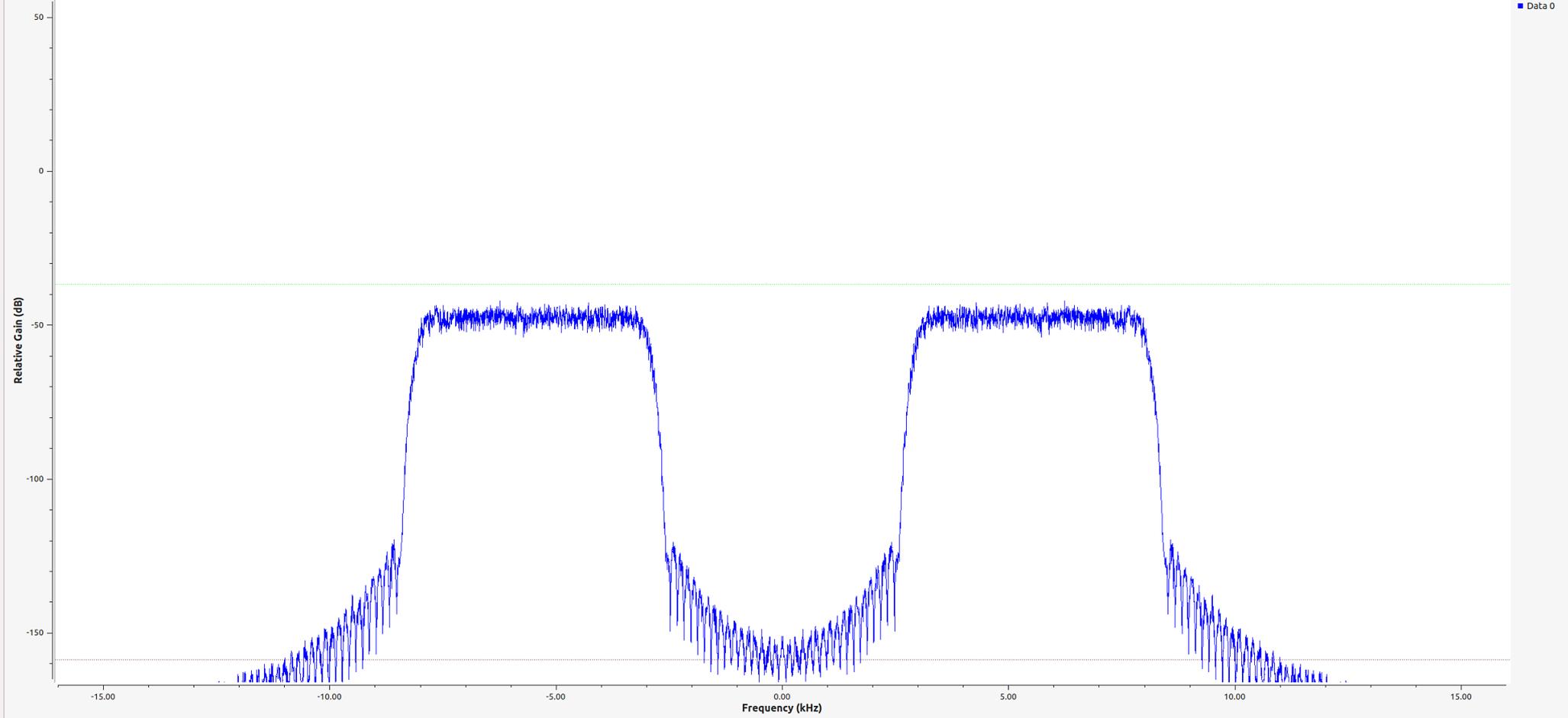




2.2.4 Filtersimulation



Frequency Display Waterfall Display Time Domain Display Constellation Display



Max Hold
 Min Hold

Average
7

Display RF Frequencies

FFT Size: 8192

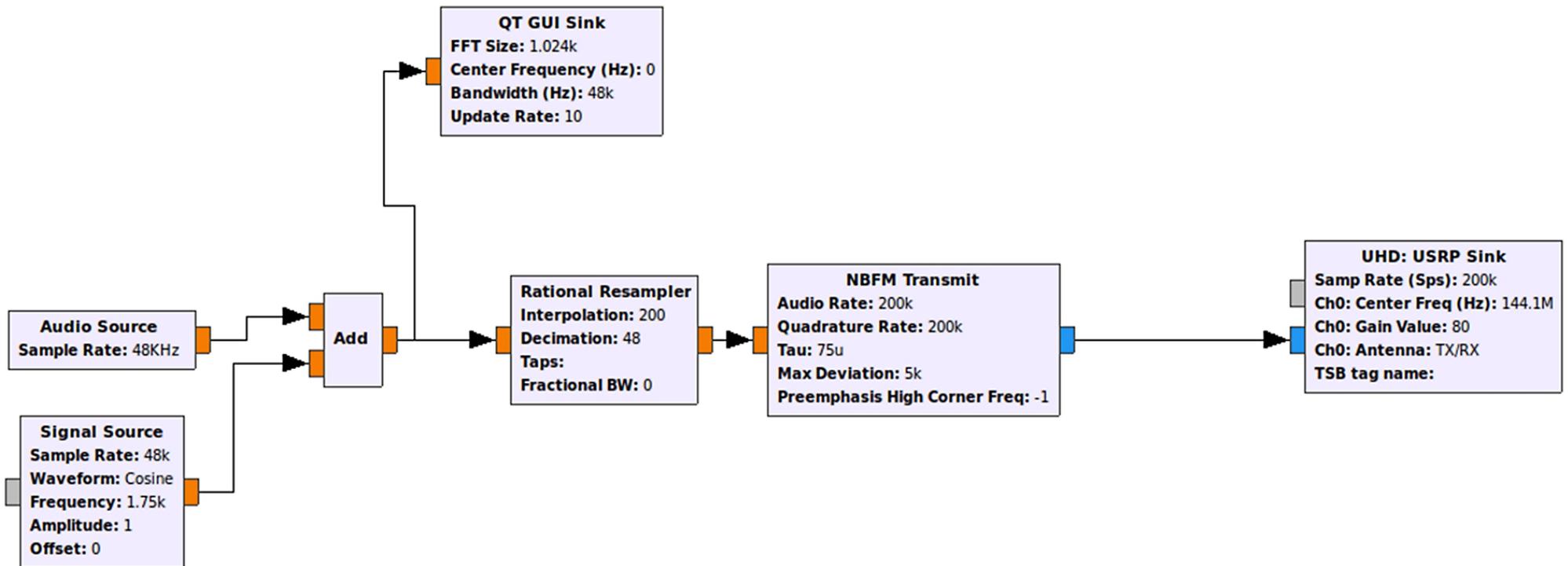
