GNU Radio Basics
&
Interfacing to Aselsan COMINT Receivers

Murat Sever

Kasım, 2018

Outline

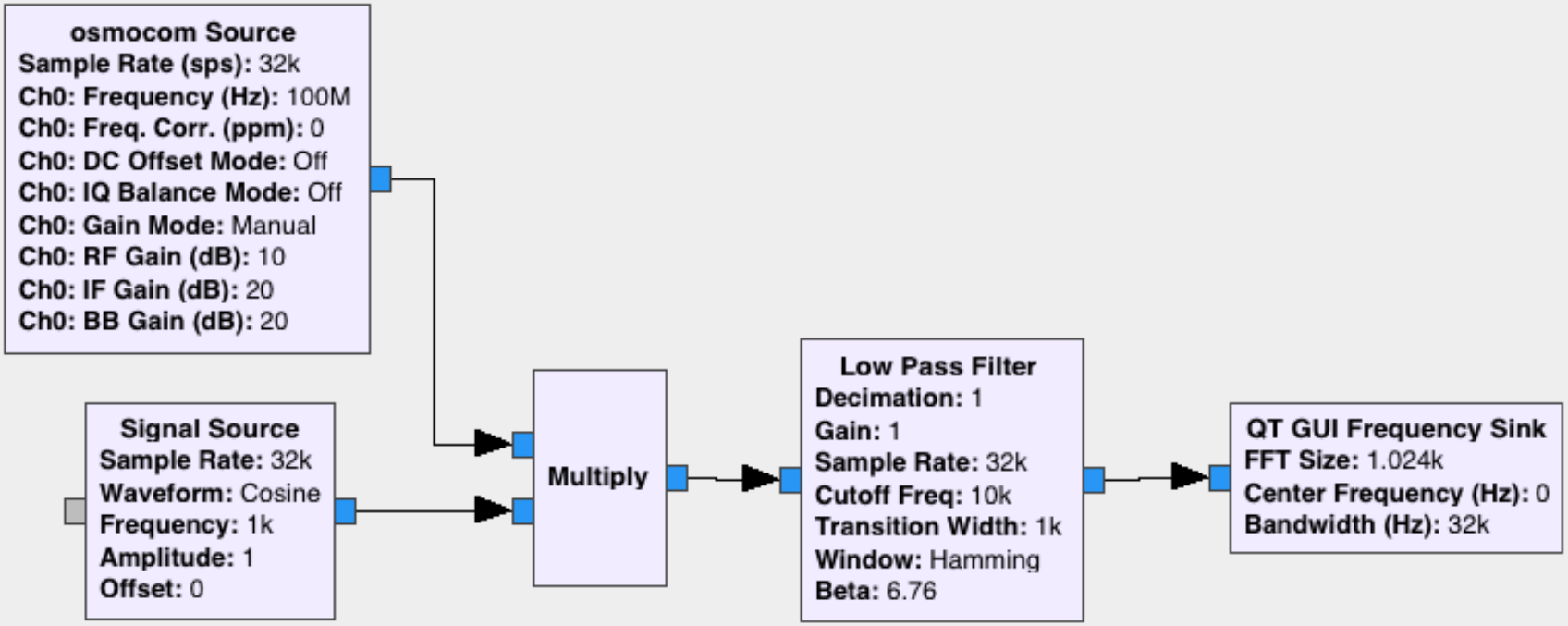
- GNU Radio 101
- GNU Radio Interface to Aselsan COMINT Receivers
- GNU Radio Live Demos
- Future Work

GNU Radio is…

- A signal processing library
- Designed for real-time
- The software part of an SDR
- Not a radio application
- The tool to **build your own** transceivers
- **FOSS**: Free and Open Source Software

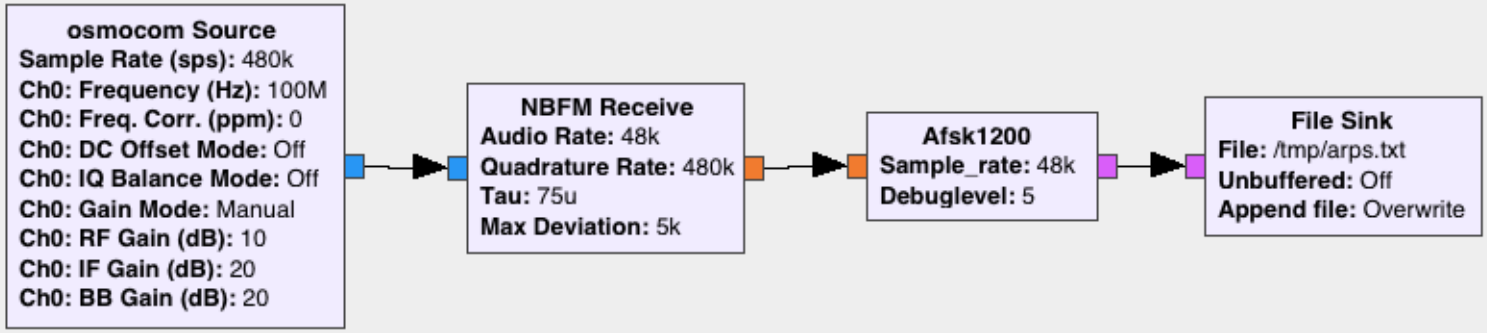
Basic Concept: Flow Graph

- Transceivers are implemented as *flow graphs*
- Similar to Simulink / schematics
- Define structure and parameters of *blocks*



