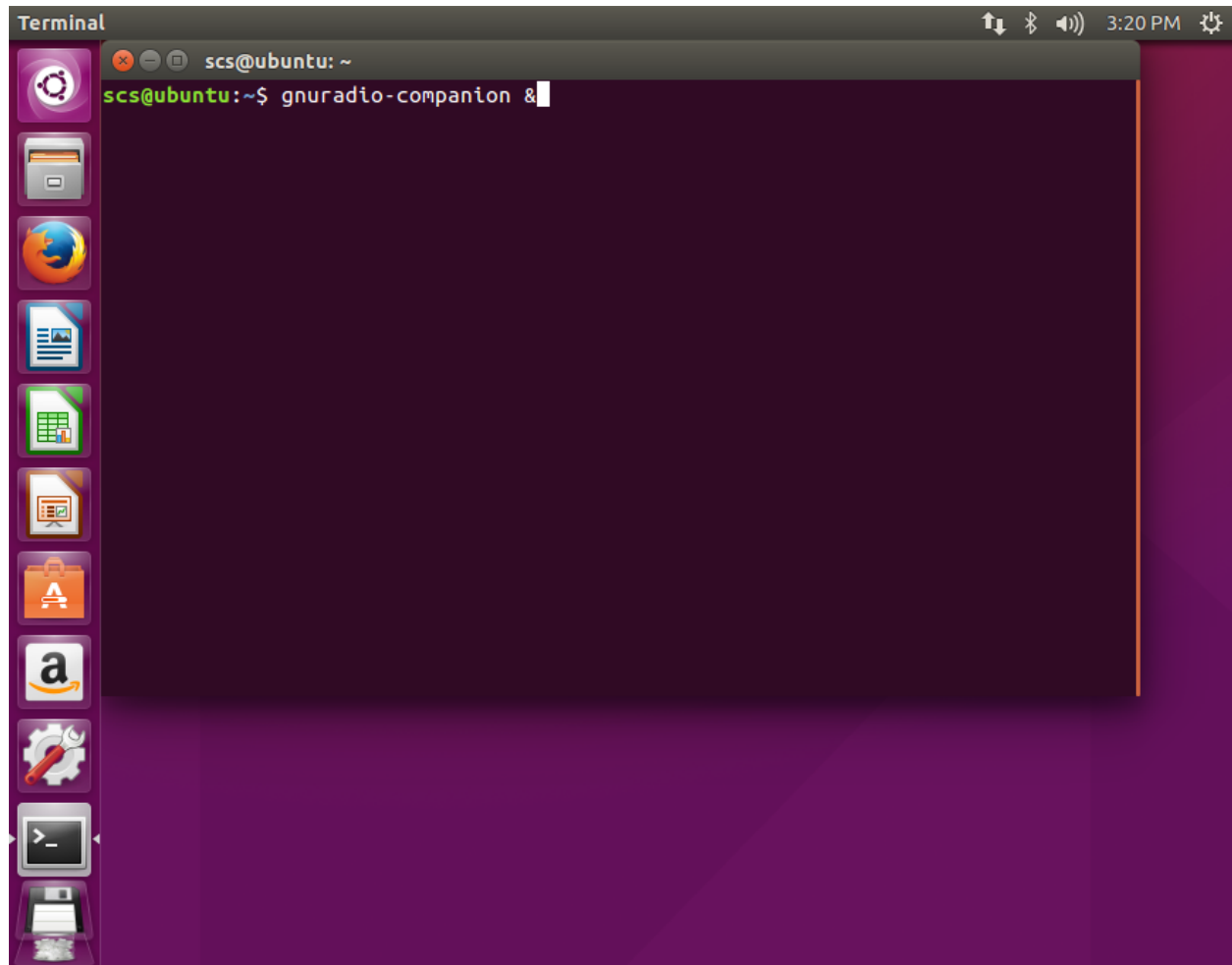


# Running the GNU Radio Example

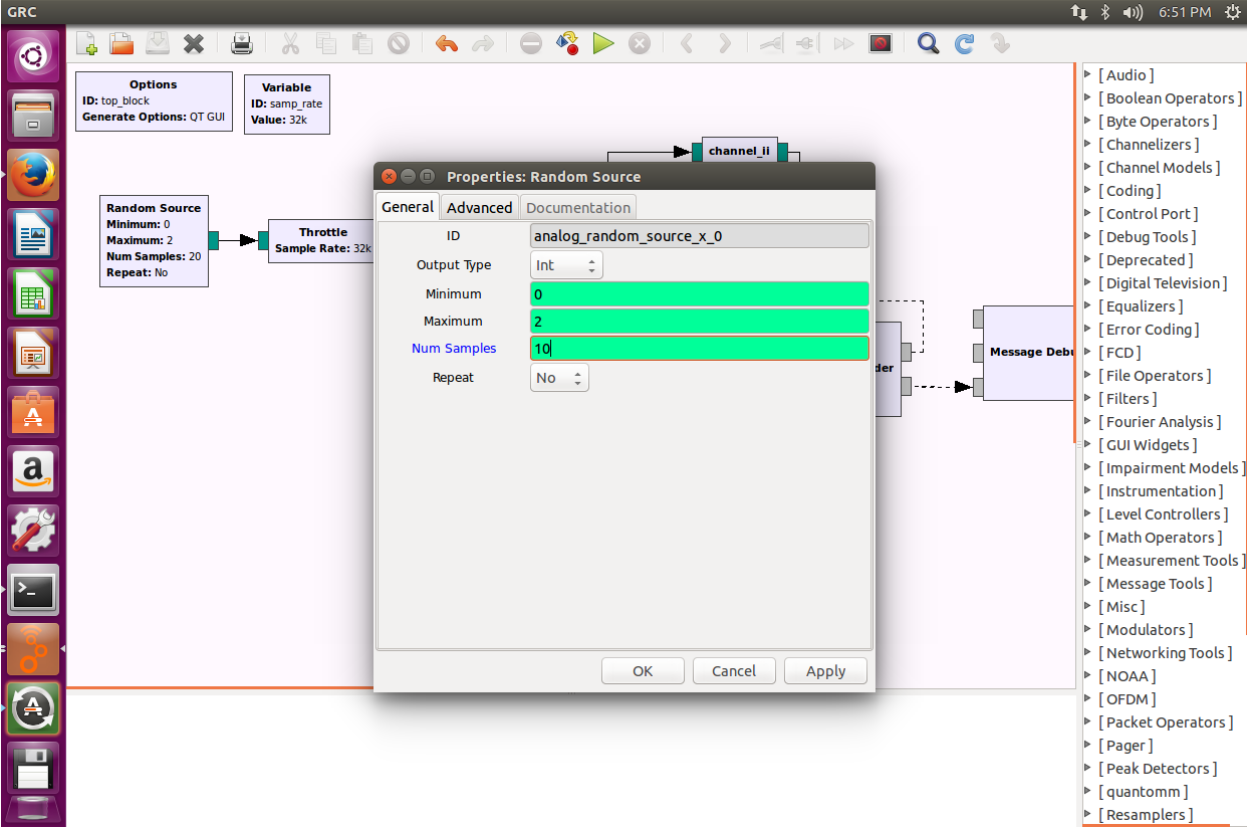
Start GNU Radio companion



Within GNU Radio companion, load flow graph gr-quantomm/examples/simulation.grc:

The screenshot displays the GNU Radio Companion 3.7.8 interface. The main workspace shows a flow graph for simulation. The flow starts with a **Random Source** block (Minimum: 0, Maximum: 2, Num Samples: 5, Repeat: No) connected to a **Throttle** block (Sample Rate: 32k). The output of the throttle goes into an **encoder** block. The output of the encoder is split into two paths, each going through a **channel\_model** block. The outputs of both channel models are then fed into a **decoder** block. The output of the decoder is connected to a **Message Debug** block. The interface also features a sidebar on the right with a list of block categories, including [Audio], [Boolean Operators], [Byte Operators], [Channelizers], [Channel Models], [Coding], [Control Port], [Debug Tools], [Deprecated], [Digital Television], [Equalizers], [Error Coding], [FCD], [File Operators], [Filters], [Fourier Analysis], [GUI Widgets], [Impairment Models], [Instrumentation], [Level Controllers], [Math Operators], [Measurement Tools], [Message Tools], [Misc], [Modulators], [Networking Tools], [NOAA], [OFDM], [Packet Operators], [Pager], [Peak Detectors], [quantomm], [Resamplers], [Stream Operators], [Stream Tag Tools], and [Symbol Coding]. At the bottom, a terminal window shows the following text: <<< Welcome to GNU Radio Companion 3.7.8 >>>, Preferences file: /home/scs/.grc, Block paths: /usr/local/share/gnuradio/grc/blocks, /home/scs/.grc\_gnuradio, /usr/share/gnuradio/grc/blocks, Loading: "/home/scs/gr-quantomm/examples/simulation.grc", >>> Done, and Showing: "/home/scs/gr-quantomm/examples/simulation.grc".