Window: Blackman-harris

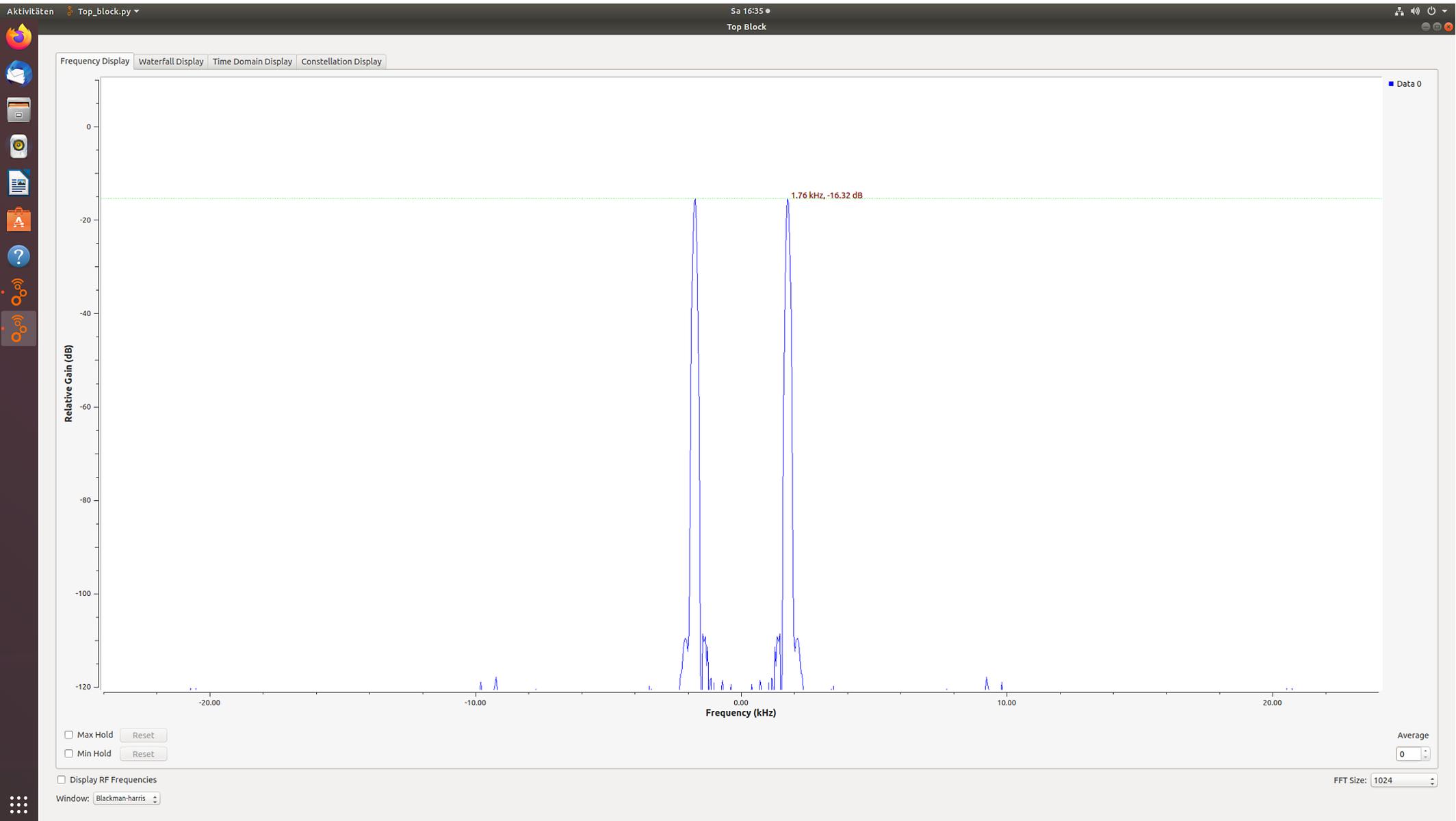


Live-Demo



2.2.5 FM-TX





2.2.6 SSB-RX



Options
 ID: vhf_weaver
 Title: vhf_weaver
 Author: dk2ft
 Generate Options: WX GUI
 Realtime Scheduling: On

