

GNU Radio

GNU Radio ist eine großartige Wahl, wenn Sie sich mit softwarebasiertem Funk beschäftigen möchten. Es bietet eine effiziente Möglichkeit, Signalverarbeitungsanwendungen zu entwickeln, ohne alles von Grund auf neu programmieren zu müssen. Mit seiner umfangreichen Bibliothek an Standardalgorithmen und wiederverwendbaren Blöcken sparen Sie viel Zeit und Mühe.

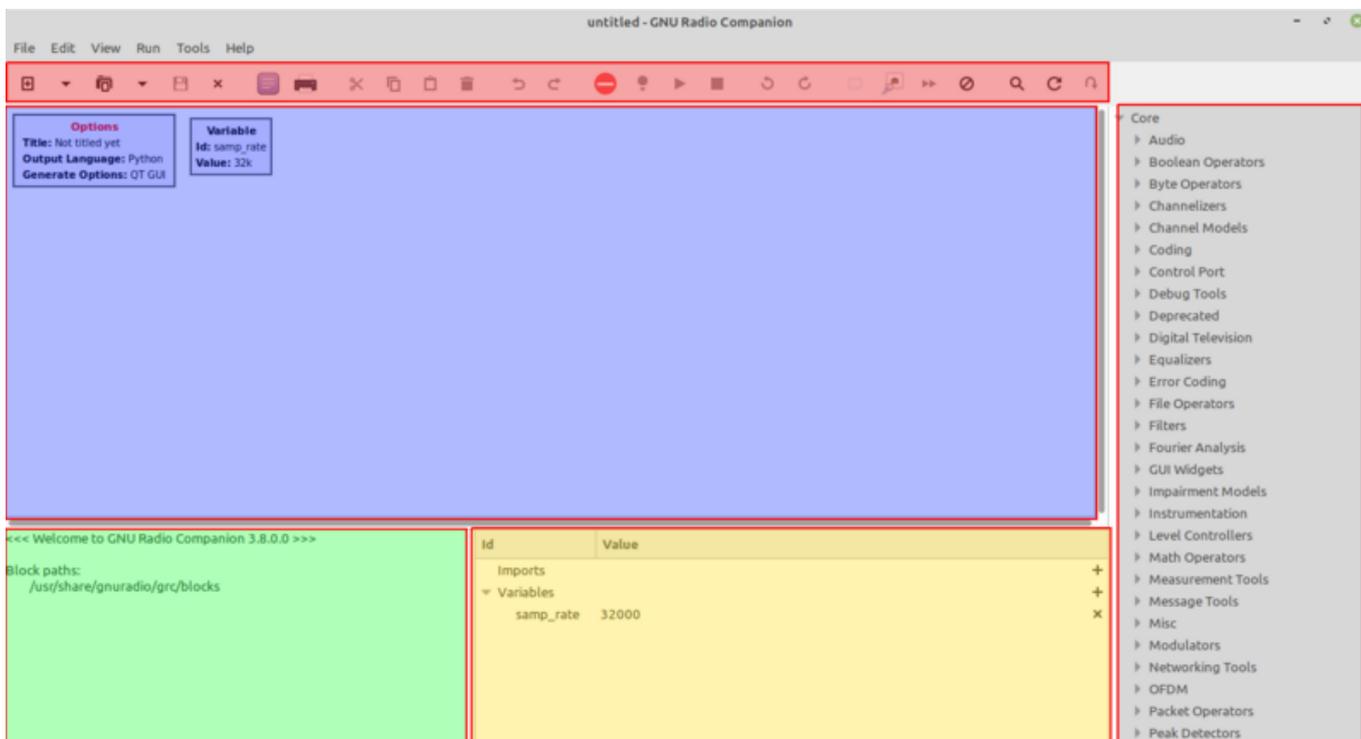
Das Framework ist nicht nur benutzerfreundlich, sondern auch sehr flexibel ¹⁾. Es läuft sowohl auf leistungsstarken Multi-Core-Systemen als auch auf energieeffizienten, eingebetteten Geräten wie z.B. dem Raspberry Pi. Die optimierten Funktionen und Beispiele erleichtern den Einstieg und helfen Ihnen, Ihre Projekte schneller umzusetzen.

