Mario Han
U28464330
Gerard Devlin
U16362958

EC415 Final Project Part B: GNU Radio

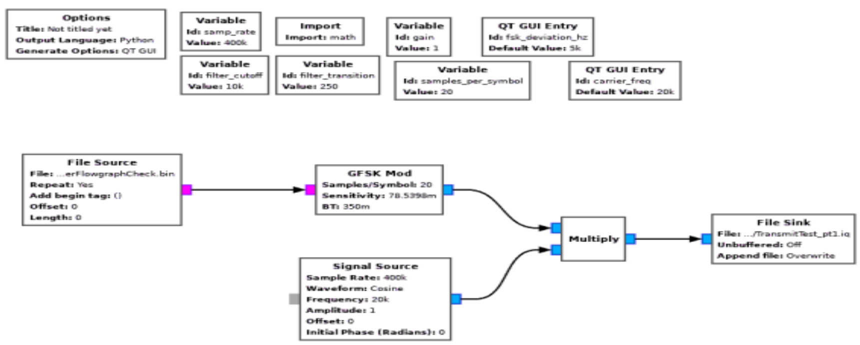
Individual Contributions:
Python – 50% each
Decoder.py – Gerard Devlin
Encoder.py – Mario Han
GRC Flowgraphs – 50% each
Receiver Flowgraph – Mario Han
Transmitter Flowgraph – Gerard Devlin and Mario Han
For the transmitter flowgraph, the team worked together on implementing it; however, we struggled getting it working so Gerard was able to go to office hours to get some help and complete it.
Transceiver Flowgraph – Gerard Devlin

<u>Questions</u>

1. The CRC is in the packet found from decodertest.bin is 0xf5a0096f

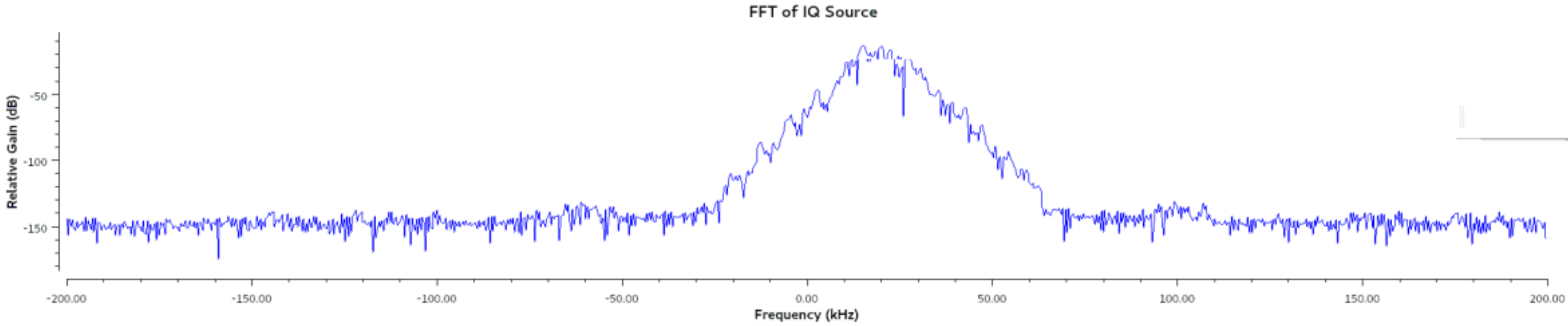
   Terminal output:
```
Milestone 2: Checking if Receiver Flowgraph works!
The CRC is: 0xf5a0096f
CRC Check:
True
```