WX GUI Chooser
 ID: freq_chooser
 Label: FREQUENCY SELECT
 Default Value: 0
 Choices: 0, 1, 2, 3, 4, 5
 Labels: 144.2, ...,401, 5760.2
 Type: Radio Buttons
 Grid Position: 3, 1, 1, 1

Low-pass Filter Taps
 ID: variable_...filter_taps_0
 Gain: 1
 Sample Rate (Hz): 1M
 Cutoff Freq (Hz): 5k
 Transition Width (Hz): 200
 Window: Hamming
 Beta: 6.76

WX GUI Slider
 ID: tune
 Label: Tune
 Default Value: 0
 Minimum: -61k
 Maximum: 61k
 Converter: Integer
 Grid Position: 5, 1, 1, 1

WX GUI Slider
 ID: fine_tune
 Label: Fine Tune
 Default Value: 0
 Minimum: -2.5k
 Maximum: 2.5k
 Converter: Float
 Grid Position: 6, 1, 1, 1

Variable
 ID: bpf_low
 Value: 100, 100, 380, 550

Variable
 ID: bpf_high
 Value: 3.9k, 2.7k, 880, 610

WX GUI Chooser
 ID: var_bw_0
 Label: Receiver...width - kHz
 Default Value: 0
 Choices: 0, 1, 2, 3
 Labels: 3.9, 2.7, 0.5, 0.05
 Type: Radio Buttons
 Grid Position: 5, 2, 1, 1

Variable
 ID: samp_rate
 Value: 1M

Variable
 ID: lo_freq_Hz
 Value: 144.2M, ...,1G, 5.7602G

WX GUI FFT Sink
 Title: FFT Plot1
 Sample Rate: 1M
 Baseband Freq: 0
 Y per Div: 10 dB
 Y Divs: 10
 Ref Level (dB): 0
 Ref Scale (p2p): 2
 FFT Size: 1.024k
 Refresh Rate: 15
 Freq Set Varname: None

Frequency Xlating FFT Filter
 Decimation: 4
 Taps: variable_low_pass_fi...
 Center Frequency: 0
 Sample Rate: 1M
 Num. Threads: 1

UHD: USRP Source
 Samp Rate (Sps): 1M
 Ch0: Center Freq (Hz): 144.2M
 Ch0: Gain Value: 1
 Ch0: Gain Type: Normalized
 Ch0: Antenna: RX2
 Ch0: Bandwidth (Hz): 1M

WX GUI FFT Sink
 Title: FFT Plot2
 Sample Rate: 250k
 Baseband Freq: 0
 Y per Div: 10 dB
 Y Divs: 10
 Ref Level (dB): 0
 Ref Scale (p2p): 2
 FFT Size: 1.024k
 Refresh Rate: 15
 Freq Set Varname: None

Signal Source
 Sample Rate: 250k
 Waveform: Cosine
 Frequency: 1.5k
 Amplitude: 1
 Offset: 0

Complex To Float

Signal Source
 Sample Rate: 250k
 Waveform: Sine
 Frequency: 1.5k
 Amplitude: 1
 Offset: 0

Multiply

Multiply

Add

Band Pass Filter
 Interpolation: 1
 Gain: 1
 Sample Rate: 250k
 Low Cutoff Freq: 100
 High Cutoff Freq: 3.9k
 Transition Width: 100
 Window: Blackman
 Beta: 6.76

Throttle
 Sample Rate: 250k

Rational Resampler
 Interpolation: 48k
 Decimation: 250k
 Taps:
 Fractional BW: 0
 Constant Source
 Constant: 100m