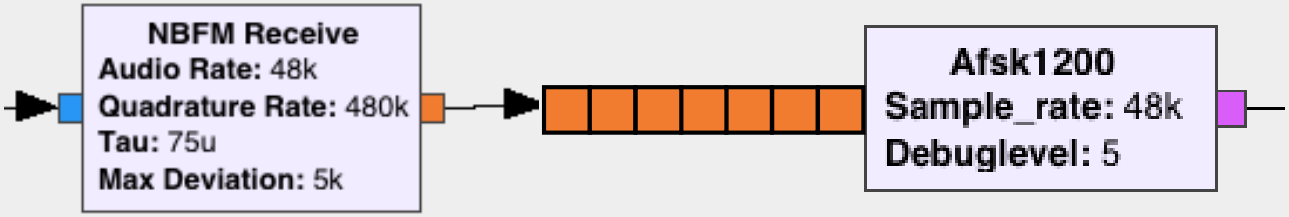
Basic Concept: Block

- Written in C++ or Python
- Implement one logical step
- Each block run in separate thread
- Can be encapsulated into hierarchical blocks

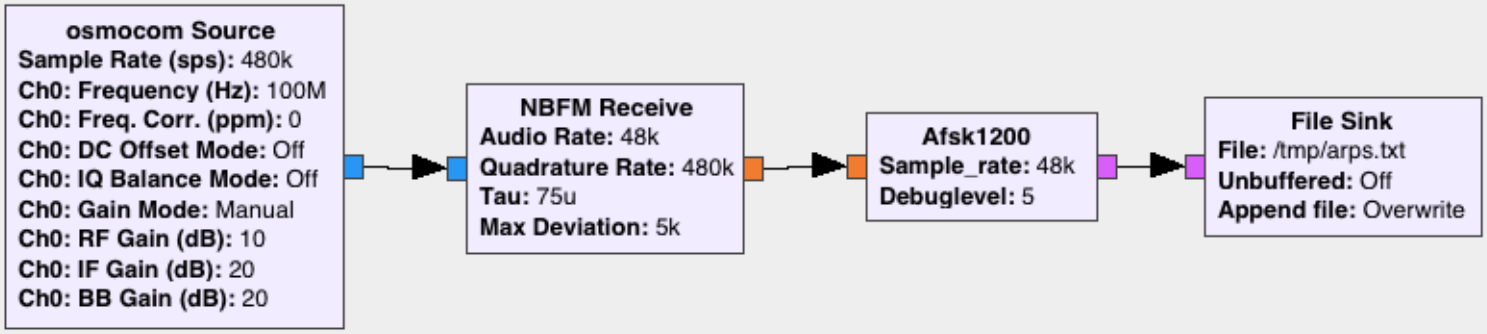


Data Streams

- Samples are buffered

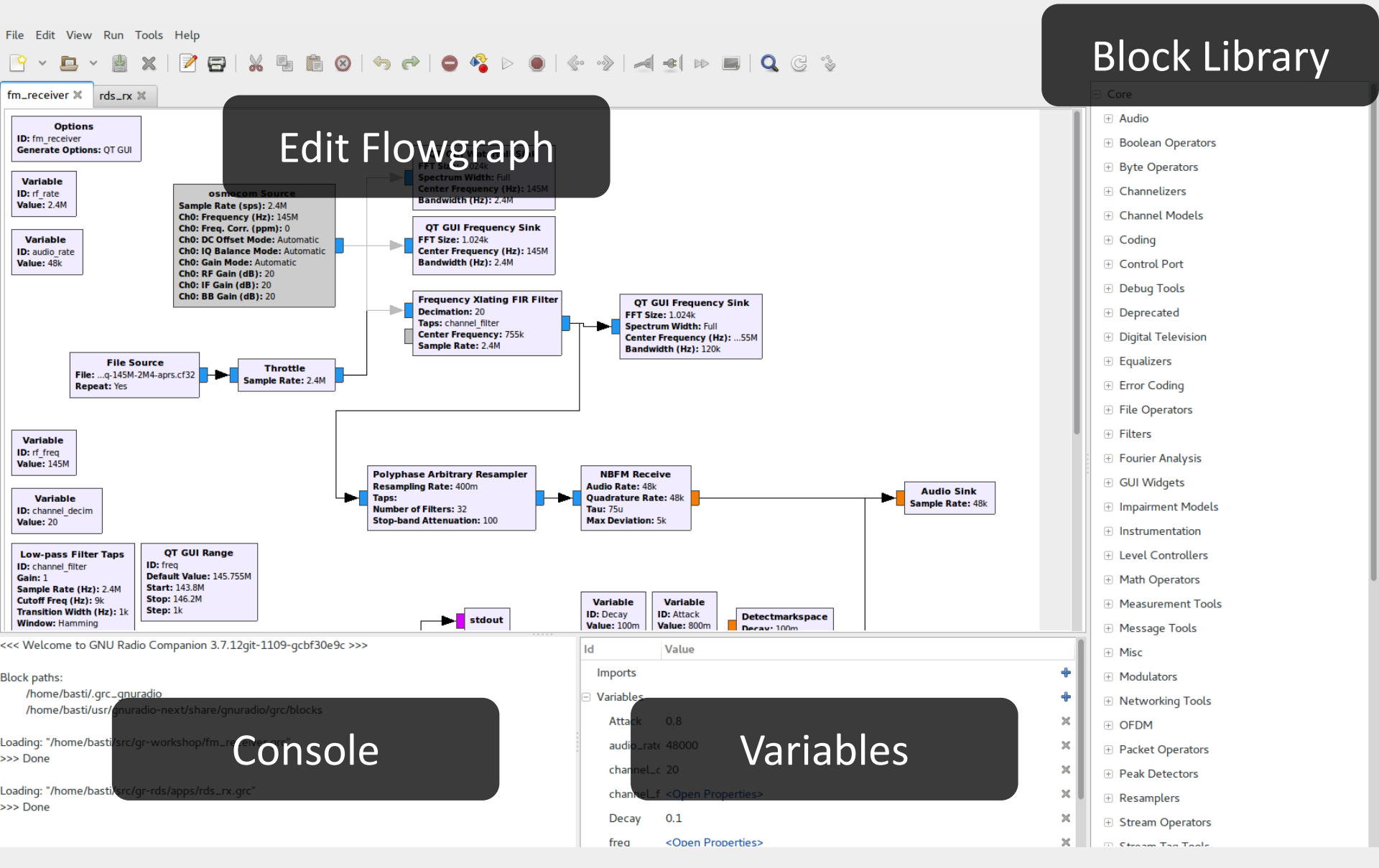


- Data types are color-coded