2. The samples/symbol in our GFSK mod is 20 and the sample rate is 400k samples/sec. Thus, 400k/20 is 20k symbols/sec.



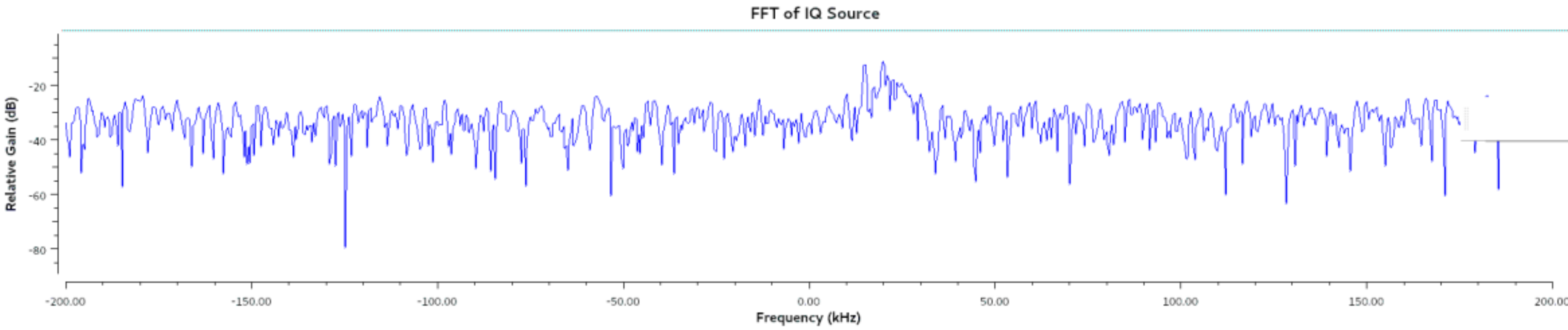
3. Difference between the peak amplitude and the base is 120 therefore 120 dB/ sqrt(2) = about 85 dB. Thus, 85 dB -140 dB (the base) = -55 dB. By estimating where -55 dB is located on the graph, we estimate the half power bandwidth is about 31 kHz (2 kHz to 33 kHz)

   FFT of IQ source when noise voltage is set to 0 V:



4. We raised the noise voltage on increments of 0.25 V and tested the results by looking at the errors we got running TransmitterTest_pt2.bin and seeing how many errors we get. For noise voltage of 1.75 V, we only get false.

   FFT of IQ Source when noise voltage is set to 1.75 V:

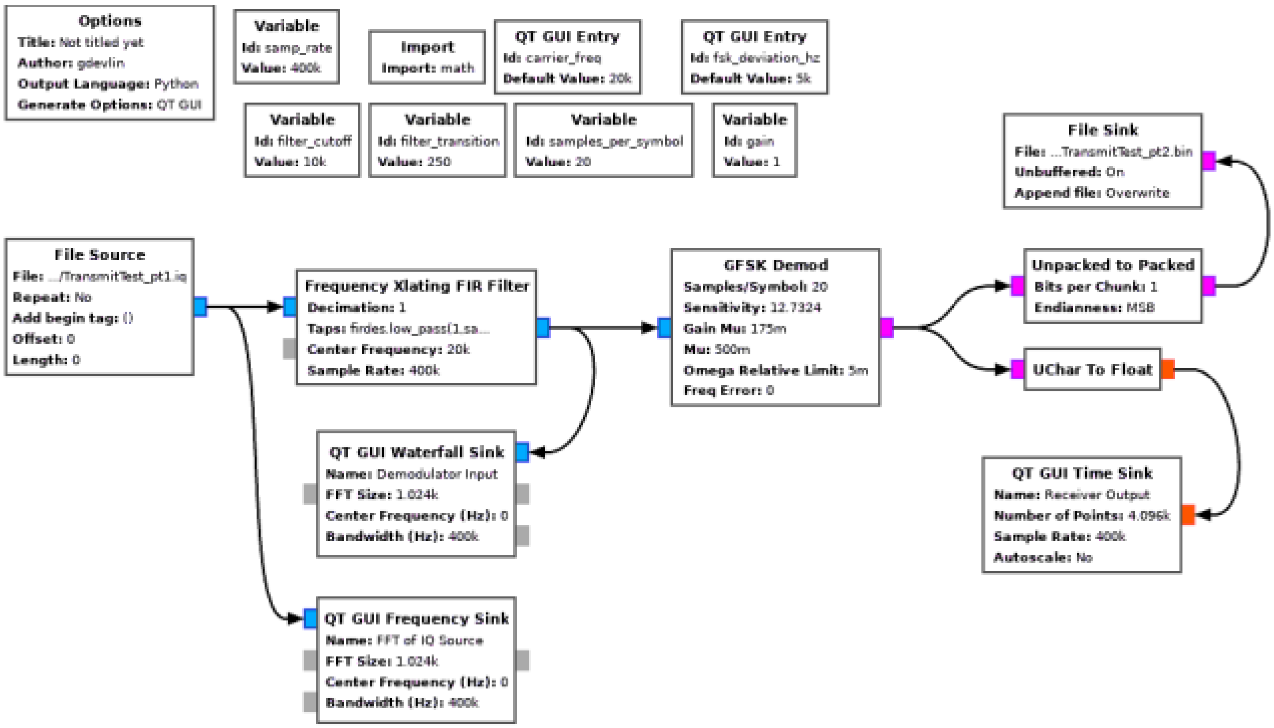


Terminal output:

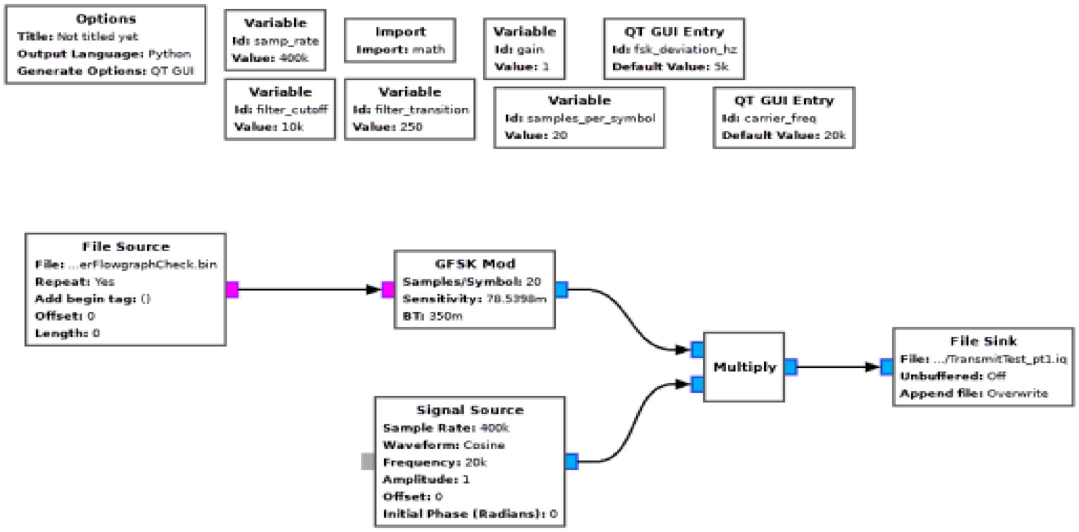
```
CRC Check:
False
Mile*tong 4: Checking Transmitter flowgraph!
The CRC is: 0xa8be39ef
CRC Check:
False
Mile*tone 4: Che*kingTransmittdr flowgraph!
The CRC is: 0xf8edea4
CRC Check:
False
�ilestooe 4: Checking Transmitter flowg*a*h!
The CRC is: 0x7d423f50
CRC Check:
False
Mildstone 4: Checking Transmitter flowgra*h!
The CRC is: 0x298dc4a9
CRC Check:
False
Mileston* *: *hecking *p**smhttgr flowgraph!
The CRC is: 0x8c7dcd64
CRC Check:
False
Milestona 4* Chackijg Transmitt*p flow*raph!
The CRC is: 0xf8ceca73
CRC Check:
False
Milesto-e 0: Checking transmitter flowgraph!
The CRC is: 0xc1274e95
CRC Check:
False
Milestone4: Checking Transmittirflowgr*ph!
The CRC is: 0xfd47f745
CRC Check:
False
*ilestone 4: Checking Transmittcr flowgraph!
The CRC is: 0x265d6dc4
CRC Check:
False
Milestone 4: Checking *ransmhtter flowgraph!
The CRC is: 0x260f8296
CRC Check:
False
Milestone 4: Checking Trans*i*tdr *lowgraph�
The CRC is: 0xaa14c546
CRC Check:
False
Mi*estol* 4z Chackijg Transmitter flowgraph!
The CRC is: 0x75b67c47
CRC Check:
False
Mileston* 4: Checking Trans*itter*flowgraph�
The CRC is: 0x345bf6e5
CRC Check:
False
```