Multiply

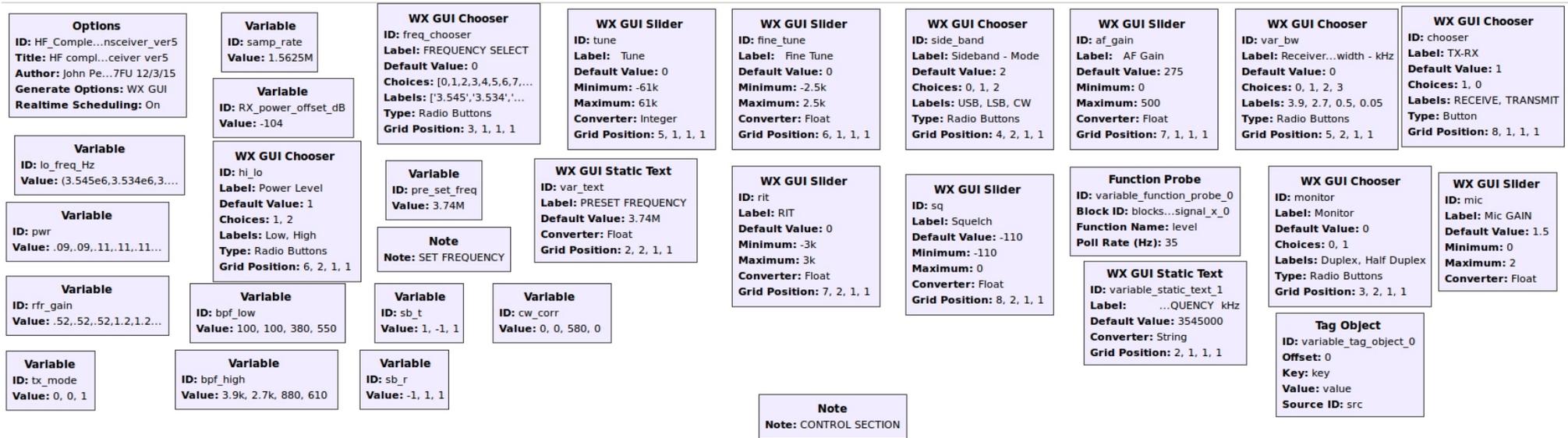
Multiply Const
 Constant: 300

WX GUI FFT Sink
 Title: FFT Plot3
 Sample Rate: 48k
 Baseband Freq: 0
 Y per Div: 10 dB
 Y Divs: 10
 Ref Level (dB): 0
 Ref Scale (p2p): 2
 FFT Size: 1.024k
 Refresh Rate: 15
 Freq Set Varname: None