Wenn Sie also nach einer Lösung suchen, die Ihnen Arbeit abnimmt, gleichzeitig aber leistungsstark und vielseitig ist, ist GNU Radio definitiv einen Blick wert!

Einführung

Wir beginnen einfach und untersuchen, wie man den GNU Radio Companion (GRC), das grafische Tool von GNU Radio, verwendet, um Radioprogramm hörbar zu machen. GRC wurde entwickelt, um die Nutzung von GNU Radio zu vereinfachen, indem es uns ermöglicht, Python-Dateien grafisch zu erstellen, anstatt sie ausschließlich im Code zu schreiben.

Die Benutzeroberfläche ist so gestaltet, dass sie eine intuitive und effiziente Navigation ermöglicht. Sie besteht aus fünf Teilen: Bibliothek, Werkzeugleiste, Terminal, Arbeitsbereich und Variablen.



Eine GNU Radio Companion (GRC) Programmdatei besteht aus zwei wesentlichen Komponenten: einer grc-Datei und einer py-Datei. Die grc-Datei enthält die Konfigurations- und Einstellungselemente eines GRC-Projekts, welche in der grafischen Benutzeroberfläche von GNU Radio Companion bearbeitet

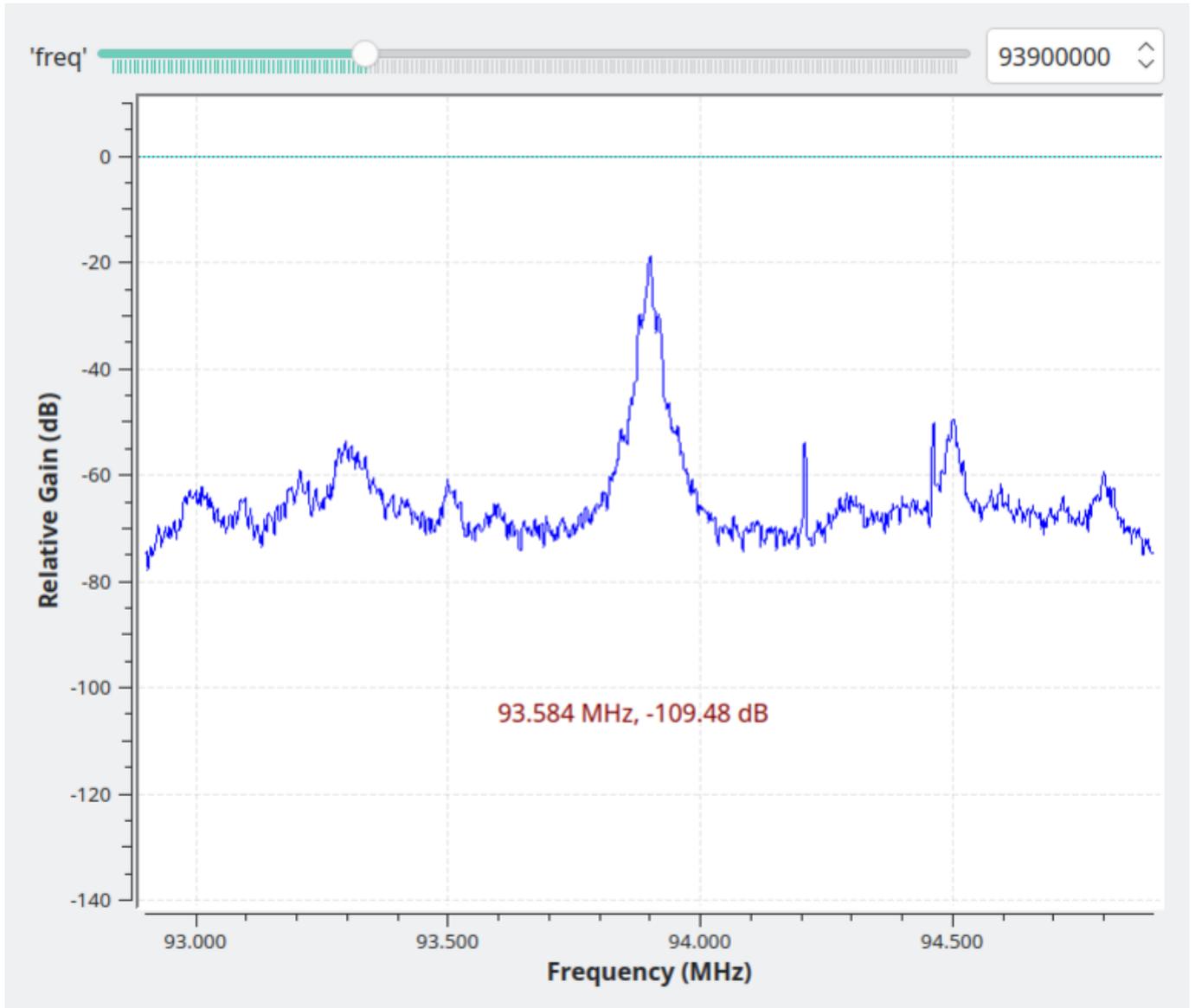
werden können. Diese Datei definiert die Struktur und die Parameter des Signalverarbeitungssystems.

