Lab 2: Implementation of Analog Modulation schemes in GNU Radio

Aim of the experiment

- To understand the basics of analog communication (i.e.,amplitude and frequency modulation schemes)
- Implementing modulation – demodulation flow graphs for both amplitude modulation (AM) and frequency modulation (FM) in GNU Radio
- Using DVB - T (RTL – SDR) dongles to receive and demodulate locally transmitted AM signals.
- Using DVB – T (RTL – SDR) dongles to receive and demodulate locally transmitted FM signals.

Important Note

Using QT GUI Tab Widget to monitor multiple signals:-

QT GUI Tab Widget block in gnuradio can be used to monitor multiple graphs by giving a label to each graph to overcome the confusion about source of a graph. For example, if you want to monitor two signals in time domain namely modulated and demodulated then use two QT GUI Time Sinks e.g sink1 and sink2

1. the block 'QT GUI Tab Widget
2. Set the ID to 'tab' or any name you want to use.
3. Set no. of tabs to 2 (The number of labeled graphs you want to monitor.)
4. In Label 0 enter the label for first graph which is 'Modulated' in our example.
5. In Label 1 enter the label for second graph which is 'Demodulated' in our example.
6. Now to connect the GUI sinks to QT GUI Tab Widget open the block of respective QT GUI Time sink (in the current example).
7. In GUI hint field of QT GUI Time sink 1 which is plotting modulated signal enter tab@0 ('ID of GUI Tab Widget'@'Index of tab')
8. In GUI hint field of QT GUI Time sink 2 which is plotting demodulated signal enter tab@1 ('ID of GUI Tab Widget'@'Index of tab")

Important Note

Following images show the setup for above example:-

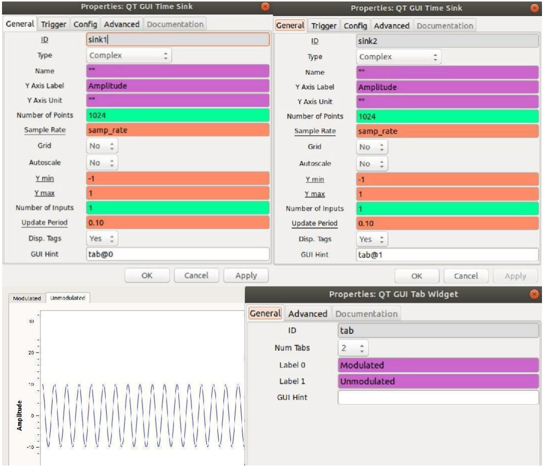


Figure: Using QT GUI Tab Widget to monitor multiple signals

Important Note

- Use the sample rate of 48KHz for all un-modulated signals(this sample is termed as Audio Rate in FM blocks-however,you don't have to use the ready made blocks)
- Use the sample rate of 960kHz for all the frequency or phase modulated signals in GNU-Radio(this rate is termed as Quadrature Rate in GNU radio FM blocks).
- Debugging steps:
  - If something is not working,trace the point of failure(by checking the signal at various nodes)
  - If you're not able to get the display after a new GNU-Radio block was added in the schematic,most likely you have entered wrong parameters in the new block(check carefully!)
  - **Make sure that you are consistently accounting for the sample rate whenever decimation(for downsampling) and interpolation(for upsampling) are used.**
- IIR filter block implementation:
  - FF coefficients=[b_0,b_1];FB coefficients=[a_0,a_1];Old Style of Taps=True, implements the discrete time filter:

$$\frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1}}{a_0 - a_1 z^{-1}}$$

  - A bug in implementation always sets the value of $a_0 = 1$. Therefore, you must use $a_0 = 1$ in all your calculations for filter coefficients.

Task 1: Implementation of DSB-FC

- Implement an entire DSB-FC AM modulation-demodulation flow graph in GNU radio.

  Note: You cannot use the built-in blocks for demodulation like AM-demod
- Parameters to be used:
  - ◇ Message signal (single-tone): 10 kHz
  - ◇ Carrier signal: 100 kHz
- You are allowed to use the ready-to-use Low Pass Filter block from GNU radio library for this
- Observe the message signal and modulated signal for DSBFC in time and frequency domains

Task 2: Demodulation of AM

- Use the DVB–T (RTL-SDR) dongle to receive and demodulate the locally transmitted DSB-FC AM signal in the lab (DSB-FC has an additional DC component, which can be removed by the 'DC Blocker' block in GNU radio).
- Parameters to be used:
  - Transmitted carrier frequency: 850 MHz with 100 kHz offset (i.e. 850.1 MHz)
  - AM carrier frequency after reception by RTL-SDR: 100 kHz (i.e. use 850 MHz as the channel frequency)
  - Message signal (audio signal) bandwidth 20 Hz to 15 kHz