Audio Sink
 Sample Rate: 48KHz



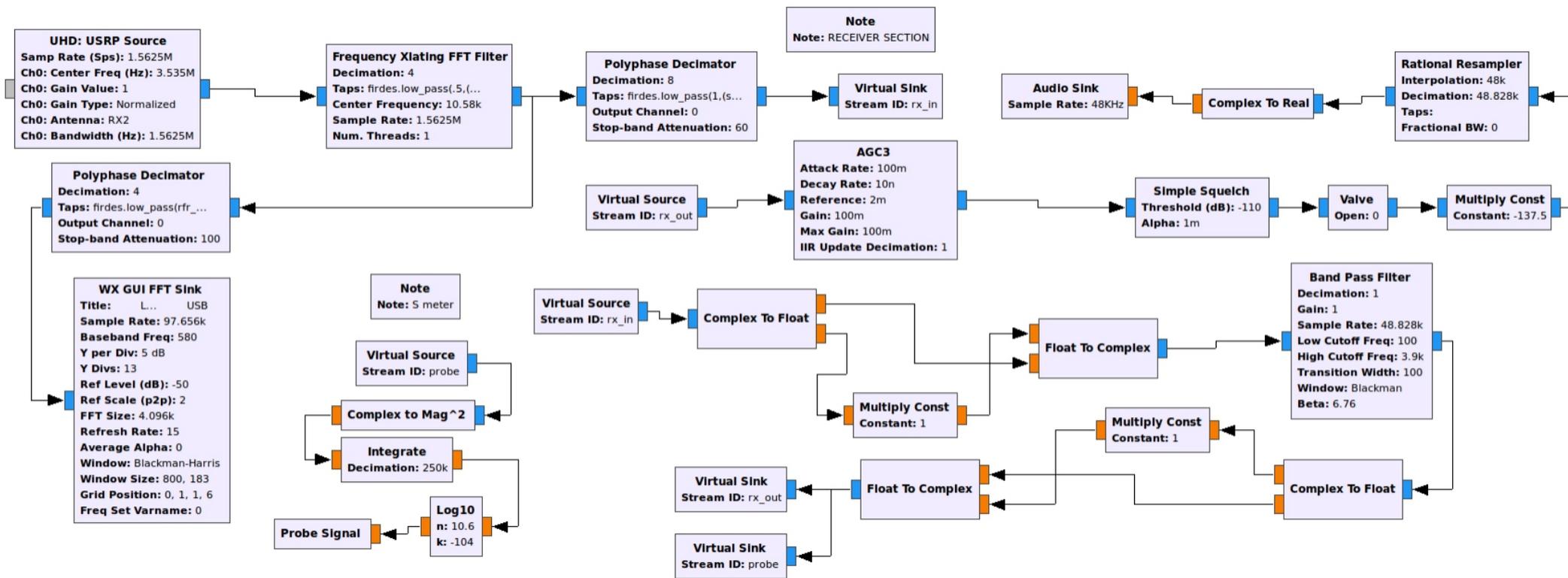
2.2.7 KW-SSB-Transceiver nach W7FU

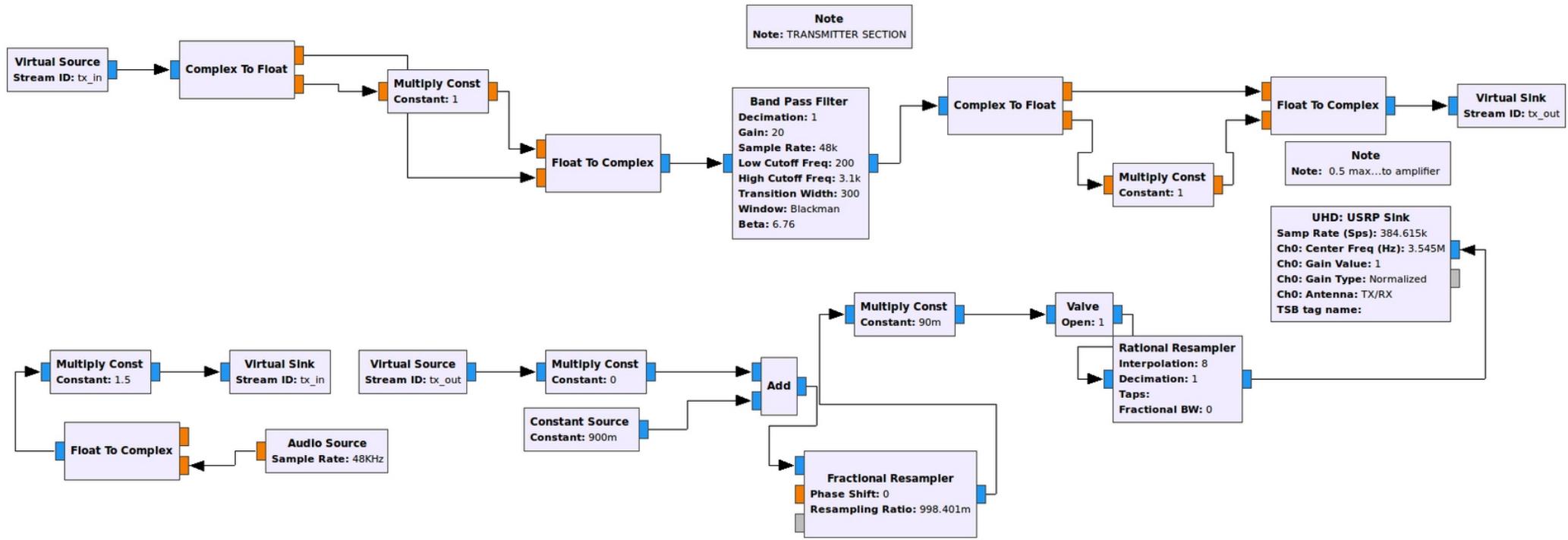


Control Segment of Universal Multi-Mode Transceiver Flowgraph

[www.w7fu.com]