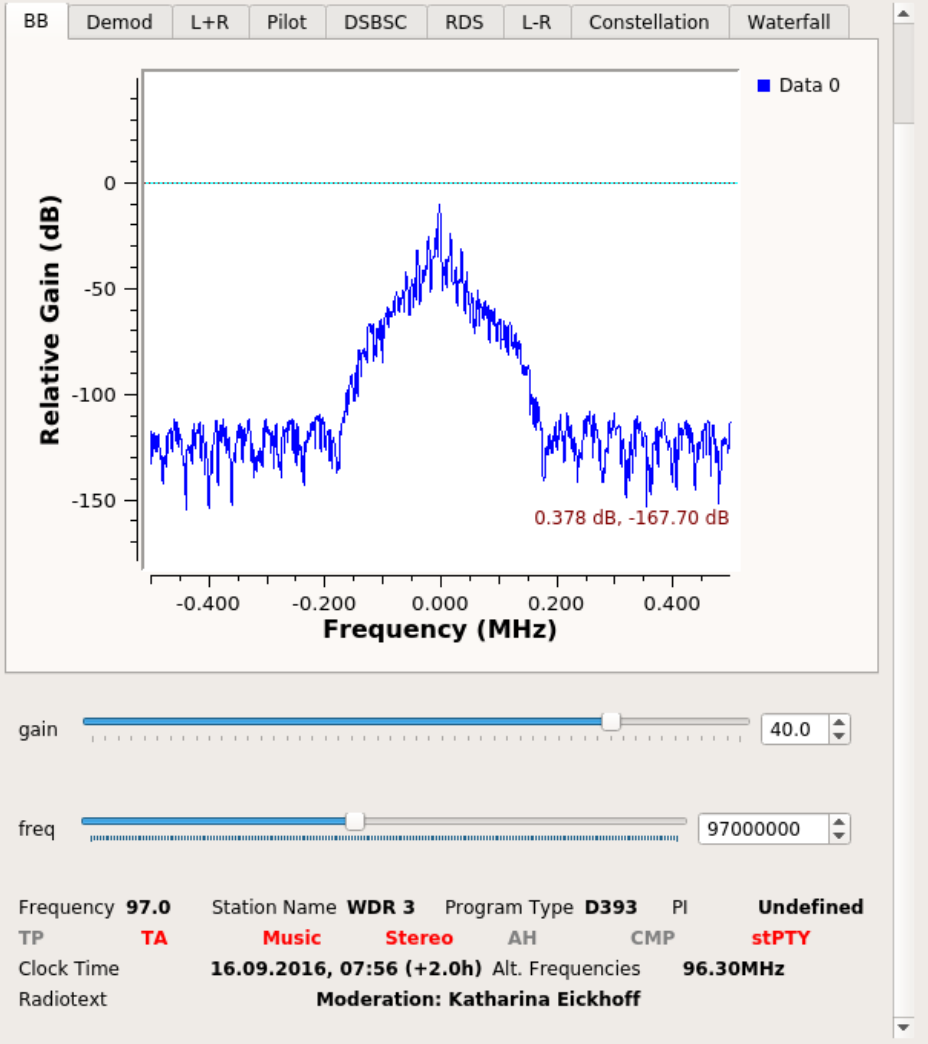


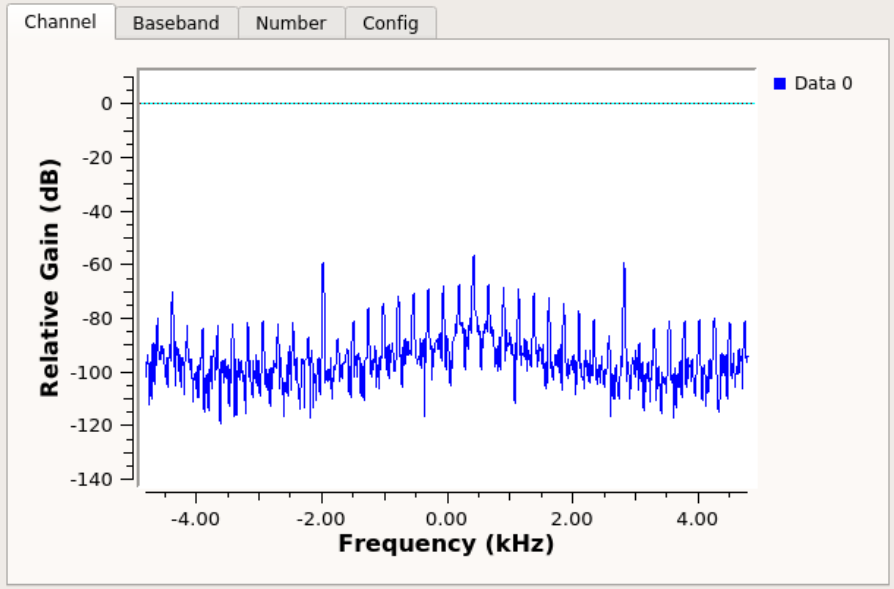
Complex Float Byte

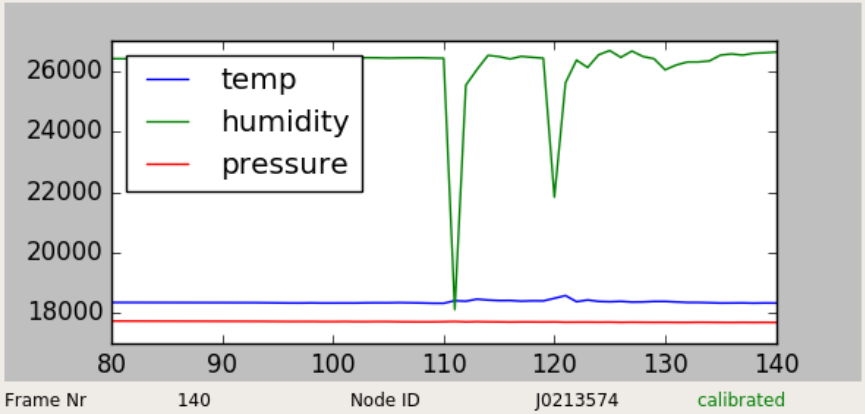
GNU Radio Companion



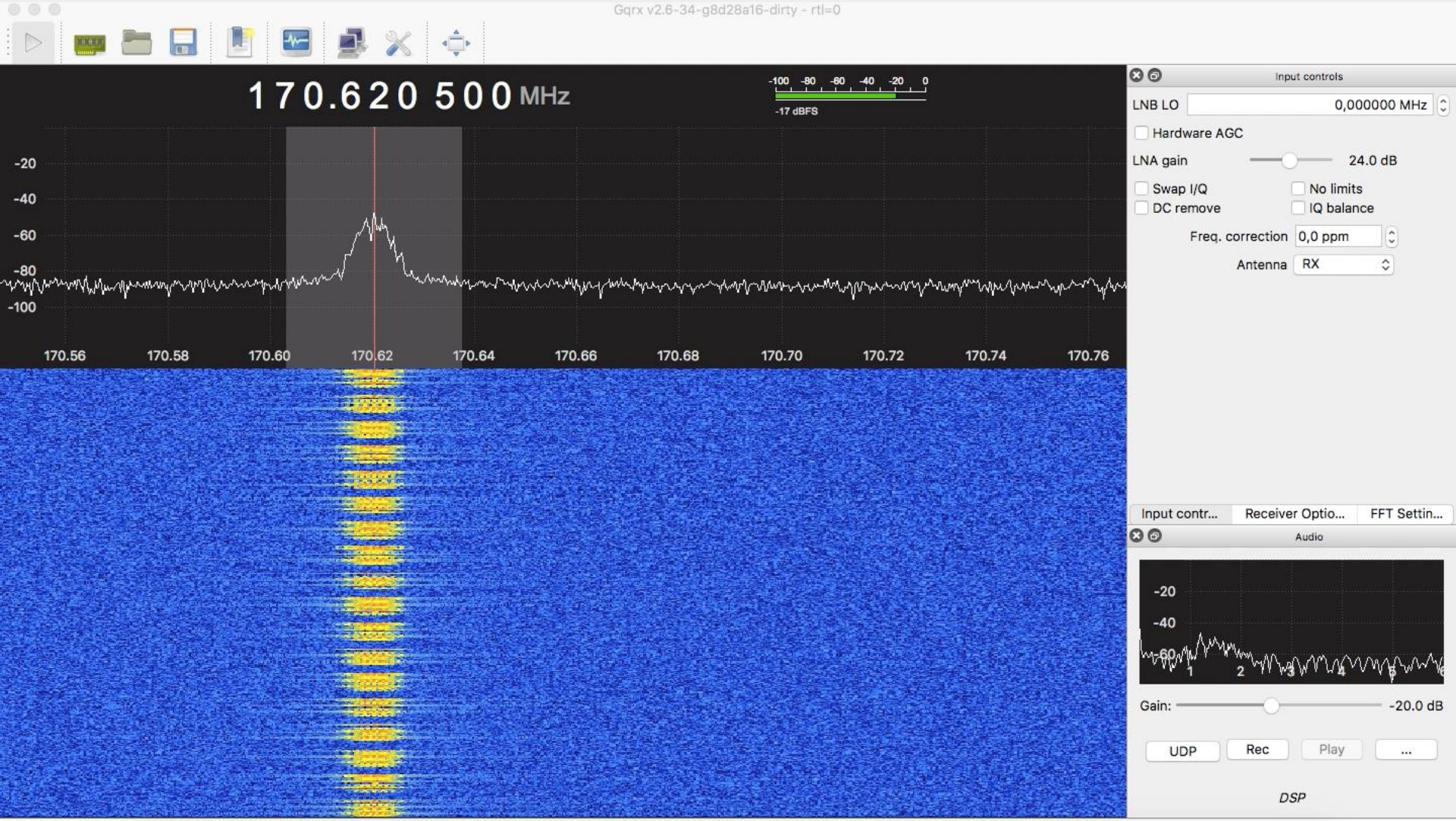
GUI Output and Instrumentation





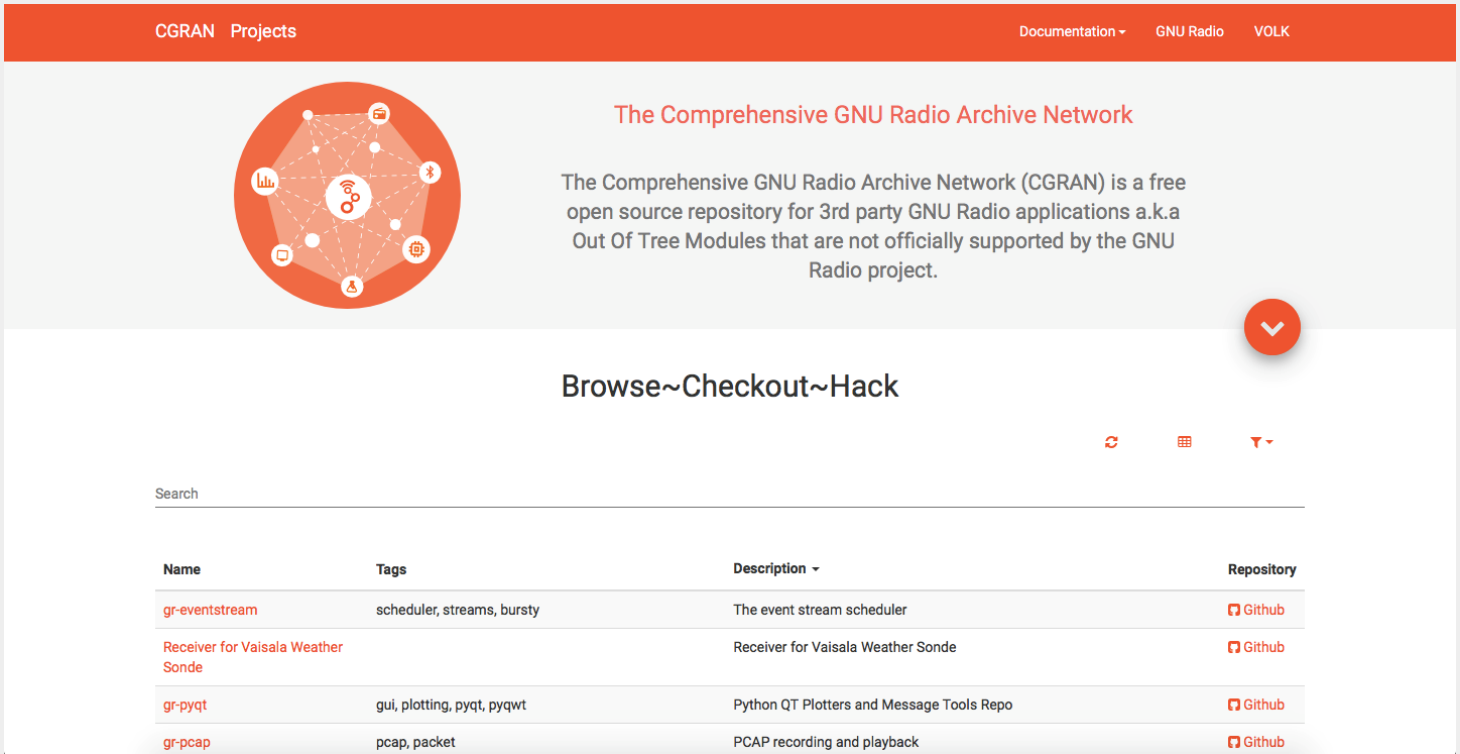


GQRX - a GNU Radio Application

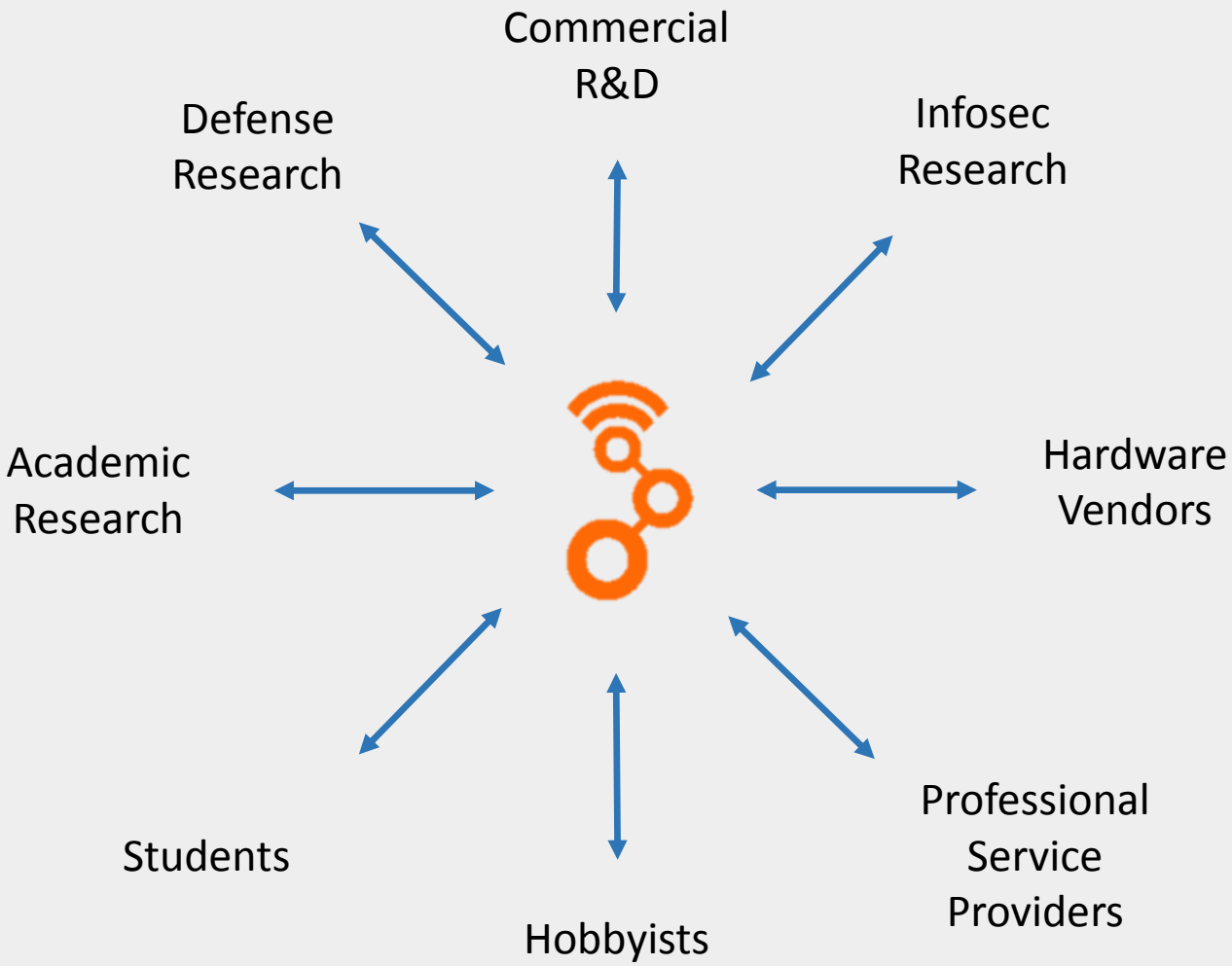


Out Of Tree Modules