Die py-Datei hingegen ist der automatisch generierte Python-Code, der auf Basis der in der grc-Datei definierten Einstellungen erstellt wird. Dieser Code kann unabhängig von der GRC-Oberfläche ausgeführt werden und stellt die eigentliche Implementierung des Signalverarbeitungsprojekts dar. Dies ermöglicht es, Projekte direkt in Python zu nutzen oder weiter anzupassen, falls dies erforderlich ist.

Beide Dateien sind essenziell für die Arbeit mit GNU Radio Companion: Die grc-Datei dient zur einfachen visuellen Entwicklung und Anpassung, während die py-Datei die technische Umsetzung und Ausführung übernimmt.

Hello World

Ein 'Hello World'-Programm dient in jeder Programmiersprache als grundlegende Einführung, um die Basisfunktionen zu verstehen. Im Kontext von GNU Radio entspricht dies einem einfachen FM-Radioempfänger, der die grundlegenden Prinzipien des Systems demonstriert. Mit diesem Beispiel ²⁾ können Sie die Signalverarbeitungskette nachvollziehen, von der Eingabe über die Filterung bis hin zur Ausgabe. Es bietet eine ideale Grundlage, um sich mit der Benutzeroberfläche und den Kernkonzepten von GNU Radio vertraut zu machen.



The screenshot displays the GNU Radio GUI. On the left, there are control panels for 'Options' (Title: Hello World, Output Language: Python, Generate Options: QT GUI) and 'Variable' (ID: samp_rate, Value: 2M). The main area shows a flow graph with the following components:

- RTL-SDR Source** (ID: rtl_source_0): Parameters include Sample Rate (2M), Frequency (93.9M), and various gain/balance settings.
- QT GUI Frequency Sink** (ID: qtgui_freq_sink_x_0): Parameters include FFT Size (1024), Center Frequency (93.9M), and Bandwidth (2M).
- WBFM Receive** (ID: analog_wbfm_rcv_0): Parameters include Quadrature Rate (2M) and Audio Decimation (1).
- Rational Resampler** (ID: rational_resampler_xxx_0): Parameters include Interpolation (3), Decimation (125), and Fractional BW (0).
- Audio Sink** (ID: audio_sink_0): Parameter is Sample Rate (48 kHz).

