BSRC REU GNU Radio tutorials

Introduction

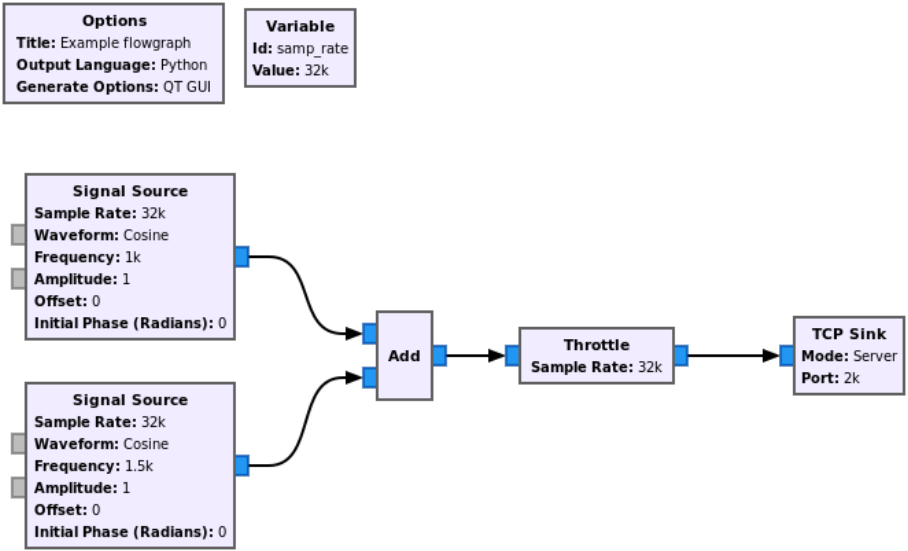
Daniel Estévez

26 June 2025

GNU Radio

- GNU Radio is an open source framework to do DSP for radio (communications, RADAR, radio astronomy...). Also useful for applications that do similar computing (even particle accelerators!).
- It comes with:
  - A GUI application called “GNU Radio companion” where systems can be implemented by dragging and dropping blocks onto a canvas (making a “flowgraph”)
  - A rich library of processing blocks accessible both through GNU Radio companion, and C++ and Python APIs
  - A “runtime”, that moves data between these blocks and runs the code of each block
- In the GNU Radio ecosystem there are out-of-tree modules, which implement new blocks that don’t fit into or exist in the in-tree library
- There are also full applications that use GNU Radio for their DSP (for instance, GQRX, or QRadioLink)

Making flowgraphs with GNU Radio companion



Making flowgraphs in Python

```
class example_flowgraph(gr.top_block):
    def __init__(self):
        gr.top_block.__init__(self, "Example flowgraph", catch_exceptions=True)
        # Variables
        self.samp_rate = samp_rate = 32000

        # Blocks
        self.network_tcp_sink_0 = network.tcp_sink(gr.sizeof_gr_complex, 1, '127.0.0.1', 2000,2)
        self.blocks_throttle_0 = blocks.throttle(gr.sizeof_gr_complex*1, samp_rate,True)
        self.blocks_add_xx_0 = blocks.add_vcc(1)
        self.analog_sig_source_x_1 = analog.sig_source_c(samp_rate, analog.GR_COS_WAVE, 1500, 1, 0, 0)
        self.analog_sig_source_x_0 = analog.sig_source_c(samp_rate, analog.GR_COS_WAVE, 1000, 1, 0, 0)

        # Connections
        self.connect((self.analog_sig_source_x_0, 0), (self.blocks_add_xx_0, 0))
        self.connect((self.analog_sig_source_x_1, 0), (self.blocks_add_xx_0, 1))
        self.connect((self.blocks_add_xx_0, 0), (self.blocks_throttle_0, 0))
        self.connect((self.blocks_throttle_0, 0), (self.network_tcp_sink_0, 0))

def main():
    tb = example_flowgraph()
    tb.start()
    tb.wait()
```

Using GNU Radio in SETI and radio astronomy

Here are some applications in which GNU Radio is useful (or can be useful):

- Full backend for small radio telescopes
- RFI analysis and simulation
- Simulation of technosignatures and other signals of interest
- Prototyping signal processing algorithms
- Teaching signal processing