- GNU Radio can be extended with OOTs
- OOTs cover more specific functionality
- There is a large number available
- CGRAN is our central database



GNU Radio is used by



GNU Radio is an Ecosystem

- Active Open Source community since 2001
- PyBombs, OOTs
- GRCon since 2011
- GNU Radio Foundation
- FOSDEM SDR DevRoom
- GSoC, SoCIS, R&S Competition, SDR Academy
- GNU Radio Europe

Installing GNU Radio

- From source: GitHub
- Package manager
  - Some are pretty outdated
- macOS / homebrew
- **PyBombs**: integrated dependency management and 3rd party application installation

Learn // Discuss // Connect

- Website: www.gnuradio.org
- Development: github.com/gnuradio
- **Mailing List: discuss-gnuradio@gnu.org**
- Wiki: wiki.gnuradio.org
- Slack: slack.gnuradio.org
- Facebook: gnuradioproject
- Twitter: @gnuradio

It all began…

- New hardware access library written for HF-ED Project
- No working GUI available at that time
- Need to verify FFT and IQ data
- Make sure receiver is configured right
  - center frequency, bandwidth…
- Visualize data

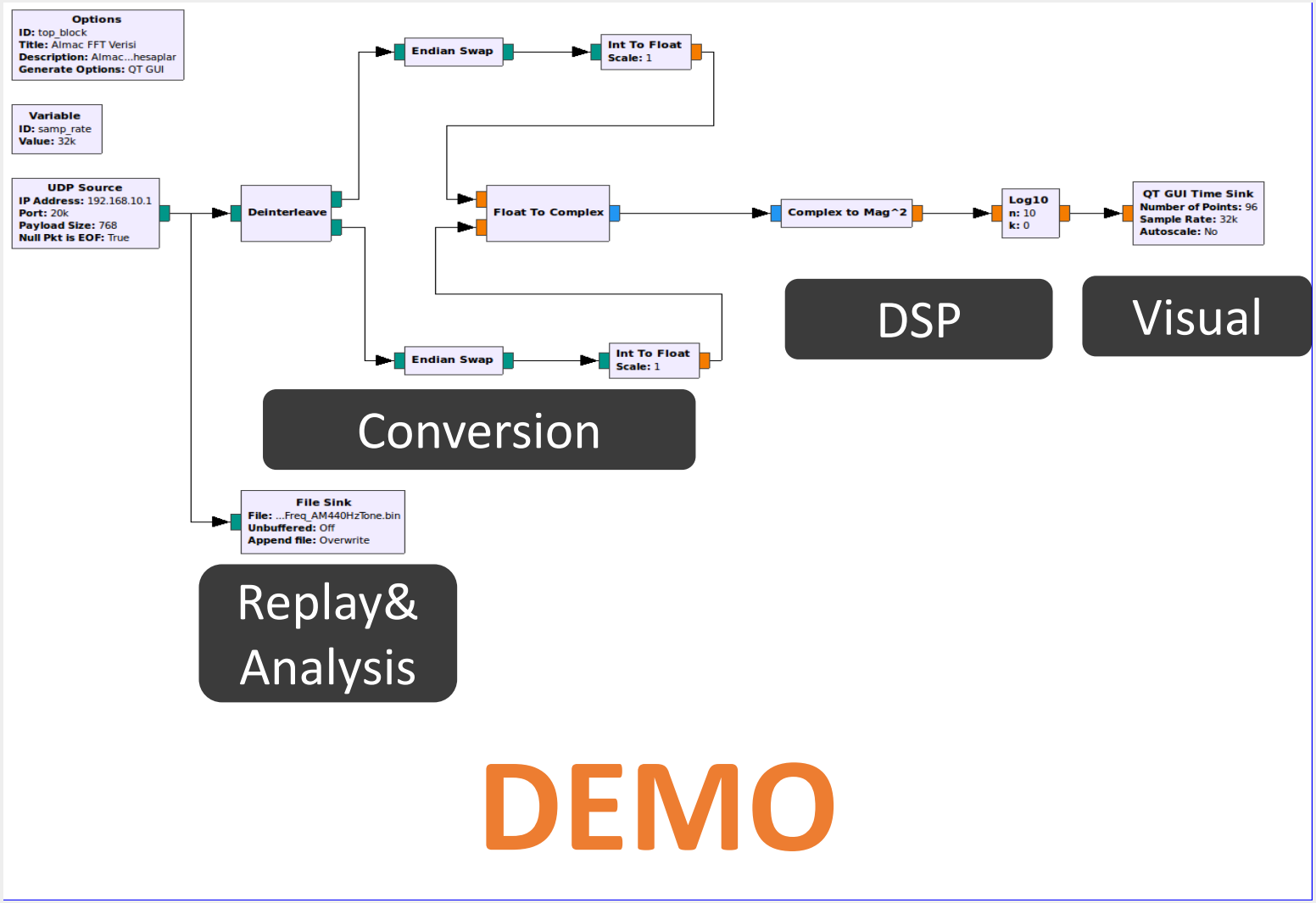
GNU Radio Interface to Aselsan Comint Receivers