Flow Graphs

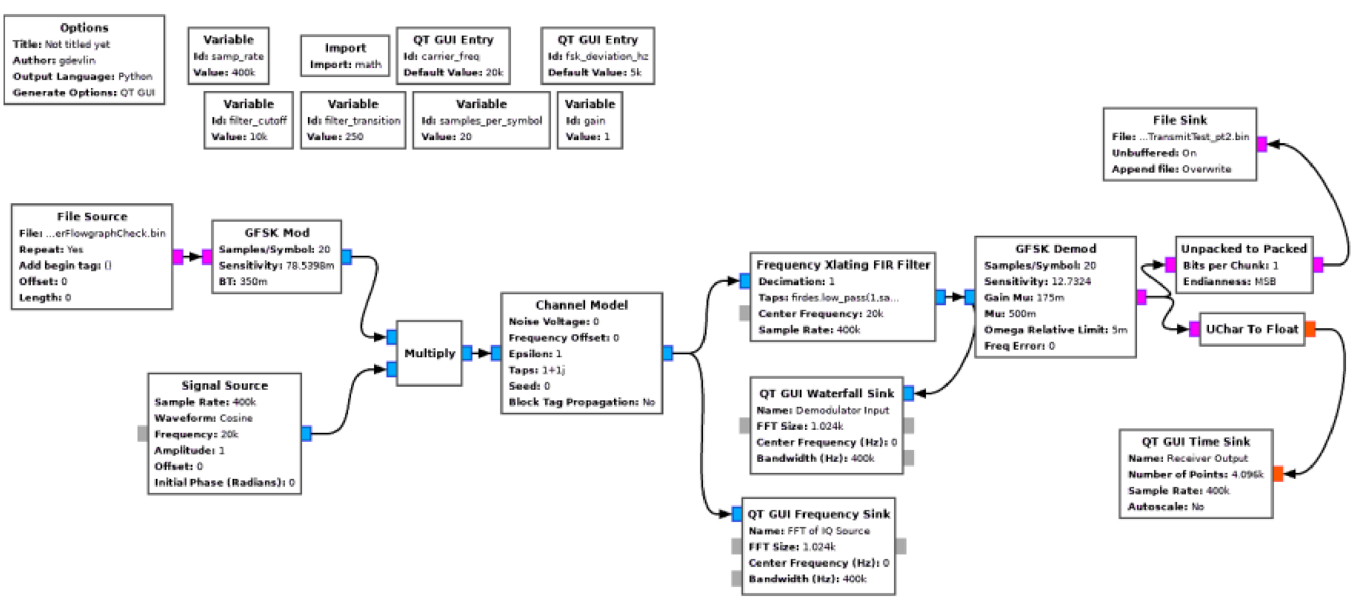
Receiver Flow Graph: The receiver flow graph was implemented using lab5 as a template. We only had to alter a few variables to ensure the correct carrier frequencies, sample rate, etc. The receiver takes in a signal and applies a low pass filter to it. It then passes the signal into the GFSK Demod block, which will demodulate the GFSK modulated signal. Finally, the unpacked to packed block takes 1 bit per byte passed to it and combines it back into bytes of data. The result is stored in a bin file through the file sink block.



Transmitter Flow Graph: The transmitter flowgraph takes in a message through the bin file in the file source block. That gets passed to the GFSK mod block where the message is converted into GFSK symbols. The output is our baseband signal. The next step is to mix the baseband with the carrier to get the passband signal. This is done by multiplying the baseband by a cos wave with the desired carrier frequency. The modulated passband signal is stored in an iq file with the file sink block.



Transceiver flow graph: The transceiver is simply a combination of the transmitter and receiver. The first part is the transmitter. It takes the message, modulates it, and then inputs it to the channel model. That passes it into the receiver flowgraph (described above). The channel model allows us to simulate noise in the signal.



Evidence of Complete System

Step 1:

Encode a message:

```
bash-4.2$ python EncoderTemplate.py M5_transceiver.bin
File path = "M5_transceiver.bin"
Enter a message: Milestone 5: A complete system. Encode, Transmit, Receive Decode!
your message:  Milestone 5: A complete system. Encode, Transmit, Receive Decode!
bash-4.2$ 
```

Step 2:

Run the transceiver

(flowgraph looks the same as above)

Step 3:

Decode the message:

```
Milestone 5: A complete system. Encode, Transmit, Receive, Decode!
The CRC is: 0x8f3b8143
CRC Check:
True
Milestone 5: A complete system. Encode, Transmit, Receive, Decode!
The CRC is: 0x8f3b8143
CRC Check:
True
```