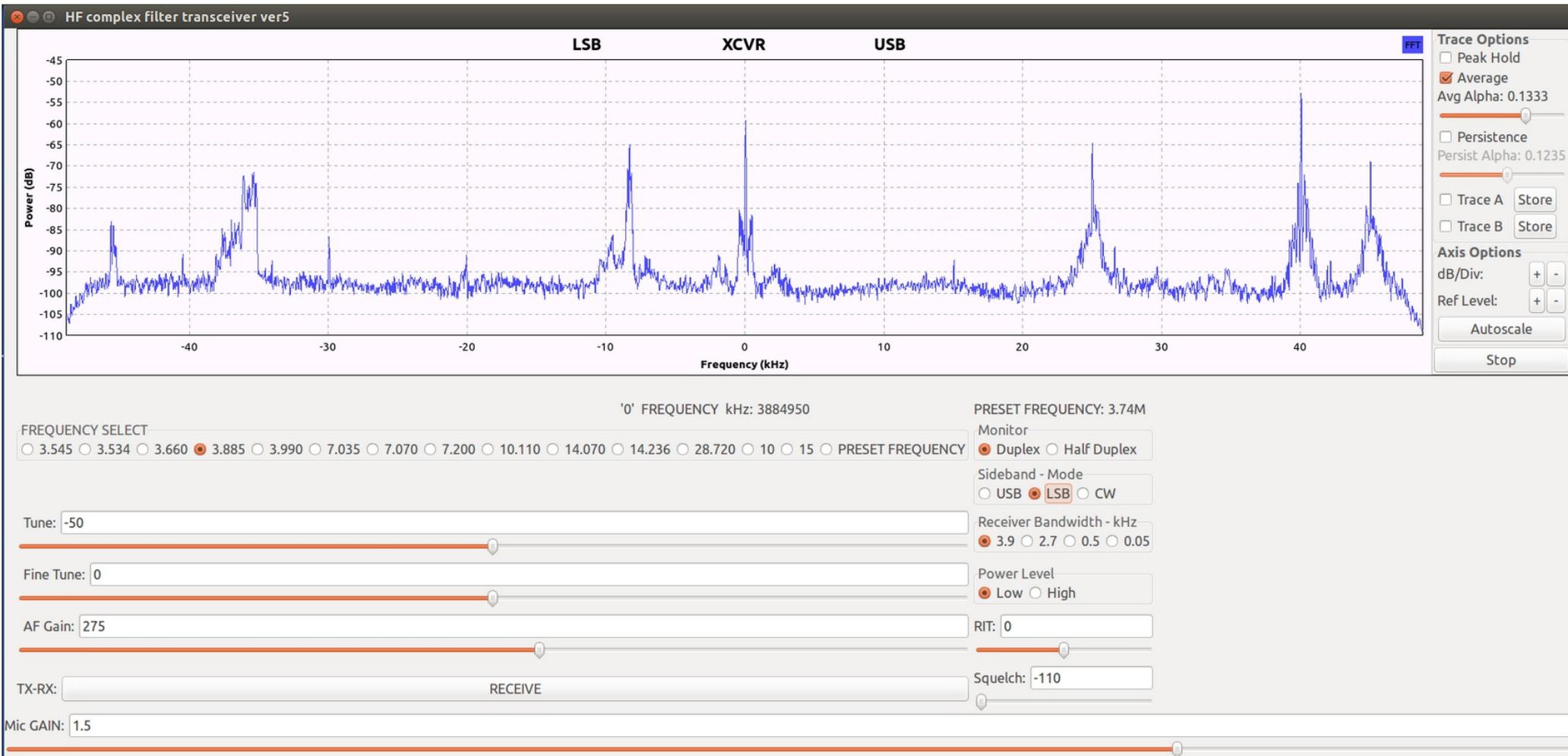






Transmitter Segment of Universal Multi-Mode Transceiver Flowgraph





Operating GUI of Universal Multi-Mode Transceiver - Receiving 75 meter LSB SSB and a single AM station



Ideales SDR für dieses Projekt: LimeSDR



[<https://limemicro.com/products/boards/limesdr/>]

Die Daten:

RF Transceiver: Lime Microsystems LMS7002M MIMO FPRF

Memory: 256 MBytes DDR2 SDRAM

USB 3.0 controller: Cypress USB 3.0 CYUSB3014-BZXC

Continuous frequency range: 100 kHz – 3.8 GHz

Bandwidth: 61.44 MHz

Power Output (CW): up to 10 dBm

Multiplexing: 2×2 MIMO

Power: micro USB connector or optional external power supply

Dimensions: 100 mm x 60 mm



Ein Problem:

- in fertigen Programmen wie SDR# oder CubicSDR sind alle Parameter schon voreingestellt.
- in GNU-Radio können bei jedem Funktionsblock alle funktionalen Parameter in weiten Grenzen eingestellt werden. Um ein Gerät zum Laufen zu bringen, muss man sich in der digitalen Signalverarbeitung einigermaßen auskennen. Natürlich gibt es im Web viele fertige GRC-Programme (.grc), mit denen man anfangen kann [<https://w7fu.com/>]



Interessante Themen aus der digitalen Signalverarbeitung:

- DFT (FFT)
- Windowing (Fensterung)
- Sample-Rate-Conversion (DDC, DUC)
- Digitale Filterung (FIR, IIR)
- Technische Parameter von Modulatoren, Demodulatoren, Mischern
- Signalaufbereitung für SSB
- Digitale Modulationsverfahren (PSK, FSK, QAM)



2.3 Betriebssysteme, Quelltext und Binaries, Programmiersprachen

Installation von Linux (empfohlen UBUNTU)

1. Linux als einziges Betriebssystem (volle Hardwarenutzung)
2. Dual-Boot-Systeme (volle Hardwarenutzung, begrenzter HD-Platz; Auswahl des Betriebssystems beim Booten)
3. Linux als virtuelle Maschine unter Windows 10 (begrenzte Performance, Probleme bei Treibern, Bedienung durch Windows-Fenster)
4. Linux als Live-System vom USB-Stick starten (einfachste Methode mit stärksten Einschränkungen, Boot-Reihenfolge ändern im Setup)



Installation von GNU-Radio (rel. einfach)

Es gibt mehrere Möglichkeiten:

1. From Binaries, „zu Fuß“
(fertig übersetzte Binärprogramme für verschiedene Betriebssysteme)
2. From Source-Code (Quellcode, meistens in C ++ geschrieben, muss zu Binaries compiliert werden)
3. From Binaries mithilfe von Skripten (vorgefertigte Befehlsabläufe, z.B. Conda)
4. From Source mithilfe von Skripten (z.B. PyBOMBS)



Installation von GNU-Radio in UBUNTU:

```
sudo add-apt-repository ppa:gnuradio/gnuradio-releases
```

```
sudo apt-get update
```

```
sudo apt-get install gnuradio
```

[<https://wiki.gnuradio.org/index.php/UbuntuInstall>]



Treiber

Ettus B200 Windows:

[[https://kb.ettus.com/Building_and_Installing_the_USRP_Open_Source_Toolchain_\(UHD_and_GNU_Radio\)_on_Windows](https://kb.ettus.com/Building_and_Installing_the_USRP_Open_Source_Toolchain_(UHD_and_GNU_Radio)_on_Windows)]

Ettus B200 Linux + Übersicht:

[<https://kb.ettus.com/B200/B210/B200mini/B205mini>]

Adalm Pluto:

[<https://wiki.analog.com/university/tools/pluto/drivers/linux>]

[<https://wiki.analog.com/university/tools/pluto/drivers/windows>]

LimeSDR:

[<https://launchpad.net/~myriadrf/+archive/ubuntu/gnuradio>]

RTL-SDR:

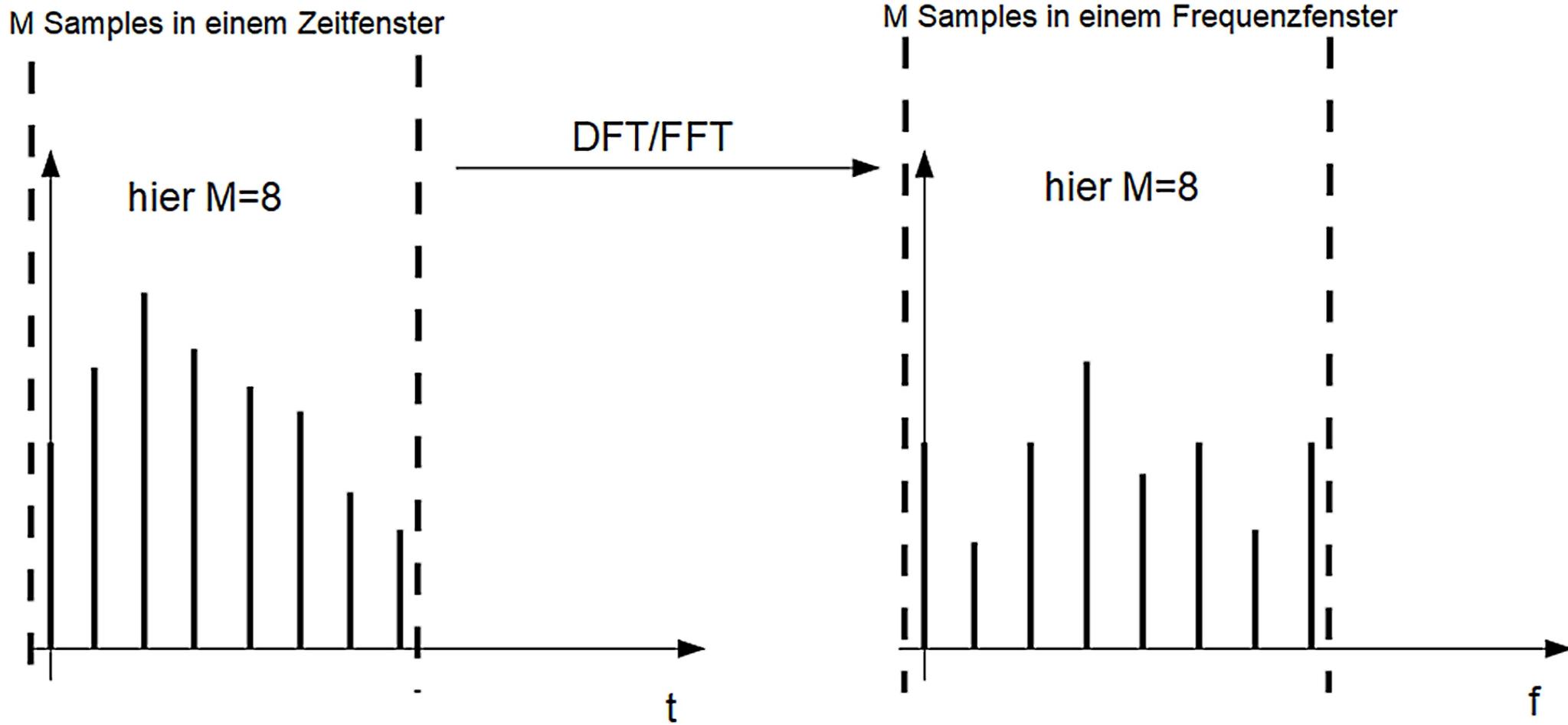
[<http://sdr.osmocom.org/trac/wiki/rtl-sdr>]

[<http://www.rtl-sdr.org>]