- Aselsan Comint Receivers
  - Two main data streams
    - FFT: Upon request
    - IQ: Continuous
  - They are both in
    - Complex
    - Int32
    - Big Endian
    - Split form (like IIII…IQQQ…Q)
- GNU Radio expects in complex, float32 and interleaved ☹

GNU Radio Demo: FFT

- Receiver tuned to 10MHz center frequency, 25k bandwidth
- 440Hz AM modulated signal is supplied by signal generator
- We request 1024-pt FFT every second

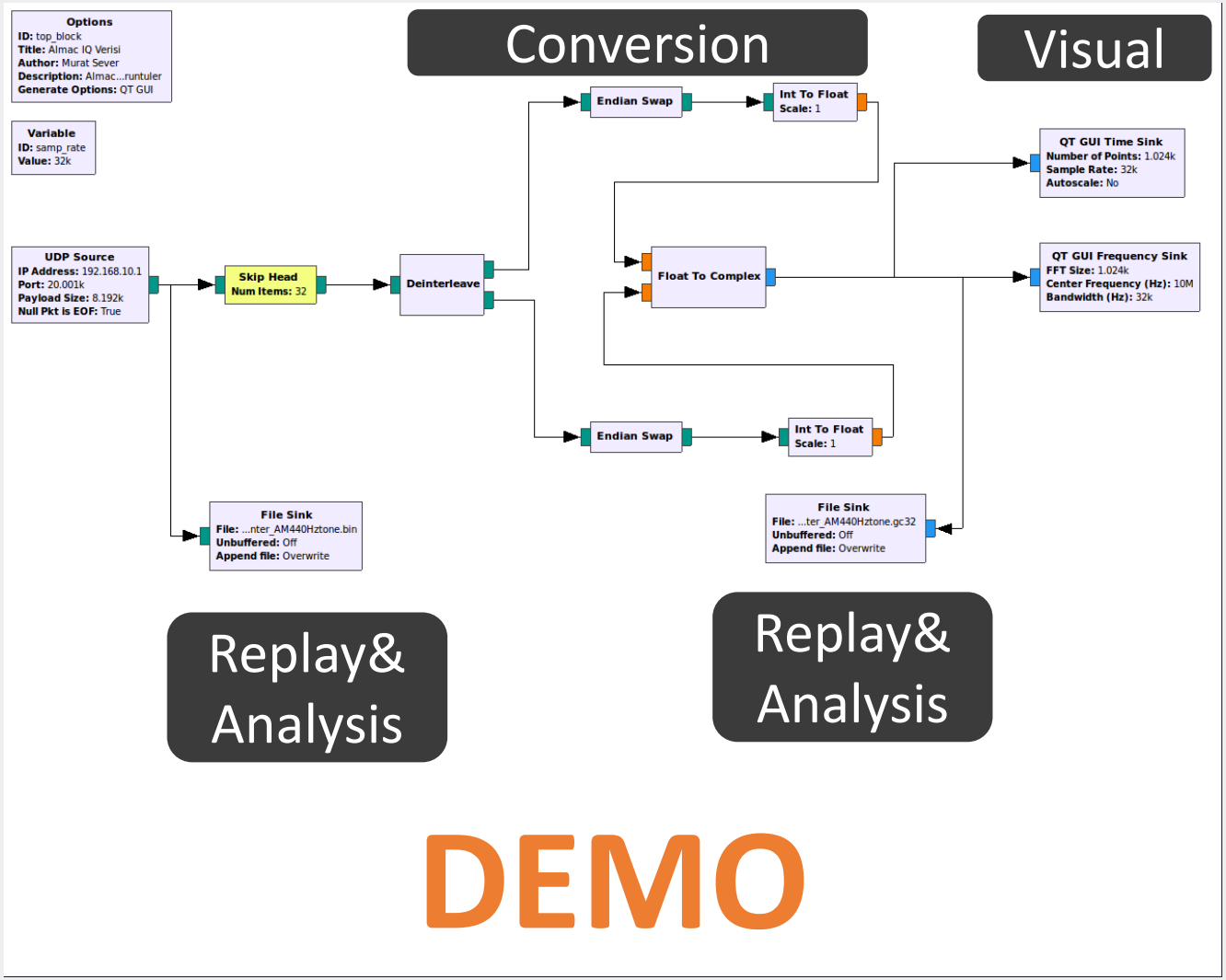
GNU Radio FFT



GNU Radio Demo: IQ

- Receiver tuned to 10MHz center frequency, 25k bandwidth
- We receive continuous IQ data and **calculate and display real-time 1024-pt FFT**
- Records given to System Eng. for offline analysis

