Lab 4 - Signal Impairments and Channel Estimation

Goal

Characterizing the impairments of phase noise and fading, and how to estimate the channel response that includes both impairments using a PN sequence.

1 Fading Measurements and Equalization

In this part of the lab, you will measure an estimate of the channel response and perform equalization on the received signal. For a general discrete channel, the impulse response $h[n]$ from multiple echos is the following:

$$h(n) = \sum_k \alpha_k \delta(n-k)$$

where in general $k > 1$ and $\alpha_k = a_k e^{j\phi_k}$ can be complex. Note that for the first part of the lab $\alpha_0 = 1$, $\alpha_1 = A$ and $\alpha_k = 0$ for $k > 1$. This is a "single-echo" model. For the general multiple echo case, the corresponding Z-transform from your DSP class is:

$$H(z) = \sum_k \alpha_k z^{-k}$$

Hence, the inverse response of the channel $H(z)$, denoted $G(z)$ is the following:

$$G(z) = \frac{1}{\sum a_k z^{-k}}$$

$$H(z)G(z) = 1$$

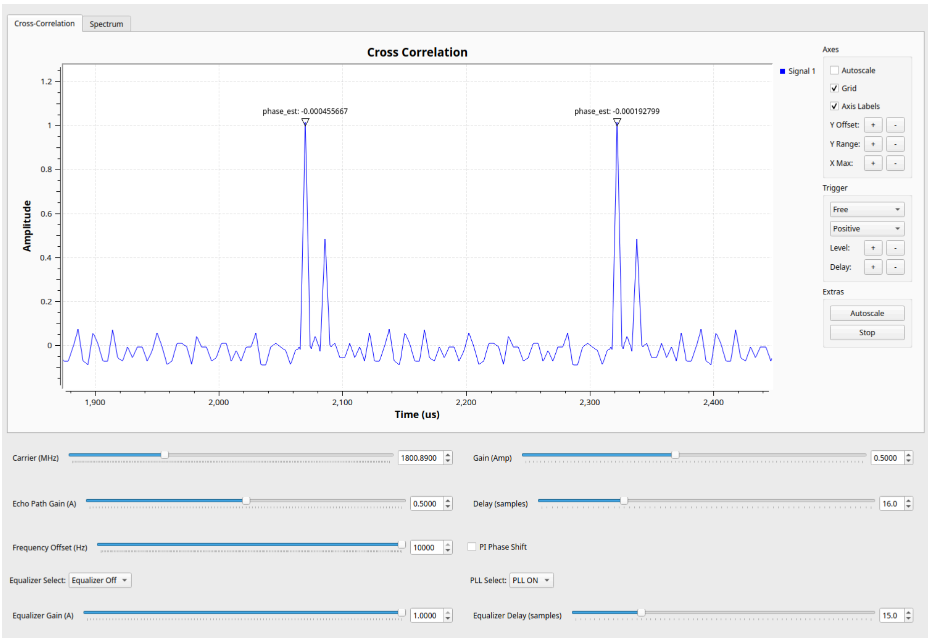
The receiver front panel, implements the filter coefficients for $G(z)$ and applies the filter to the received signal samples. Note: $G(z)$ is an IIR (Infinite Impulse Response) filter, since $H(z)$ is assumed to be FIR.

1. Before proceeding to the next part, make sure to zoom-out frequency-axis to its maximum (i.e. set $F \in [-\frac{f_s}{2}, \frac{f_s}{2}]$) on your power spectrum graph so that you can see the complete sinc-spectrum.
2. Set your fading profile on the transmitter to the following:
   a) Echo Gain = [0.5]

b) Delays = [16]

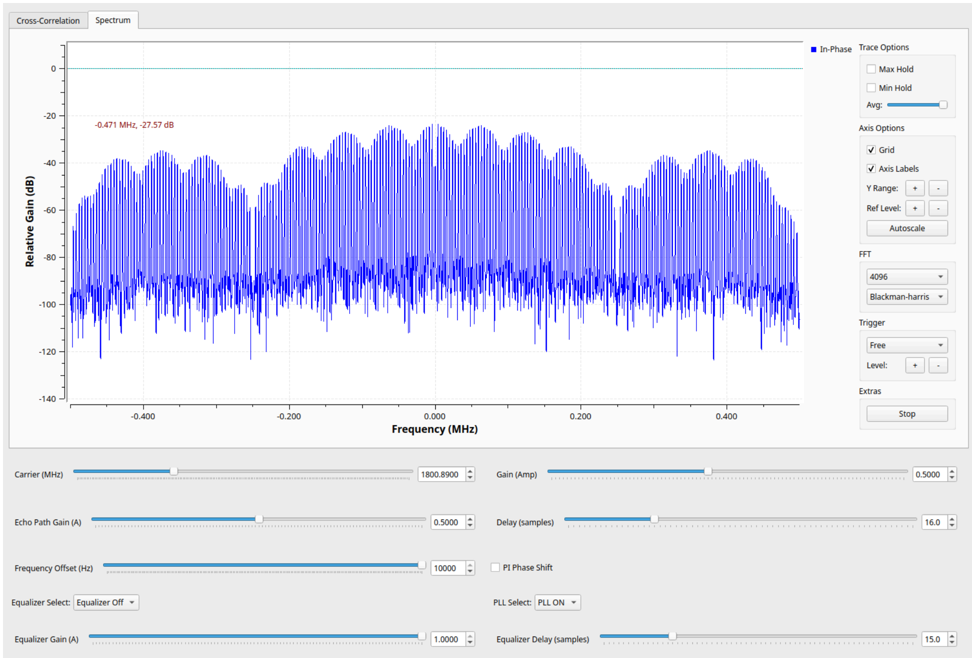
This now corresponds to a single echo channel impulse response $h(t) = \delta(t) + \frac{1}{2}\,\delta(t - 16T_{\text{sample}})$. The $\delta(t)$ term corresponds to the direct path while the other term corresponds to the single echo path.

3. Zoom in the cross-correlation and measure the gains (i.e. the measured amplitudes on the correlation graph) of the direct and echo path terms, then measure the relative time delay of the direct path to the echo path. Does the time delay correspond correctly with the number of sample delays and the sampling time? A sample screenshot is below



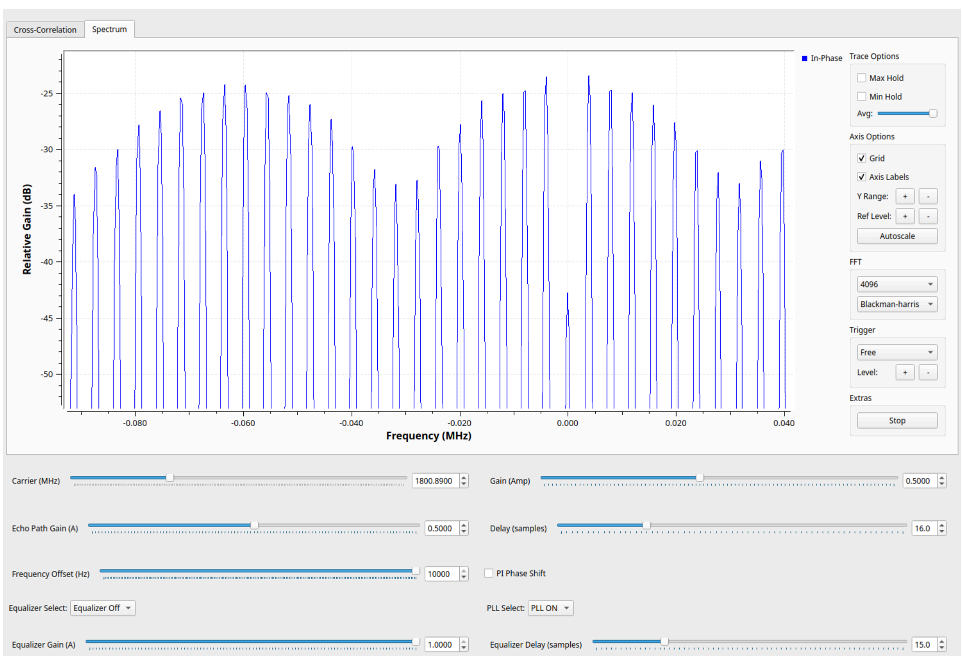
(1.0 and 0.5)

4. Switching to the frequency domain and using the modulated spectrum determine the same parameters. (If you are unsure of how to do this, vary the echo path gain A and the delay T in samples and determine how it affects the spectrum and thus how to determine A and T). A sample screenshot of the spectrum is below where the amplitude is in log units.



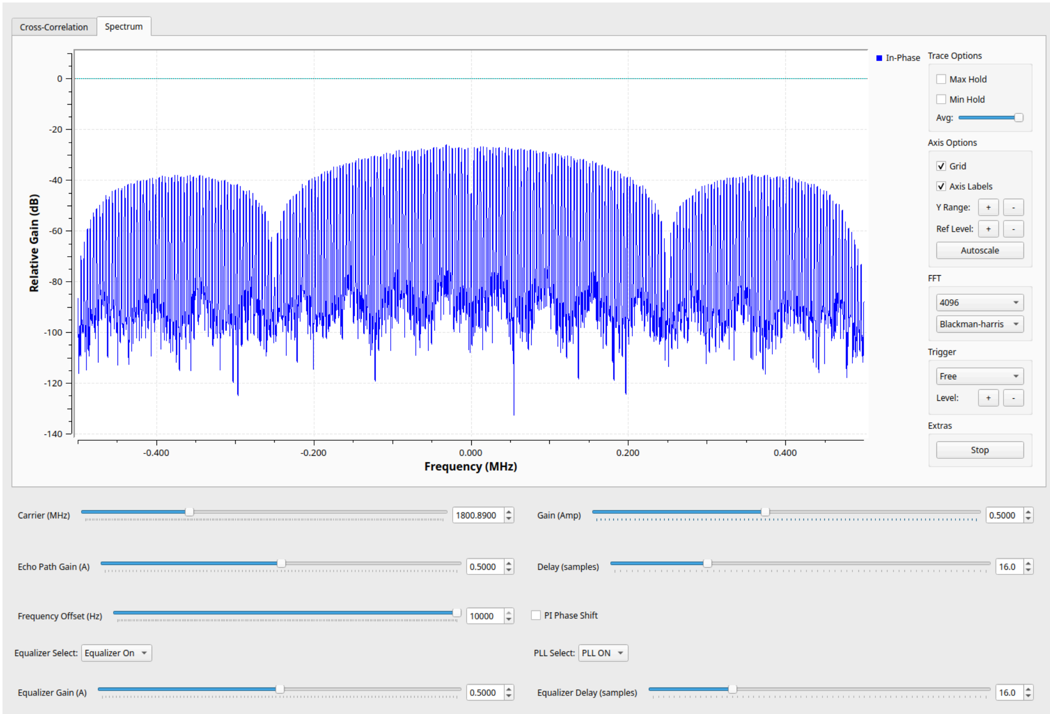
When we increase the echo gain the amplitude of the ridges increases