Below the flow graph is a terminal window showing the execution of the program:

```

Executing: /usr/bin/python3 -u /home/cs/
Dokumente/work/SDR/GNU Radio/hello_world.py
gr-osmosdr 0.2.0.0 (0.2.0) gnuradio 3.10.12.0
built-in source types: file rtl_tcp uhd hackrf
bladerf rfspace airspy soapy redpitaya
[INFO] [UHD] linux; GNU C++ version 14.2.1 20250207;
Boost_108700; DPK_24.11; UHD_4.8.0.makepkg-0-
g308126a4
Found Rafael Micro R828D tuner
RTL-SDR Blog V4 Detected
Using device #0 RTLSDRBlog Blog V4 SN: 00000001
Found Rafael Micro R828D tuner
RTL-SDR Blog V4 Detected
Exact sample rate is: 2000000.052982 Hz

>>> Done

Generating: "/home/cs/Dokumente/work/SDR/GNU
Radio/hello_world.py"
    
```

On the right side, a component palette is visible, with 'GUI Widgets' highlighted in green.

Die beiden Dateien, **hello_world.grc** und **hello_world.py**, sind in der bereitgestellten ZIP-Datei enthalten. Bitte laden Sie die ZIP-Datei herunter und entpacken Sie diese, um Zugriff auf die Dateien zu erhalten. [hello_world.zip](#)

Das hört sich dann beispielsweise so an:

previous play next stop
 mute max volume

repeat shuffle

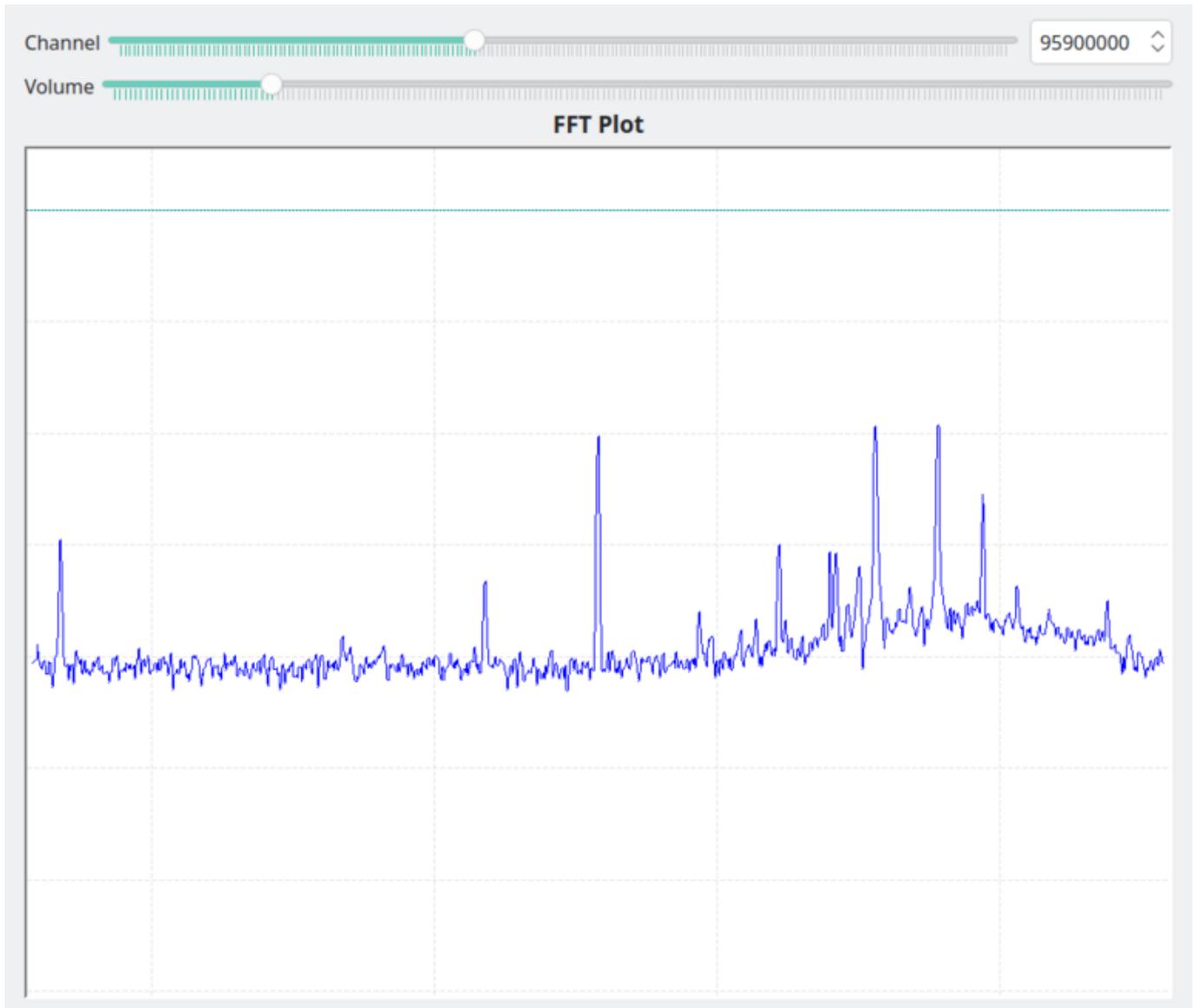
-

Update Required To play the media you will need to either update your browser to a recent version or update your [Flash plugin](#).

Das ist die einfachste Version. Den „Vollausbau“ zeigt da folgende Kapitel.

Hello World fortgeschritten

Dieses Beispiel stammt von [Michael Ossmann: Software Defined Radio with HackRF](#)



Last

update: [digitalessdr:sdr_software:sdr_tools:gnuradio](https://gatenero.duckdns.org/digitalessdr:sdr_software:sdr_tools:gnuradio) https://gatenero.duckdns.org/digitalessdr:sdr_software:sdr_tools:gnuradio
08.05.2025

The screenshot displays the GNU Radio GUI with a signal processing flowchart and a terminal window. The flowchart includes blocks for 'Signal Source', 'QT GUI Range', 'QT GUI Frequency Sink', 'Low Pass Filter', 'Rational Resampler', 'NBFM Receive', 'Multiply Const', and 'Audio Sink'. The terminal window shows the following output:

```
Loading: "/home/cs/Dokumente/work/SDR/SDR_with_HackRF/lesson1_channel_slider.grc"
>>> Done

Generating: "/home/cs/Dokumente/work/SDR/SDR_with_HackRF/lesson1_channel_slider.py"

Executing: /usr/bin/python3 -u /home/cs/Dokumente/work/SDR/SDR_with_HackRF/lesson1_channel_slider.py

gr-osmosdr 0.2.0.0 (0.2.0) gnuradio 3.10.12.0
built-in source types: file rtl_tcp uhd hackrf
bladerf rfspice airspy soapy redpitaya
[INFO] [UHD] linux; GNU C++ version 14.2.1 20250207;
Boost_108700; DPK_24.11; UHD_4.8.0.makepkg-0-g308126a4
Using HackRF One with firmware 2024.02.1

>>> Done
```

ID	Value
audio_gain	0.3
center_freq	97900000.0
channel_fre	95900000.0
channel_wi	200000.0
samp_rate	2000000.0

Die beiden Dateien, **hello_world_fotgeschritten.grc** und **hello_world_fotgeschritten.py**, sind in der bereitgestellten ZIP-Datei enthalten. Bitte laden Sie die ZIP-Datei herunter und entpacken Sie diese, um Zugriff auf die Dateien zu erhalten. [hello_world_fortgeschritten.zip](#)

1)

GNU/Linux, Microsoft Windows, BSD-Betriebssystem, macOS, Unix-ähnliches System

2)

von [Dave Rowntree: Hacking the Radio Spectrum with GNU Radio](#)

From:

https://gatenero.duckdns.org/digitalessdr:sdr_software:sdr_tools:gnuradio - **Digitalessdr**

Permanent link:

https://gatenero.duckdns.org/digitalessdr:sdr_software:sdr_tools:gnuradio

Last update: **08.05.2025**