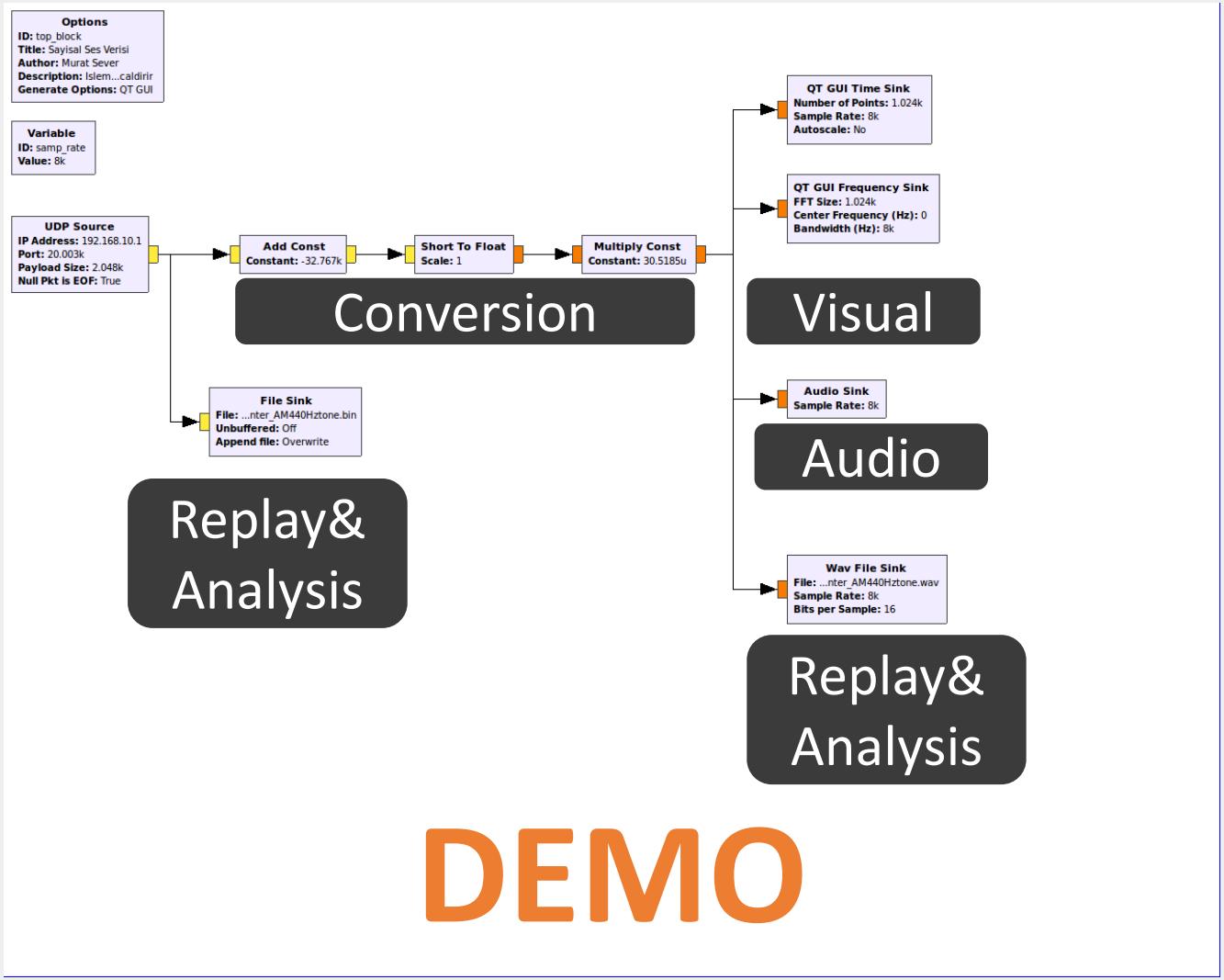
GNU Radio IQ


Options
ID: top_block
Title: Almac IQ Verisi
Author: Murat Sever
Description: Almac...runtuler
Generate Options: QT GUI
Variable
ID: samp_rate
Value: 32k
UDP Source
IP Address: 192.168.10.1
Port: 20.001k
Payload Size: 8.192k
Null Pkt is EOF: True
Skip Head
Num Items: 32
Deinterleave
Endian Swap
Int To Float
Scale: 1
Float To Complex
QT GUI Time Sink
Number of Points: 1.024k
Sample Rate: 32k
Autoscale: No
QT GUI Frequency Sink
FFT Size: 1.024k
Center Frequency (Hz): 10M
Bandwidth (Hz): 32k
Endian Swap
Int To Float
Scale: 1
File Sink
File: ...nter_AM440Hztone.bin
Unbuffered: Off
Append file: Overwrite
File Sink
File: ...ter_AM440Hztone.gc32
Unbuffered: Off
Append file: Overwrite
Conversion
Visual
Replay& Analysis
Replay& Analysis
DEMO

Later demod blocks have arrived…

- Later, demod blocks have been added to the Project
- Still need to verify they work as expected
- But again no GUI or player available to listen to the digital audio stream (DAUD)
- GNU radio to the rescue ☺

GNU Radio DAUD



Future Work

- Aselsan Source block need to be written
- Multi-channel data can be received
- Phase measurements can be made online
- Easy data recording
- All COMINT algorithms can be implemented and prototyped
- Digital demodulation blocks ready to use
- Key to other FOSS apps
  - Inspectrum, gr-inspector, …

Thanks