Increasing the delay creates more samples (number of ridges increases)



deltaF = 1/(250 microsec) = 4000 Hz = spacing between peaks

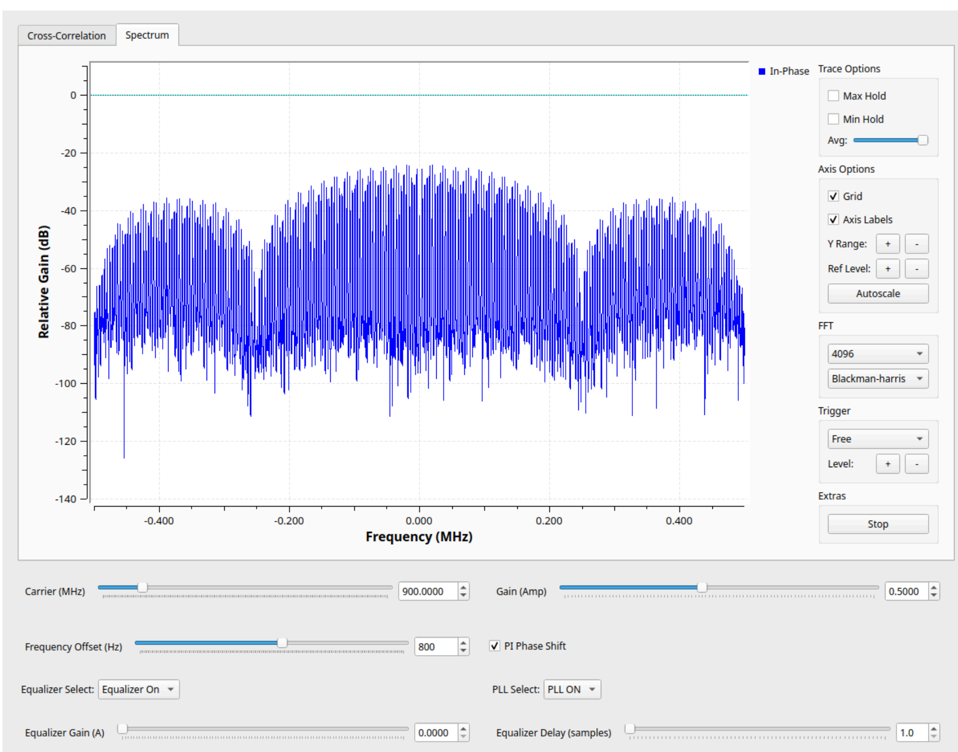
5. Assuming prior knowledge of the fading gains/delays at the transmitter, enter in the same gain and delay values into the equalizer gain and equalizer delay to undo the effects of the channel. What do you observe? Why?



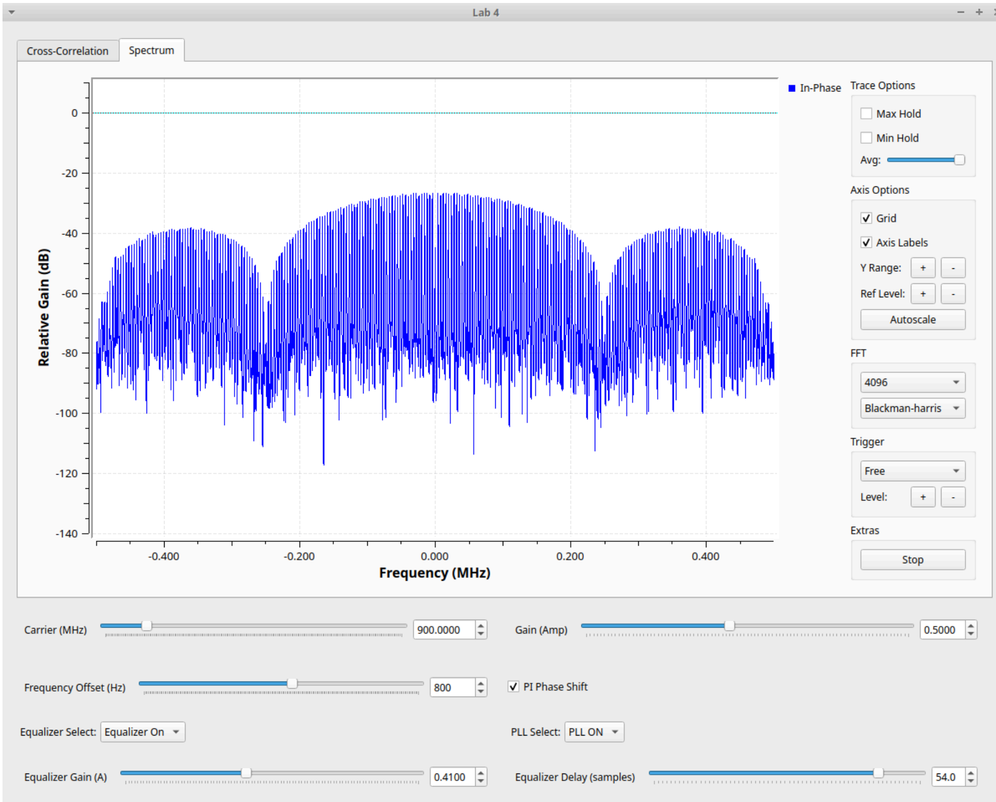
Ripples in the passband and stopband are gone (canceled out)

6. Now open “Lab4_random.py”, which has a random value of *A* and *T* expressed in samples. Using the equalizer slide controls, equalize this channel and thus determine *A* and *T*. <u>Make sure to provide a screenshot of the “spectrum” before and after equalization.</u>

Before:

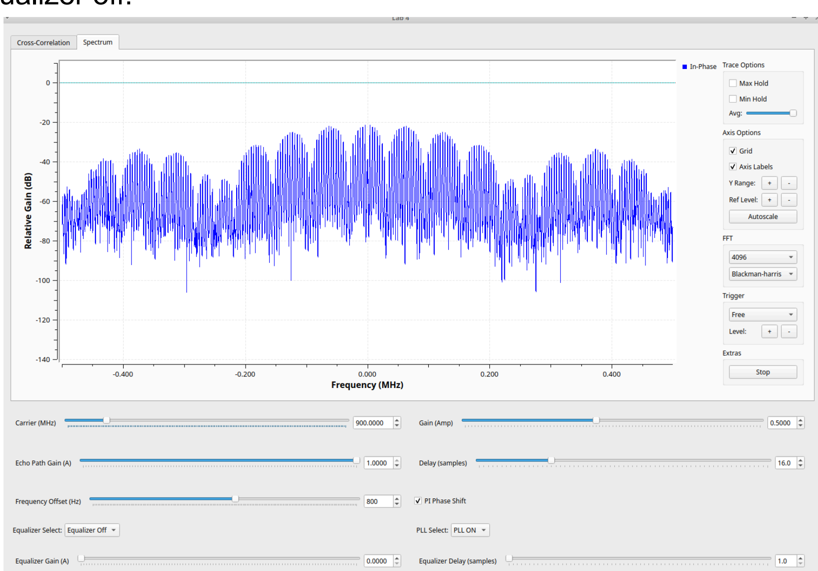


After:

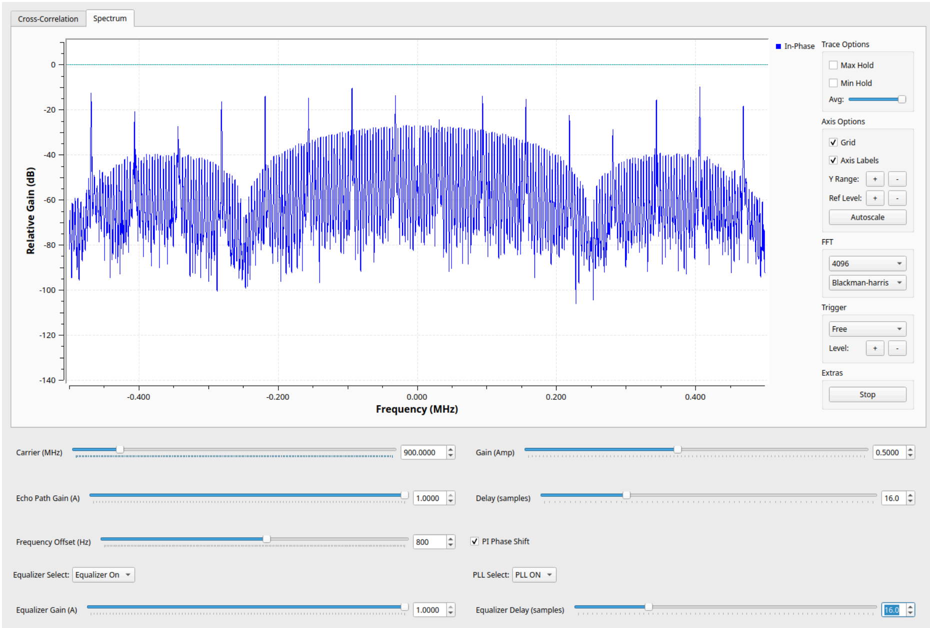


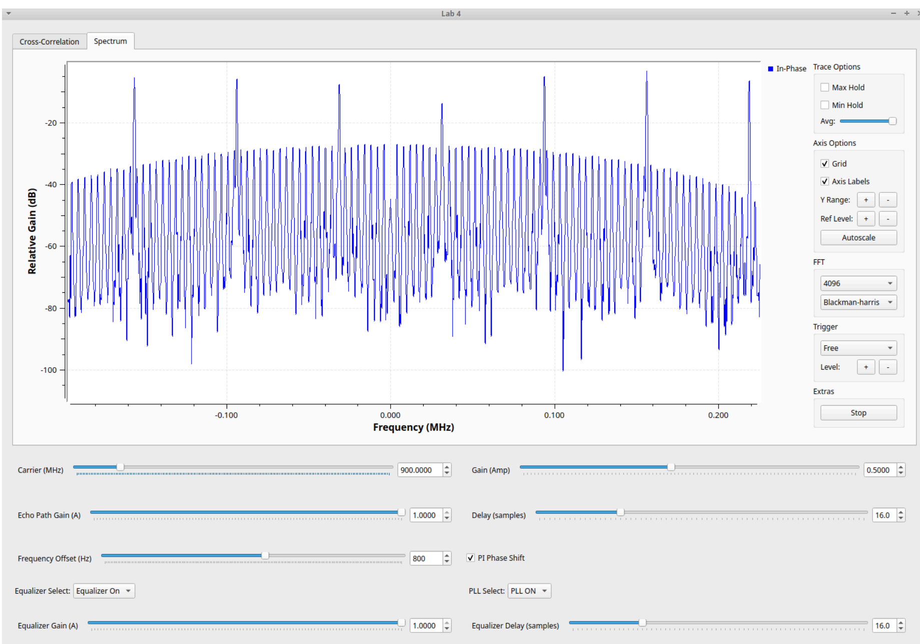
7. Open “Lab4.py” and set A = 1 and T = 16 samples. Turn the equalizer off and screenshot the spectrum. Now turn on the equalizer with the same values for A and T. Observe the “spikes” in your power spectrum. At what frequencies do they occur? Why? (Hint: What are the zeros of $H(z)$ and the corresponding poles of $G(z)$?). A sample screenshot is below.