Double click Random Source and configure Num Samples:



Execute the flow graph:

The screenshot displays the GNU Radio Companion interface. At the top, the window title is "examples - GNU Radio Companion". The main workspace shows a flow graph with the following components and connections:

- Options:** ID: top\_block, Generate Options: QT GUI
- Variable:** ID: samp\_rate, Value: 32k
- Random Source:** Minimum: 0, Maximum: 2, Num Samples: 10, Repeat: No
- Throttle:** Sample Rate: 32k
- encoder:** Receives input from the Throttle block and outputs to two channel blocks.
- channel\_ji:** Two blocks receiving input from the encoder.

Below the flow graph, a terminal window shows the execution output:

```
Generating: "/home/scs/gr-quantomm/examples/top_block.py"
Executing: "/home/scs/gr-quantomm/examples/top_block.py"
encoder.work():bases: [0 1 1 1 0 1 0 0 1 0 0 1 0 0 1 1] angles: [ 45 45 45 0 135 0 0 135 90 0 135 90 90 45 135 90]
encoder.handle_msg():feedback: [0L, 0L, 1L, 1L, 0L, 0L, 1L, 0L]
encoder.handle_msg():plain text: [1, 1, 0] cipher text: [1, 1, 0]
decoder.handle_msg():plain text: [1L, 1L, 0L] cipher text: [1L, 1L, 0L]
* MESSAGE DEBUG PRINT PDU VERBOSE *
encoder.work():bases: [0 0 1 0] angles: [90 45 0 90]
encoder.handle_msg():feedback: [1L, 1L, 0L, 1L, 0L, 0L, 1L]
encoder.handle_msg():plain text: [0, 1, 0, 1] cipher text: [1, 0, 1, 1]
decoder.handle_msg():plain text: [0L, 1L, 0L, 1L] cipher text: [1L, 0L, 1L, 1L]
()
pdu_length = 3
contents =
0000: 01 01 00
*****
* MESSAGE DEBUG PRINT PDU VERBOSE *
()
pdu_length = 4
contents =
0000: 00 01 00 01
*****
>>> Done
```

## Running outside gnuradio-companion

```
scs@ubuntu:~/gr-quantomm/examples$ python top_block.py
```

```
encoder.work():bases: [0 1 1 0 0 1 1 1 0 1 1 1 0 1 1 1]
```

```
angles: [ 45 135 90 0 45 45 45 90 45 45 135 90 45 45 135 135]
```

```
encoder.handle_msg():feedback: [0L, 0L, 0L, 0L, 0L, 1L, 1L, 0L]
```

```
encoder.handle_msg():plain text: [1, 1] cipher text: [1, 1]
```

```
decoder.handle_msg():plain text: [1L, 1L] cipher text: [1L, 1L]
```

```
* MESSAGE DEBUG PRINT PDU VERBOSE *
```

```
()
```

```
pdu_length = 2
```

```
contents =
```

```
0000: 01 01
```

```
*****
```

```
encoder.work():bases: [1 0 0 0] angles: [90 0 0 0]
```

```
encoder.handle_msg():feedback: [1L, 0L, 1L, 1L, 0L, 1L, 0L, 1L]
```

```
encoder.handle_msg():plain text: [1, 1, 0, 1, 0] cipher text: [0, 1, 1, 1, 1]
```

```
decoder.handle_msg():plain text: [1L, 1L, 0L, 1L, 0L] cipher text: [0L, 1L, 1L, 1L, 1L]
```

```
* MESSAGE DEBUG PRINT PDU VERBOSE *
```

```
()
```

```
pdu_length = 5
```

```
contents =
```

```
0000: 01 01 00 01 00
```

```
*****
```

## Building

```
cd gr-quantomm/build
```

```
cmake ../
```

```
make
```

```
sudo make install
```

# Reload Blocks