  Note:
  The transmit frequency and RTL-SDR frequencies have a small offset between them. A USRP kit is used for local AM transmission at 850 MHz (with a small offset in the actually transmitted frequency). Similarly, RTL–SDR will not tune to precisely 850 MHz (if you specify 850 MHz as the channel frequency), as the frequency references are not precise. For better tuning, use a slider for compensating the offset between the two frequencies. Observe the FFT of received signal (by directly connecting FFT sink and RTL–SDR) and play with the slider to tune correctly.
- Feed the demodulated signal to the Audio Sink block. If the demodulation is done properly, you should be able to listen to the transmitted music signal. Observe the demodulated spectrum.

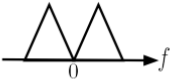
Task 3: Implementation of DSBSC

You have to implement DSBSC signal. Here you just have to suppress the carrier signal and the implementation is same as AM DSBFC.

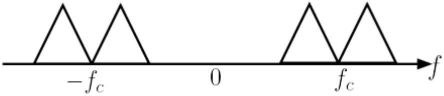
1. Implement the modulation flowgraph for DSB-SC transmission. Use a single tone at 10kHz as the message, and 500kHz as the carrier frequency.
2. Observe the spectrum of the modulated signal.
3. Implement the demodulation flowgraph.
4. Observe the spectrum of the demodulated message signal.
5. Replace the tone message signal with an audio message.Can you recover the message post-demodulation?

Task 4: Implementation of SSB

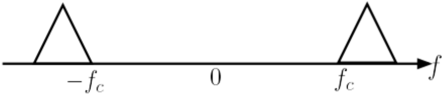
You have already implemented DSB-SC modulation.Here, only one sideband of the message is transmitted. For real messages, note that this is sufficient. This requires only half the bandwidth of DSB transmission!



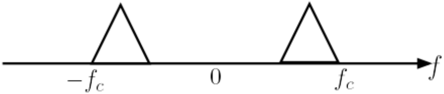
Message signal



DSB-SC transmitted signal



SSB-SC (upper sideband) transmitted signal



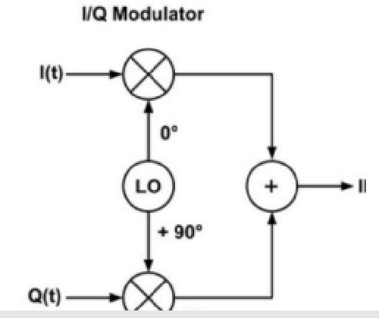
SSB-SC (lower sideband) transmitted signal

1. Implement the modulation flowgraph for SSB-SC (upper sideband) transmission. Use a single tone at 10kHz as the message, and 500kHz as the carrier frequency.
   Hint: Think of the transmitted(passband) signal as
   $$s_p(t) = Re([s_I(t) + js_Q(t)]e^{j2\pi f_c}) \\ = s_I(t)cos(2\pi f_c t) - s_Q(t)sin(2\pi f_c t)$$
   *W*hat should s_I and s_Q be for SSB transmission? Use Hilbert Txform
2. Observe the spectrum of the modulated signal.
3. Implement the demodulation flowgraph.
4. Observe the spectrum of the demodulated message signal.
5. Replace the tone message signal with an audio message.Can you recover the message post-demodulation?
6. Repeat the above to achieve lower sideband transmission.

Task 5: IQ modulation and Demodulation

Part1: IQ Modulation

1. An IQ modulator is typically used for upconverting a complex baseband signal to a passband intermediate frequency (IF) or radio frequency (RF) signal.
2. Its equation can be given as,
   $s(t) = I(t) * sin(2\pi f_c t) + Q(t) * sin(2\pi f_c t + 90^o)$
3. Implement the IQ modulation using two music files given (music.wav vocal.wav)
   Treat one as in-phase signal and the other as quadrature-phase.
4. Use the carrier frequency as 100KHz, sampled at appropriate rate.
5. Store the modulated signal in data1.dat file in file sink


I/Q Modulator
I(t)
0°
LO
+ 90°
Q(t)
+

Part2: IQ Demodulation

1. Demodulate the IQ modulated signal using shown block diagram, from data stored in data1.dat file.
2. Use a low pass filter to filter higher unwanted frequencies.
3. Play the demodulated audio files. Observe if you get a faithful reconstruction.