Equalizer off:



Equalizer on: (Spikes are periodic)





Write Up and Scoring for This Part of Lab (60 points)

1. (30 points) Screenshots and answers to all questions posed in the Section.

2. (10 points) Describe how you would create an FIR approximation of the equalization filter $g(n)$ denoted $\hat{g}(n)$. Assuming the following $H(z)$:

$$H(z) = 1 + Az^{-T}$$

$$G(z) = 1/\ 1 + Az^{-T}$$

You may assume that $|A| < 1$, and T is a positive integer. (Hint: What can you say about

$g(n)$ for large values of $n > N$.)

$$g(n) = \sum_{k=0}^{N} (-A)^k \, \delta(n - kT)$$

Since A decays exponentially, for large N, the higher-order terms become negligible and the truncated series provides a good estimation.

a) Find a $\hat{g}(n)$ when $A = 0.5$ and $T = 4$.

$$\hat{g}(0) = (-0.5)^0 = 1$$
$$\hat{g}(4) = (-0.5)^1 = -0.5$$
$$\hat{g}(8) = (-0.5)^2 = 0.25$$
$$\hat{g}(12) = (-0.5)^3 = -0.125$$
$$\hat{g}(16) = (-0.5)^4 = 0.0625$$
$$\hat{g}(20) = (-0.5)^5 = -0.03125$$

As N gets larger, it approaches 0

3. (10 points) Based on your observations from this lab, describe the relationship between invertibility of a channel $H(z)$ in respect to its poles and/or zero locations. In other words, what kind of channels $h[n]$ are invertible (i.e. does a causal equalizer filter $g[n]$ exist?) and what kind of channels are non-invertible.

invertible Channels: Channels where all zeros of H(z) lie inside the unit circle (minimum phase channels) are invertible with a causal equalizer filter g[n].

Non-Invertible Channels: Channels with zeros on or outside the unit circle are non-invertible as their inverses will be either non-causal or unstable.

4. (10 points) Suppose that we want to estimate the channel using a particular training PN sequence called the "DB" sequence. All you know about this code is that it has good autocorrelation properties (i.e. $\sum w(k)w^*(n+k) = \delta(n)$) and the following periodic property:

$$w(n) = \begin{cases} w(n + mL) & 0 \le m \le 9 \\ 0 & \text{otherwise} \end{cases}$$

Effectively, the "DB" sequence is a length L sequence that repeats itself 10 times. Now, assuming the input to your receiver is the "DB" sequence $w(n)$ with a frequency offset of f_0:

$$x(n) = w(n)e^{j2\pi f0n}$$

a) Find the cross-correlation between the received sequence $x(n)$ and the "DB" sequence $w(n)$, denoted $r_{xw}(n)$

$$r_{xw}(n) = \sum_k x(n+k)w^*(k)$$

for values of $n = \{0, L, \ldots, 9L\}$. For this part you may assume that the term A is a constant where

$$A = \sum_k |w(k)|^2 e^{j2\pi f_0 k}$$

$$r_{xw}(n) = \sum_{m=-\infty}^{\infty} w(m) e^{j2\pi f_0 m} w(m+n)$$

b) Using your calculation of $r(0)$, and the fact that $|w(k)|^2 = 1$, comment on the effect that frequency offset has on the *magnitude* of the cross-correlation.

Zero Cross-Correlation: When the frequency offset f0 does not align with the frequencies that make the sum non-zero, the cross-correlation r(0) is close to zero.
Non-Zero Cross-Correlation: When the frequency offset f0f_0f0 aligns with the frequencies that make the sum non-zero (i.e., f0=k/9L), the cross-correlation r(0) will be non-zero and its magnitude will depend on the alignment of f0 with these frequencies.

c) Explain how you would calculate the frequency offset from your calculated values of $r_{xw}(mL)$.

$$f_0 = \Delta\phi/2\pi L$$

d) What is the maximum frequency offset f_{max} that can be tracked using the cross correlation approach and the "DB" code $w(n)$ of length L (Hint: Think of the Nyquist sampling theorem).

According to the Nyquist-Shannon sampling theorem, the sampling rate must be at least twice the maximum frequency component of the signal to avoid aliasing.

$$f_{max} = fs/2 = \mathrm{L/T}$$

where T is the duration of the code w(n) and L is the length of the code

e) Based on your analytic results in this question, and measurements in this lab, what would you expect to happen when you increase the length of the PN Sequence? What are the advantages and disadvantages of changing the PN sequence length? What is the effect of repetition of PN sequences, say we repeated this for 100 times instead of 10, how would your answers change (qualitatively)?

Increasing the length of the PN sequence improves frequency resolution and localization of the frequency offset peak but increases computational complexity. (enhances robustness but increases processing load).

2 Orthogonal Frequency Division Multiplexing

In this section, you will learn about the basics concepts in Orthogonal Frequency Division Multiplexing (OFDM). OFDM is a multi-carrier modulation scheme which consists of closely spaced sub-carriers orthogonal to each other. Each of these sub-carriers can be used to carry data (often lower bit-rate compared to a wideband carrier), providing resiliency to frequency selective fading, interference and multipath effects. Since we pack a large number of these subcarriers within a given bandwidth, although each of these subcarriers individually carry less data compared to a wideband single-carrier system, the sum of all these sub-carriers provide very high data rates and improve spectral efficiency.

Firstly update the Git repository so that we have the latest files.

1. To do so, open a terminal on the VM and type cd ece157A.
2. Type git pull, so that the Git repository is up-to-date.
3. Change directory by executing cd grc lab5.

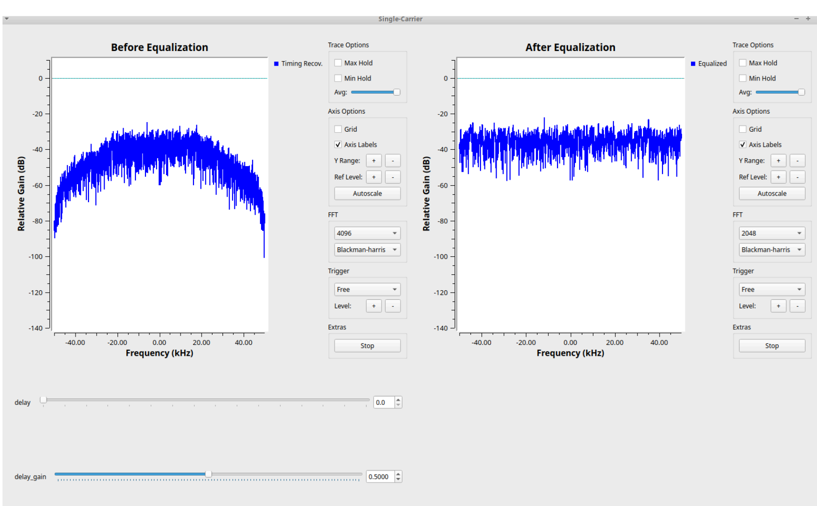
2.1 Single-carrier vs Multi-carrier comparison

In the first part of this section you will compare a single carrier system (similar to one in the previous section) with a multi-carrier system.

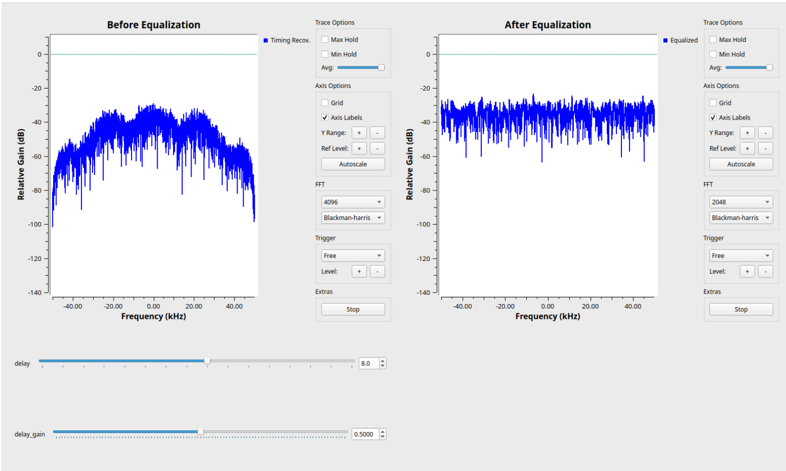
Single-Carrier

1. Run the *lab5_single_carrier.py*, which gives you control over two variables delay and delay gain. A sample screenshot is given below. Keeping the delay gain at 0.5, vary the delay value between 0, 8 and 15. Capture screenshots for the above values.