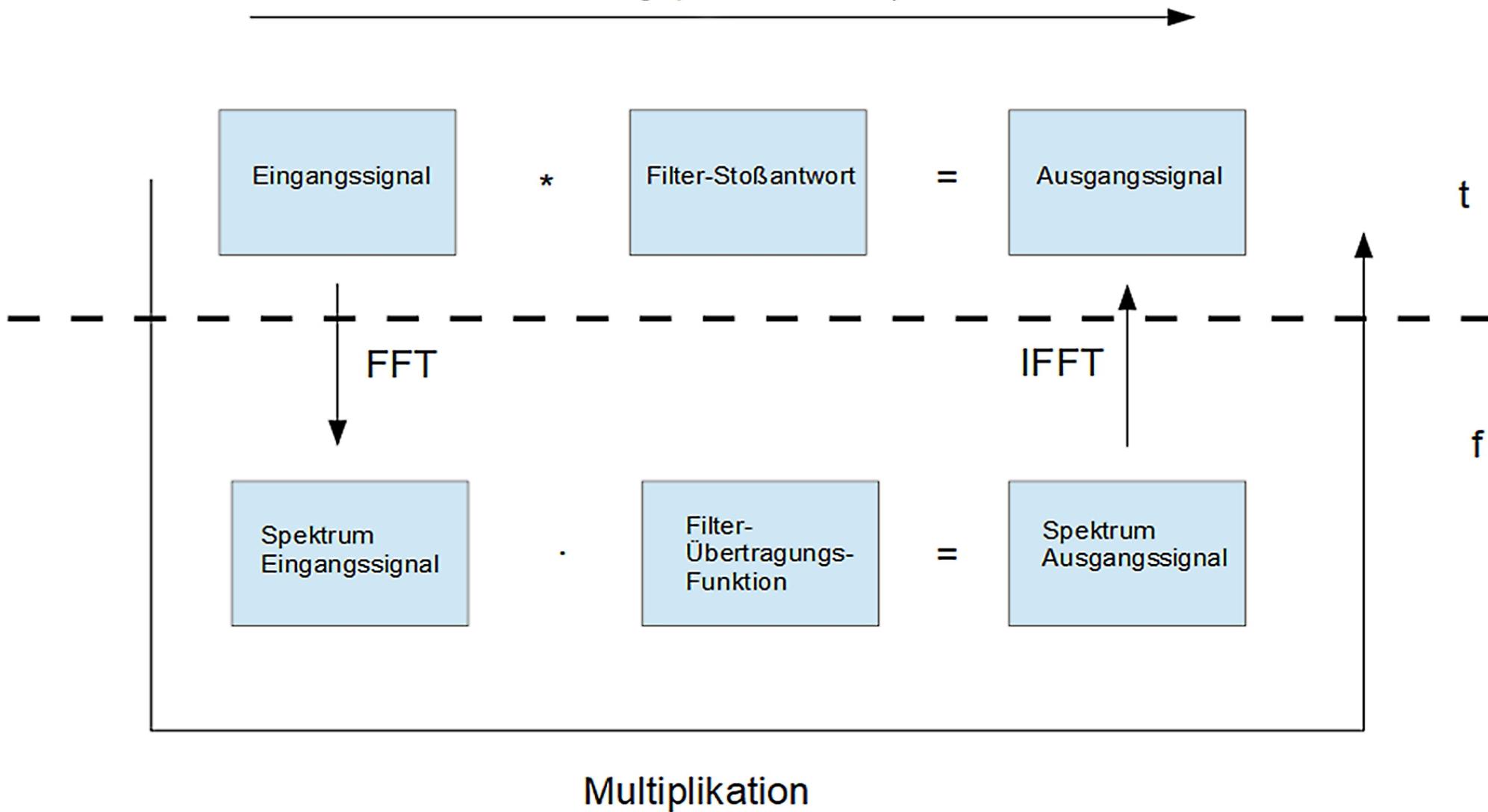
Etwas Signaltheorie zum Schluss:

Stark vereinfachte Darstellung der Diskreten Fourier-Transformation (DFT)



Filterung eines Signals:

Faltung (Convolution)



2.4 Glossar:

A/D Converter : Analog/Digital-Umsetzer, charakterisiert durch die Abtastrate und die Auflösung

Baseband: Basisband, Frequenzband um 0 Hz hinter der Mischstufe von ZF0-Empfängern

D/A Converter: Digital/Analog-Umsetzer, bei SDR im Sender

Direct Conversion: ein Verfahren bei SDR, bei dem das HF-Signal direkt ins Basisband umgesetzt wird

DDC: Digital Down Conversion, digitale Abwärtstastung durch Verringerung der Abtastrate

DSP: Digital Signal Processing, Digitale Signalverarbeitung

DUC = Digital Up Conversion, digitale Aufwärtstastung durch Erhöhen der Abtastrate durch Interpolation

FPGA: Field Programmable Gate Array. Ein Logik-Schaltkreis mit schneller Hardware. Wird mit Firmware programmiert und leistet die Daten-Vorverarbeitung in einem SDR. Er entlastet so den nachfolgenden PC.

GNU Radio: Eine Open-Source-Software mit Bibliothek-Funktionen für die digitale Signalverarbeitung



GPIO: General Purpose Input and Output, digitale Verbindung eines PC-kontrollierten Systems

GRC: GNU Radio Companion, die grafische Oberfläche von GNU-Radio

.grc: Dateiendung für ausführbare Flussdiagramme in GNU-Radio

GUI: Graphical User Interface, grafische Benutzer-Oberfläche

I and Q: In-Phase-Signal und Quadratur-Phase-Signal

NCO: Numerically Controlled Oscillator, Oszillator, dessen Frequenz vom PC gesteuert wird

Python: Programmiersprache

Quadrature: Signal, das gegenüber dem „In-Phase-Signal“ um 90° phasenverschoben ist.

RFIC: Radio Frequency Integrated Circuit, hochintegriertes analoges IC

RTL SDR: einfaches SDR



SDR: Software Defined Radio.

UHD: Universal Hardware Driver, Dateninterface zwischen USRP und SDR

USRP: Universal Software Radio Peripheral, Begriff der Firma ETTUS. Beschreibt heute eine Klasse von hochwertigen SDR, die über UHD mit einer Reihe von SDR-Software betrieben werden können.

Widget: Ein Icon, mit dem Software kontrolliert werden kann.



Quellen (vom 04. 04. 2021):

<https://wiki.gnuradio.org/index.php/Tutorials>

https://wiki.gnuradio.org/index.php/Frequency_Xlating_FIR_Filter

https://wiki.gnuradio.org/index.php/Simulation_example:_Single_Sideband_transceiver

https://wiki.gnuradio.org/index.php/Simulation_example:_Narrowband_FM_transceiver

<https://w7fu.com>

<https://files.ettus.com/tutorials/labs/>

<http://oe7wpa.com/index.php/projekte/software/sdr-software/gnu-radio/gnu-radio-beispiele>

<http://www.stargazing.net/david/gnuradio/index.html>

Reinelt, J.: HackRF One und GNU-Radio, Eine Einstiegshilfe, DARC-Buchreihe, DARC-Verlag, 2016