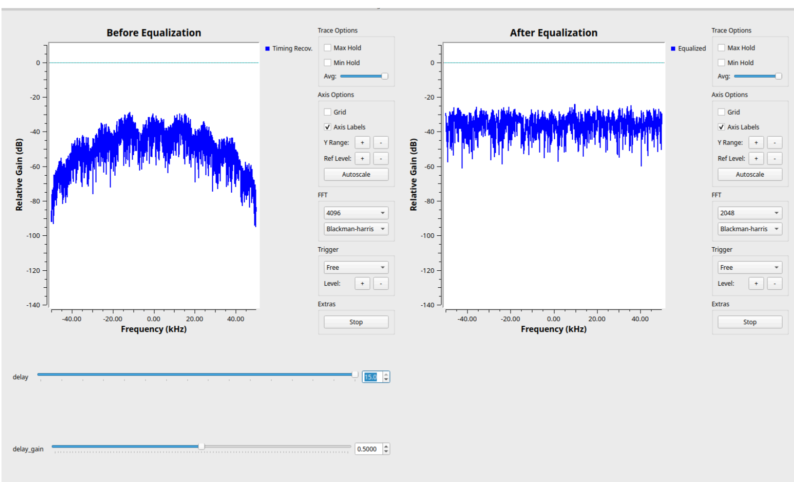
0:



8:

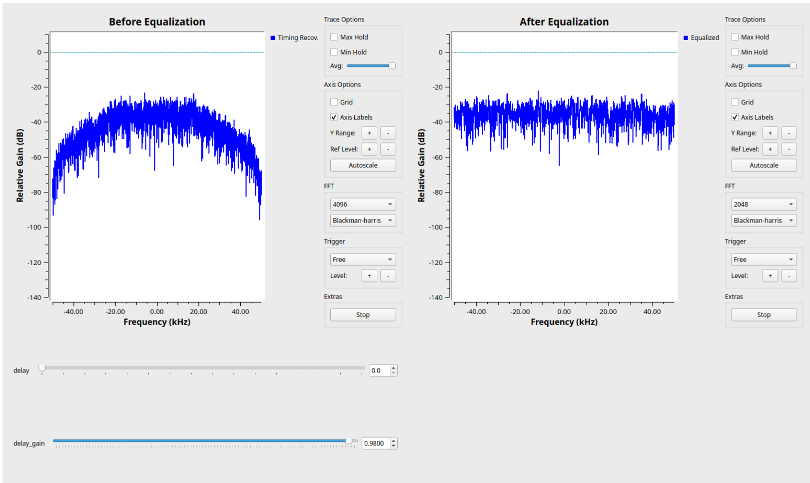


15:

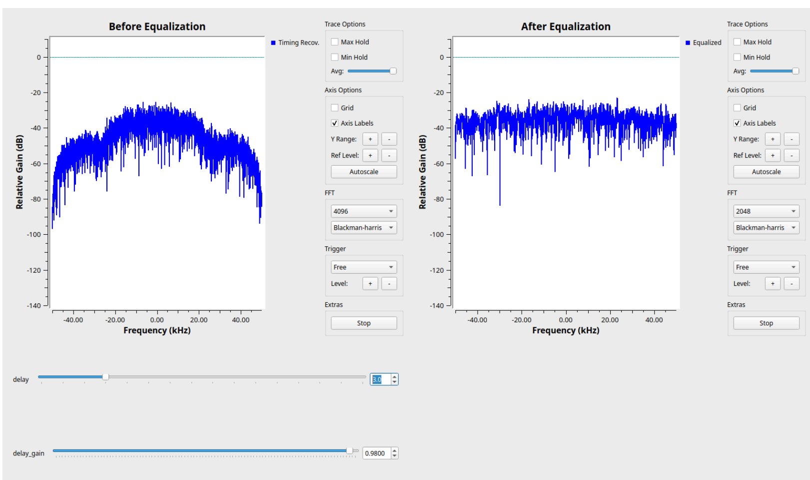


2. Now increase the delay gain to 0.98. Then vary the delay value in increments of 3 starting from 0 (0, 3, 6, 9, 12 and 15). Observe the spectrum after equalization and compare how it changes while increasing the delay and when the delay gain changes.

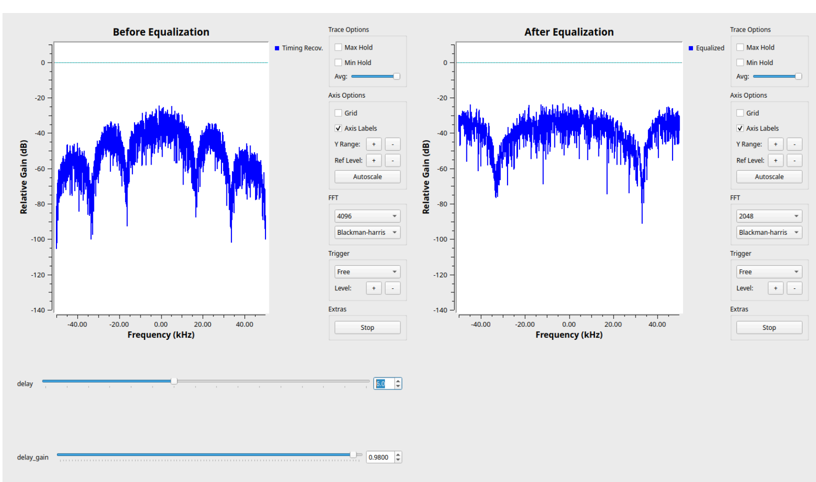
0:


Before Equalization
Timing Recov.
Relative Gain (dB)
Frequency (kHz)
After Equalization
Equalized
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
2048
Blackman-harris
Trigger
Free
Level:
Extras
Stop
delay
0.0
delay_gain
0.9800

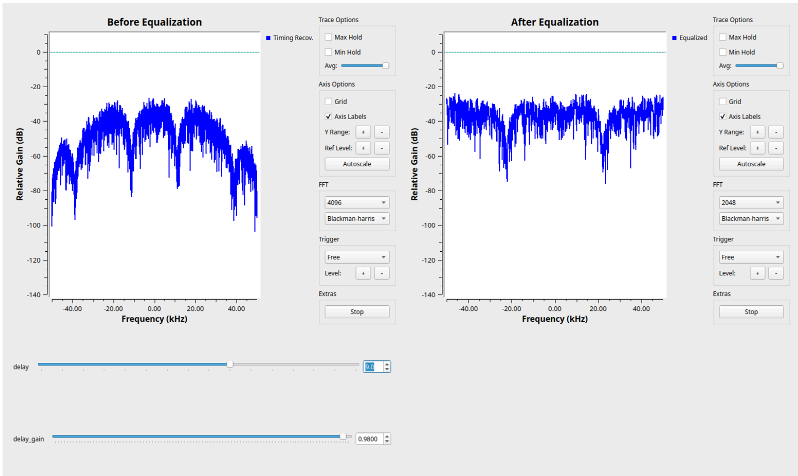
3:


Before Equalization
Timing Recov.
Relative Gain (dB)
Frequency (kHz)
After Equalization
Equalized
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
2048
Blackman-harris
Trigger
Free
Level:
Extras
Stop
delay
delay_gain
0.9800

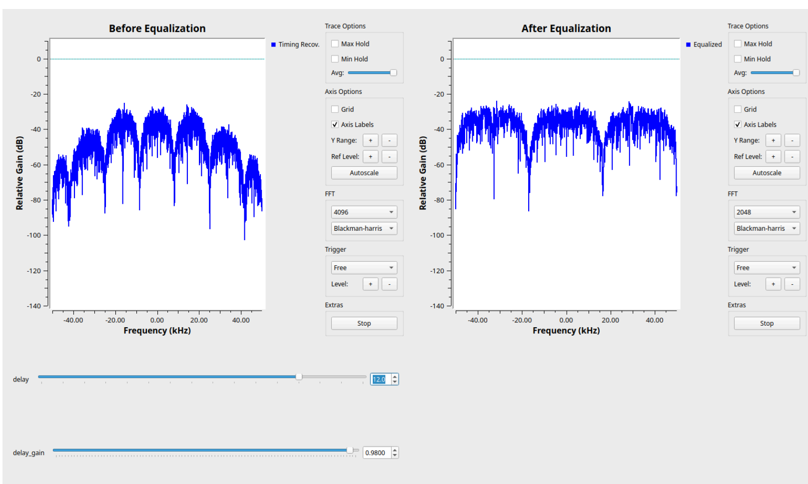
6:


Before Equalization
Timing Recov.
Relative Gain (dB)
Frequency (kHz)
After Equalization
Equalized
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
2048
Blackman-harris
Trigger
Free
Level:
Extras
Stop
delay
delay_gain
0.9800

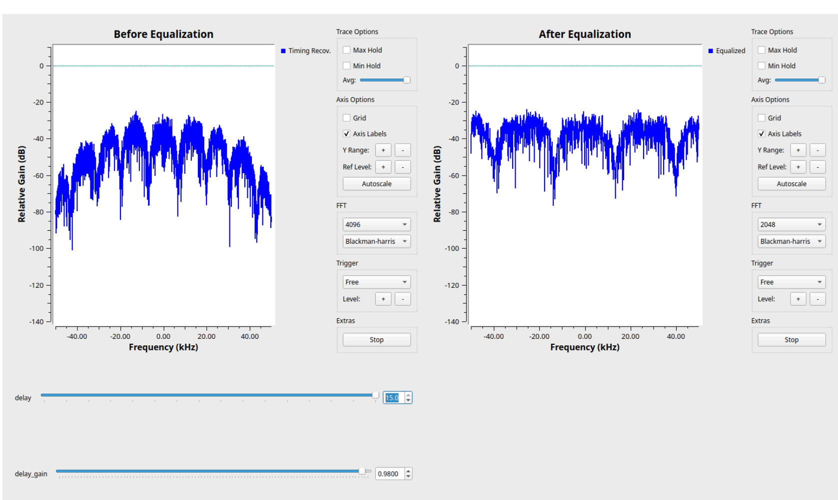
9:


Before Equalization
Timing Recov.
Relative Gain (dB)
Frequency (kHz)
After Equalization
Equalized
Relative Gain (dB)
Frequency (kHz)
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
2048
delay
delay_gain
0.9800

12:


Before Equalization
Timing Recov.
Relative Gain (dB)
Frequency (kHz)
After Equalization
Equalized
Relative Gain (dB)
Frequency (kHz)
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
2048
delay
delay_gain
0.9800

15:

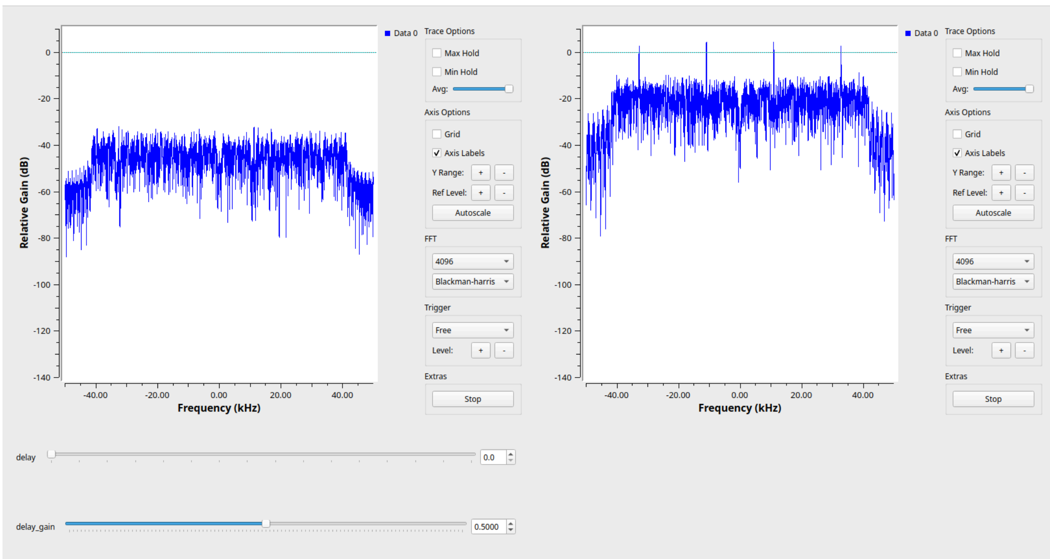

Before Equalization
Timing Recov.
Relative Gain (dB)
Frequency (kHz)
After Equalization
Equalized
Relative Gain (dB)
Frequency (kHz)
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
2048
delay
15.0
delay_gain
0.9800

Multi-Carrier OFDM

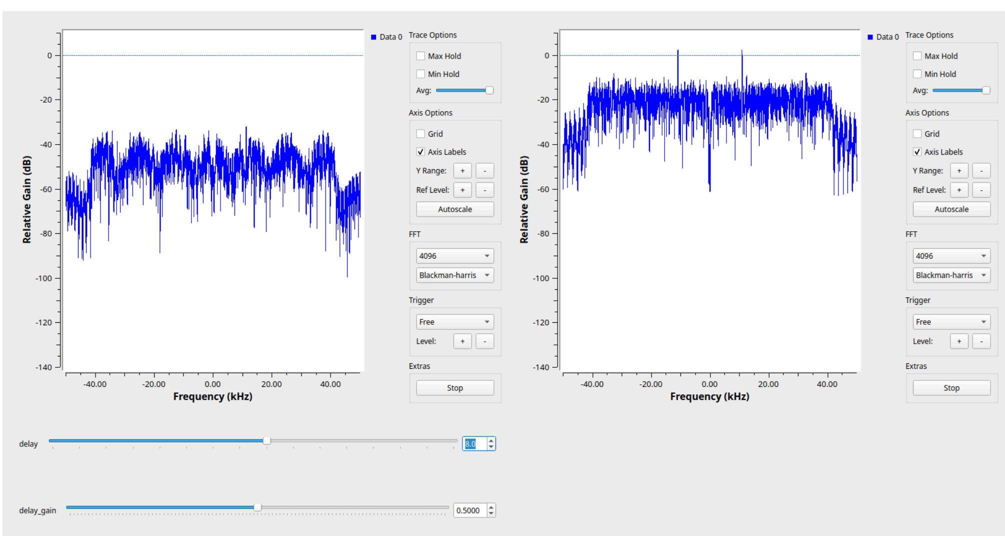
1. Run the *lab5_multi_carrier.py*. A sample screenshot of the output is shown below.

2. Similar to the single-carrier case, vary the delay between 0, 8 and 15 keeping the delay gain at 0.5. Capture screenshots for those values.

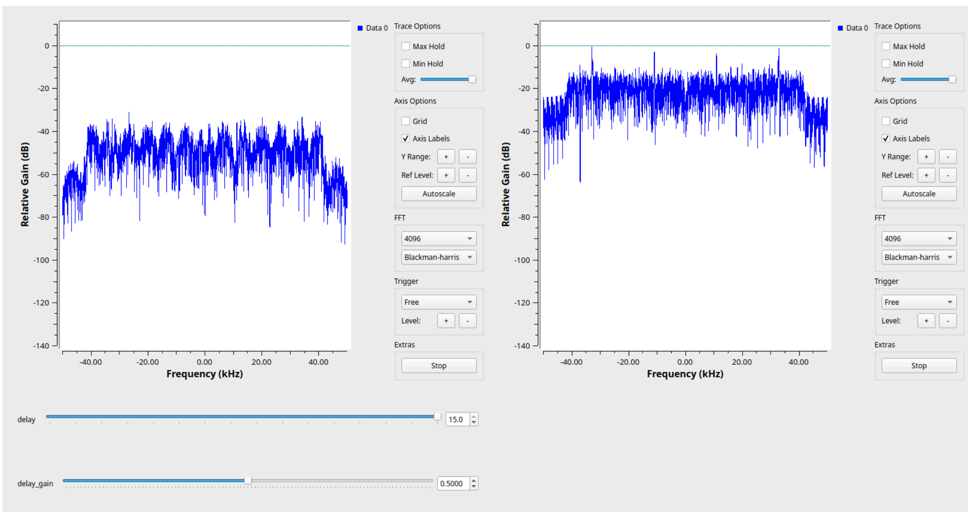
0:



8:

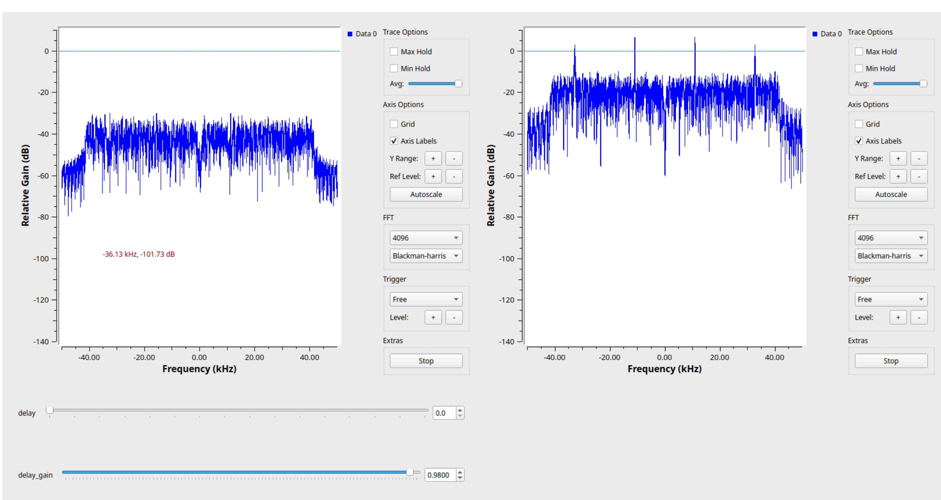


15:

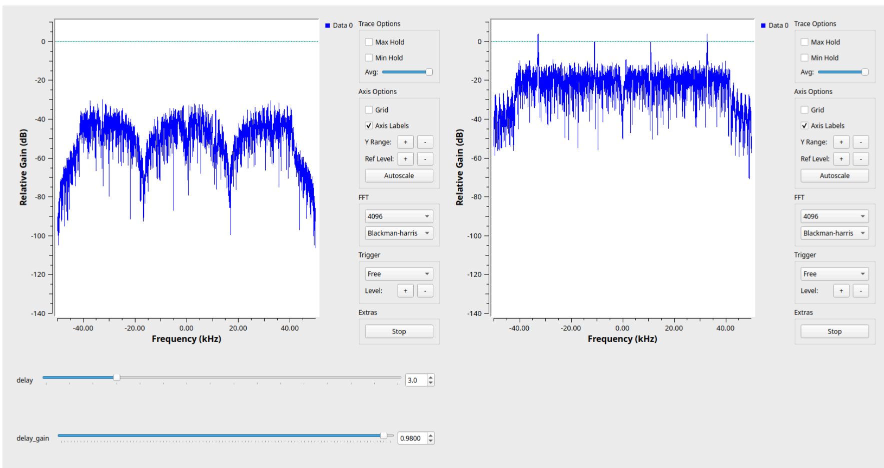


3. Repeat the same step as in the single-carrier case, by increasing the delay gain to 0.98. Then vary the delay value in increments of 3 starting from 0 (0, 3, 6, 9, 12 and 15). Observe the spectrum after equalization and compare how it changes while increasing the delay and when the delay gain changes.

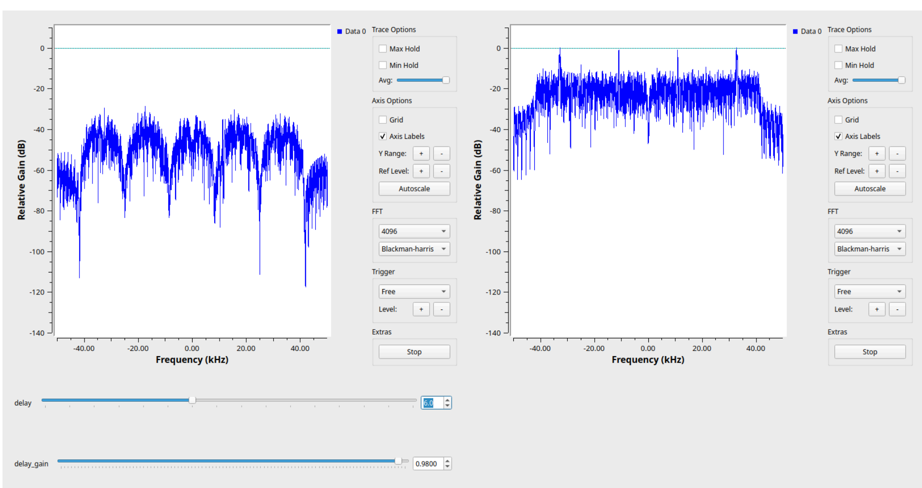
0:



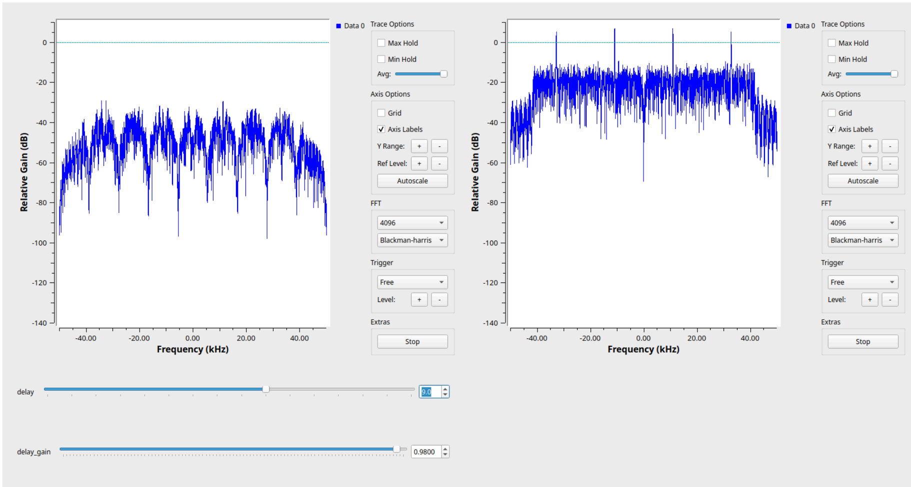
3:


Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-140
-40.00
-20.00
0.00
20.00
40.00
Frequency (kHz)
Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-140
-40.00
-20.00
0.00
20.00
40.00
Frequency (kHz)
delay
3.0
delay_gain
0.9800

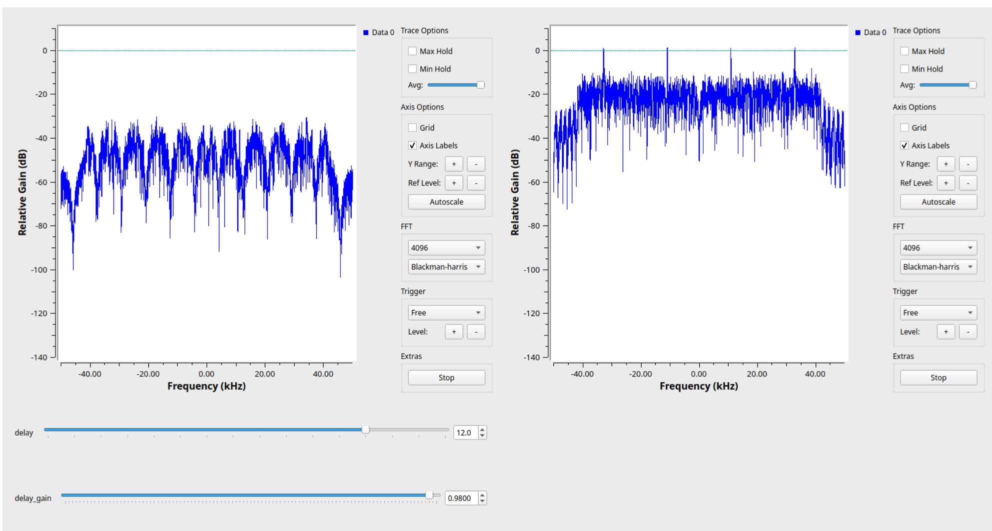
6:


Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-140
-40.00
-20.00
0.00
20.00
40.00
Frequency (kHz)
Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-140
-40.00
-20.00
0.00
20.00
40.00
Frequency (kHz)
delay
delay_gain
0.9800

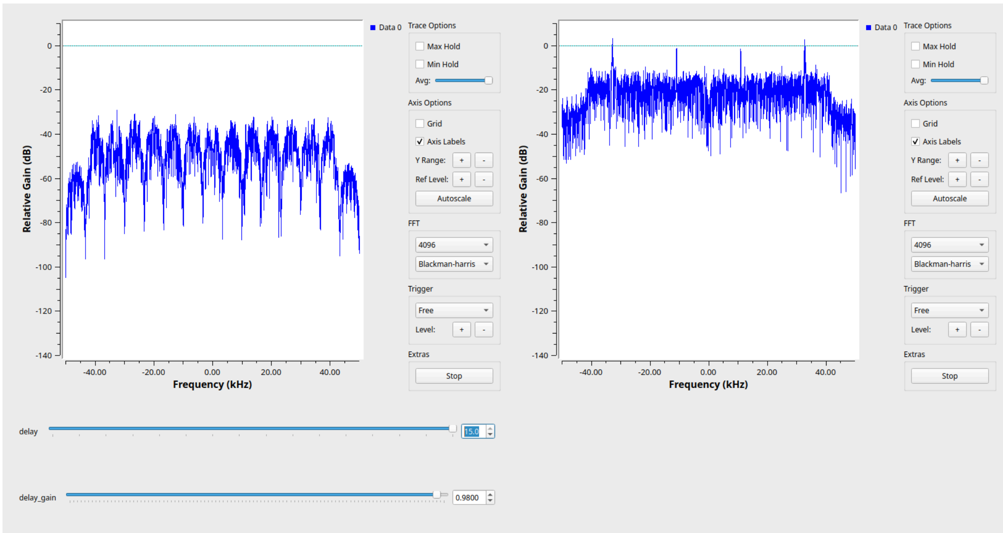
9:


Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
Frequency (kHz)
delay
delay_gain
0.9800

12:

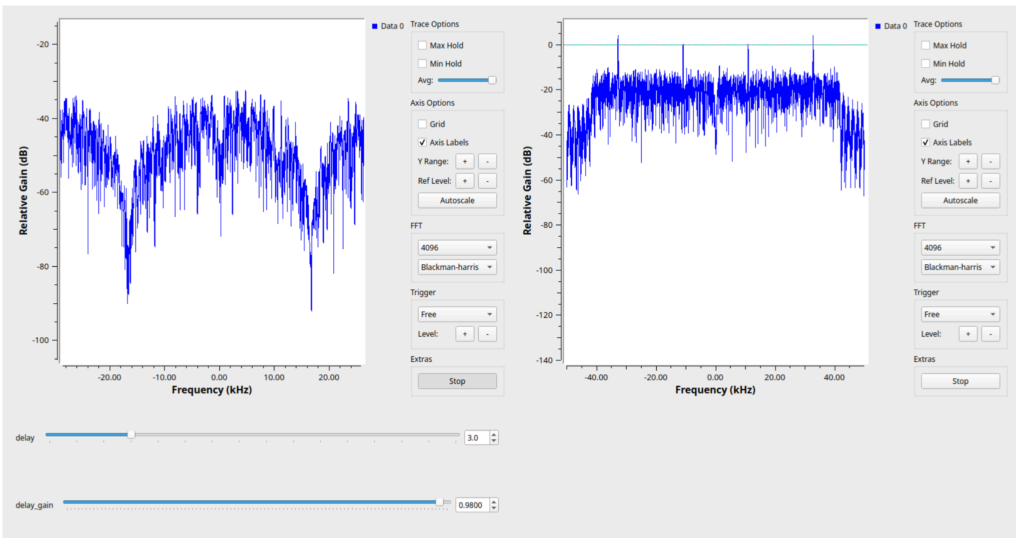

Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
Frequency (kHz)
delay
12.0
delay_gain
0.9800

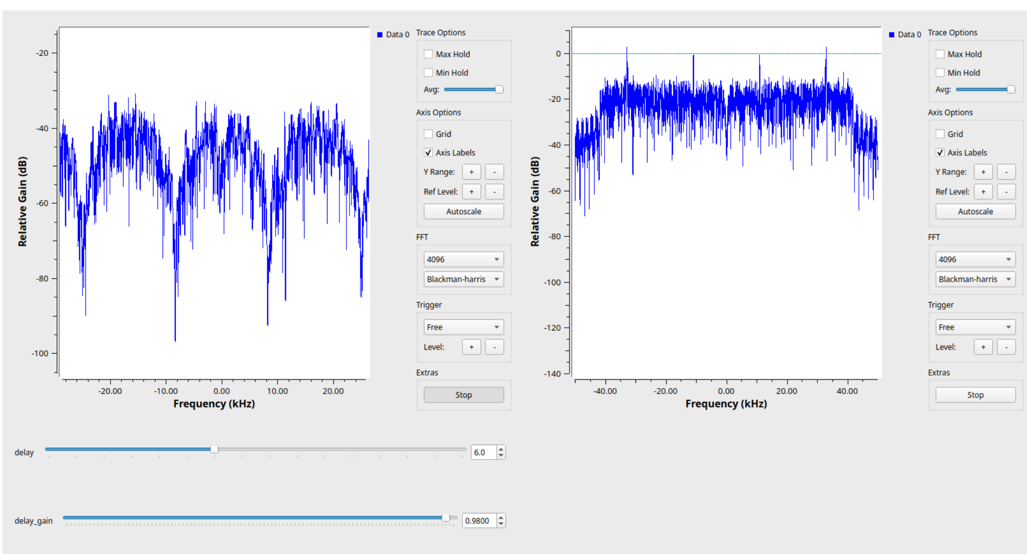
15:

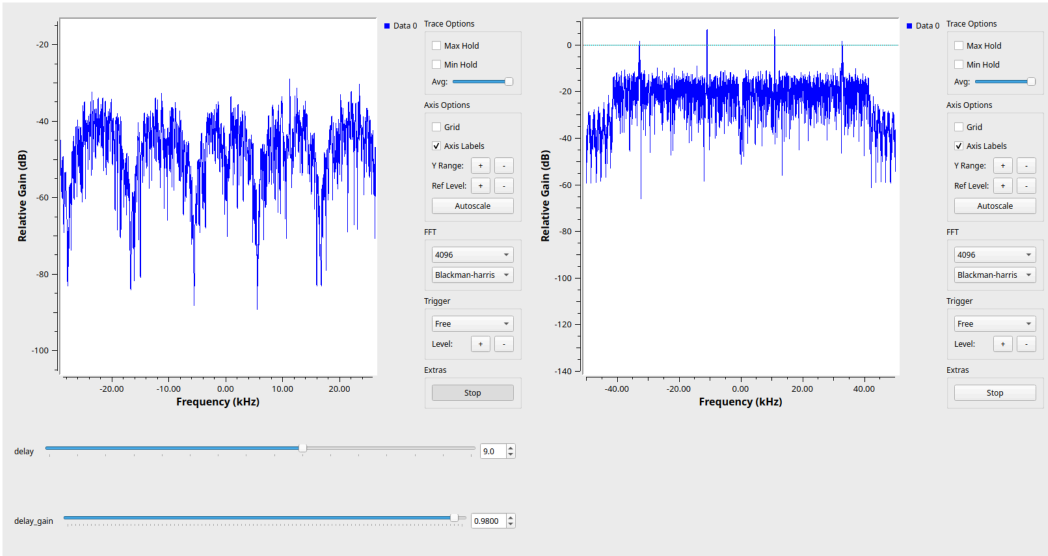


Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
Frequency (kHz)
delay
delay_gain
0.9800

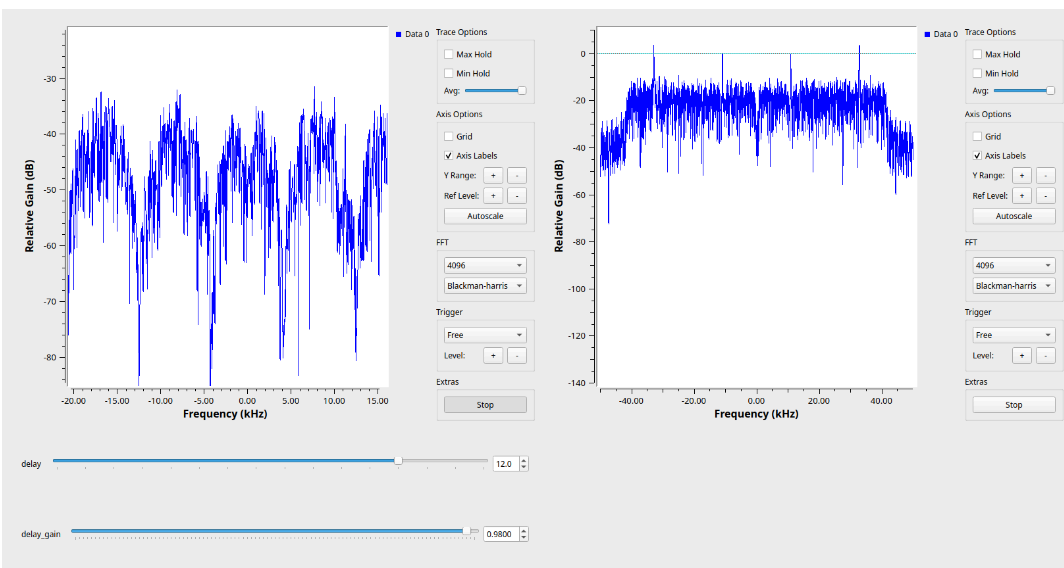
4. Measure the coherence bandwidth, i.e., the spacing between nulls within the frequency spectrum in terms of frequency



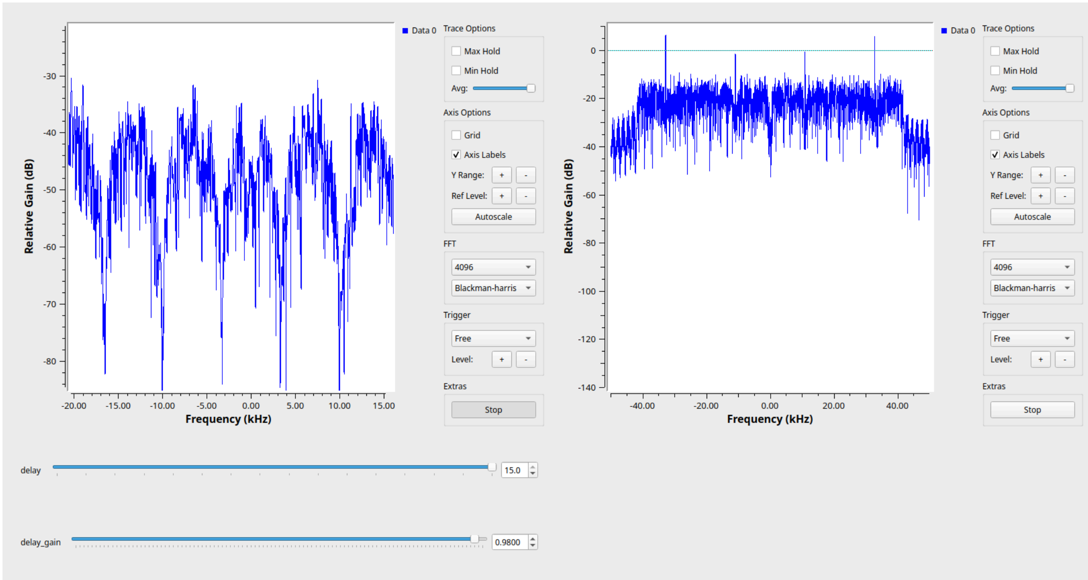




Relative Gain (dB)
-20
-40
-60
-80
-100
-20.00
-10.00
0.00
10.00
20.00
Frequency (kHz)
Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-140
-40.00
-20.00
0.00
20.00
40.00
Frequency (kHz)
Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
delay
9.0
delay_gain
0.9800



Relative Gain (dB)
-30
-40
-50
-60
-70
-80
-20.00
-15.00
-10.00
-5.00
0.00
5.00
10.00
15.00
Frequency (kHz)
Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-140
-40.00
-20.00
0.00
20.00
40.00
Frequency (kHz)
Data 0
Trace Options
Max Hold
Min Hold
Avg:
Axis Options
Grid
Axis Labels
Y Range:
Ref Level:
Autoscale
FFT
4096
Blackman-harris
Trigger
Free
Level:
Extras
Stop
delay
12.0
delay_gain
0.9800



Write-up for section 2.1

1. Provide the screenshots for step 2 and 3 for both single-carrier and multi-carrier cases. There should be 3 screenshots for step 2 and 6 screenshots for step 3 in each of the cases, totalling 18 screenshots.

2. Comment on how the spectrum before and after equalization is affected by varying the delay and delay gain and how single-carrier and multi-carrier perform in each scenario.

The effect we can observe as we vary the delay and delay gain in the spectrum before is there begins to appear more side lobes almost proportionally to the delay factor. In the single-carrier these side lobes in the before spectrum are attenuated; in contrast, the side lobes for the spectrum before equalization of multi-carrier are all at the same attenuation. In the spectrum after equalization for single-carriers there also appears to be sidelobes while in the multi-carrier spectrum after equalization appears to retain its shape regardless of delay gain. For this reason, the multi-carrier spectrum after equalization appears to not have much change regardless of any delay while the single-carrier spectrum after equalization is impacted by the delay.

3. Which system was more resilient to the variation in delays. Can you think of the reason why one performed better over the other?

As mentioned in the previous question, the multi-carrier is more resilient to the variation in delays this is because of its ability to mitigate multipath delay spread effectively, simpler and more efficient equalization. In the single-carrier the signal is affected by multipath delay and leads to significant ISI which will degrade the performance. Multi-carrier will divide the signal into narrow sub-channels and for other reasons it confines the multipath effects to a smaller portion of the signal which will lead to simplifying the equalization process and improving delay performance.

4. Can you reason why the spacing is referred to as coherence bandwidth?

The reason why the spacing is referred to as coherence bandwidth is it demonstrates the importance of aligning the sub-carrier spacing with the channel's frequency coherence properties. This alignment ensures that each sub-carrier experiences flat fading, simplifying equalization, minimizing ISI, and making efficient use of the available spectrum.

5. Find the relation between the coherence bandwidth and delay values (think the delay in terms of samples and sampling rate).

Coherence bandwidth is inversely proportional to the multipath delay. In terms of coherence bandwidth, it indicates the range of frequencies over which the channel can be considered to have a constant gain and phase response.

Delay Spread in Seconds:

B_coherence = 1/t_delay

Delay Spread in samples:

D = t_delay * fs

Coherence Bandwidth in Terms of Samples and Sampling Rate:

B_coherence = fs / D

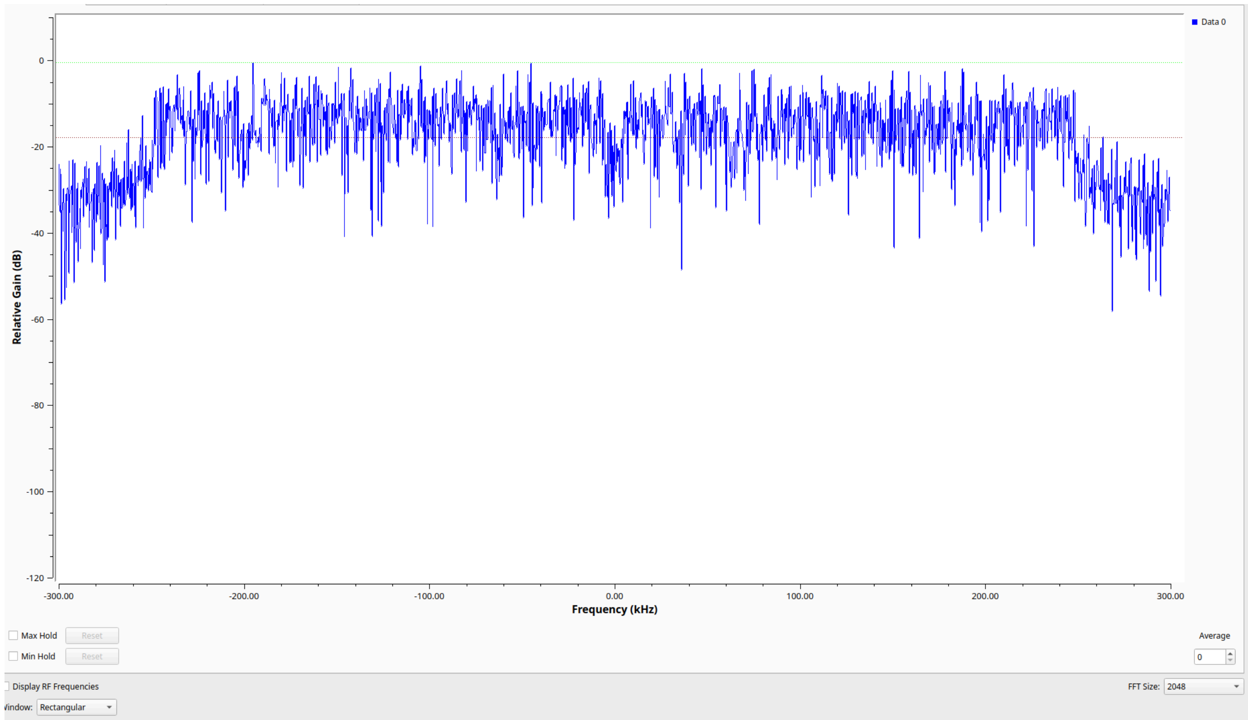
2.2 OFDM

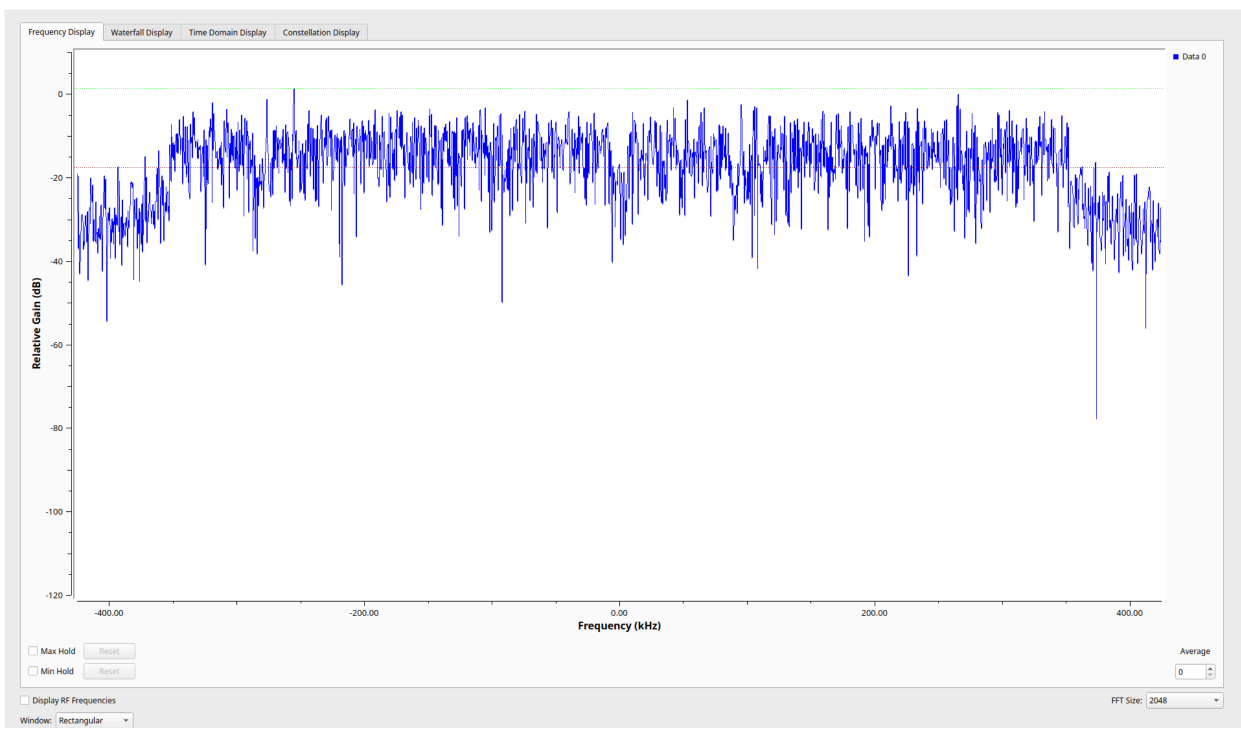
For this part you will need to use two computers with each having one SDR plugged in for running transmitter and receiver code separately. Note that receiver code will capture data for 20 seconds and then timeout. There won't be any data after 20 seconds from running the code. This is done so that we can measure the throughput. The code will save a file named *rx_file.txt* in the same folder as the source code.

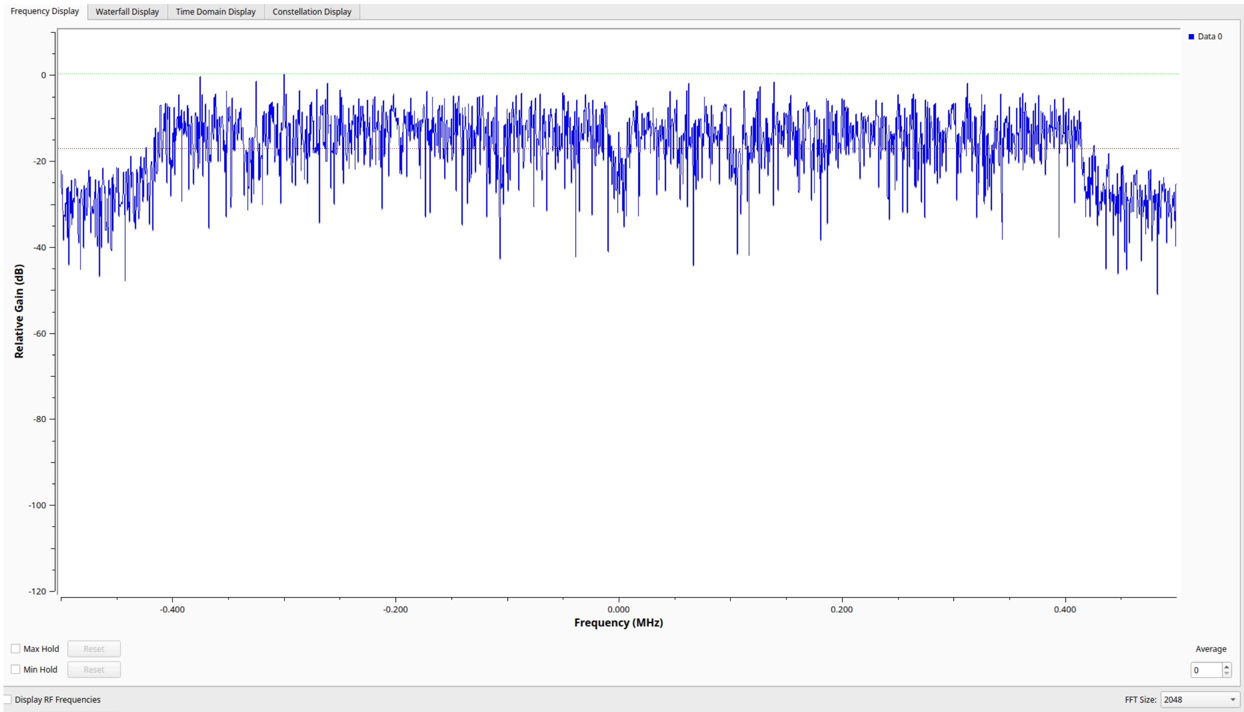
Throughput is obtained by measuring the size of the file and dividing it by 20 seconds. This will give you the approximate throughput of the system. Note that we don't have any MAC layer or higher layer capabilities, so there is no retransmission of lost packets or erroneous packets. So we only obtain the raw throughput rather than the actual data throughput which takes the channel effects into account.

1. Open GNURadio by running gnuradio − companion in two PCs and run *ofdm_tx.grc* code in one PC and *ofdm_rx.grc* code in the other PC.

2. Vary the sampling rate from the *ofdm_tx.grc* GNURadio GUI by changing the *samp_rate* variable and observe the spectrum of the signal noting down the bandwidth. Capture the screenshots with sampling rates 600KHz, 850KHz and 1MHz.


Data 0
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-300.00
-200.00
-100.00
0.00
100.00
200.00
300.00
Frequency (kHz)
Max Hold
Reset
Min Hold
Reset
Average
0
Display RF Frequencies
FFT Size: 2048
Rectangular

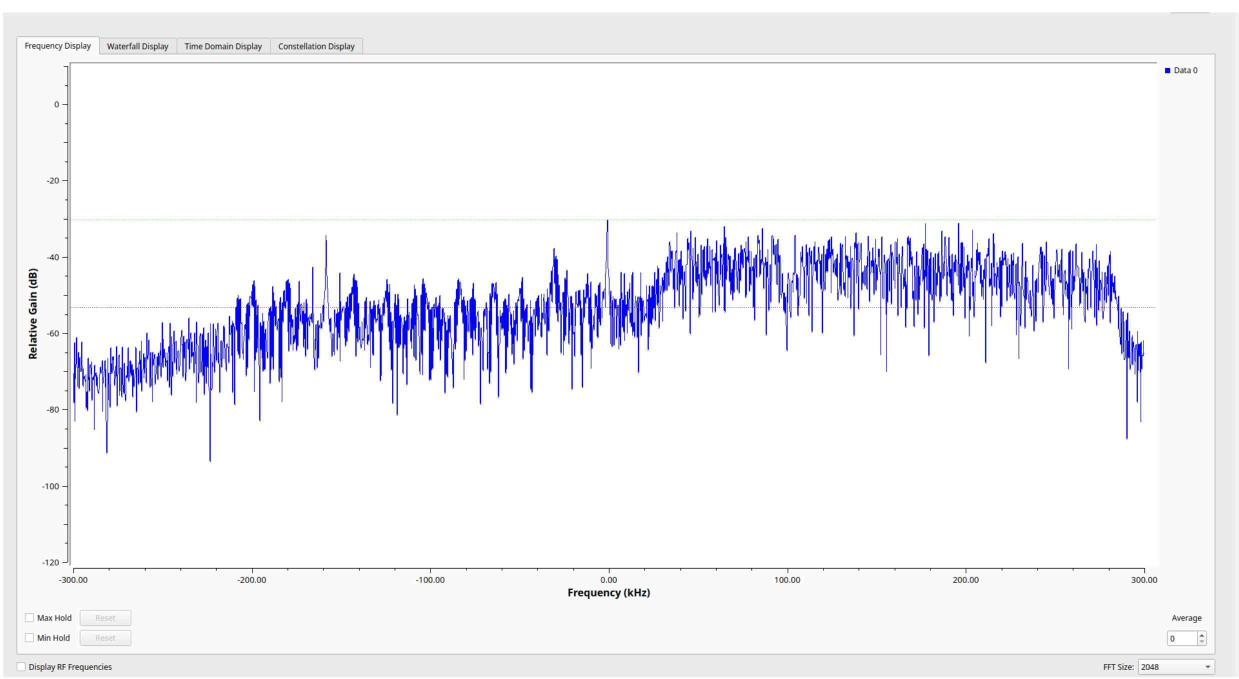

Frequency Display
Waterfall Display
Time Domain Display
Constellation Display
Data 0
Relative Gain (dB)
0
-20
-40
-60
-80
-100
-120
-400.00
-200.00
0.00
200.00
400.00
Frequency (kHz)
Max Hold
Reset
Min Hold
Reset
Average
0
Display RF Frequencies
FFT Size: 2048
Window: Rectangular



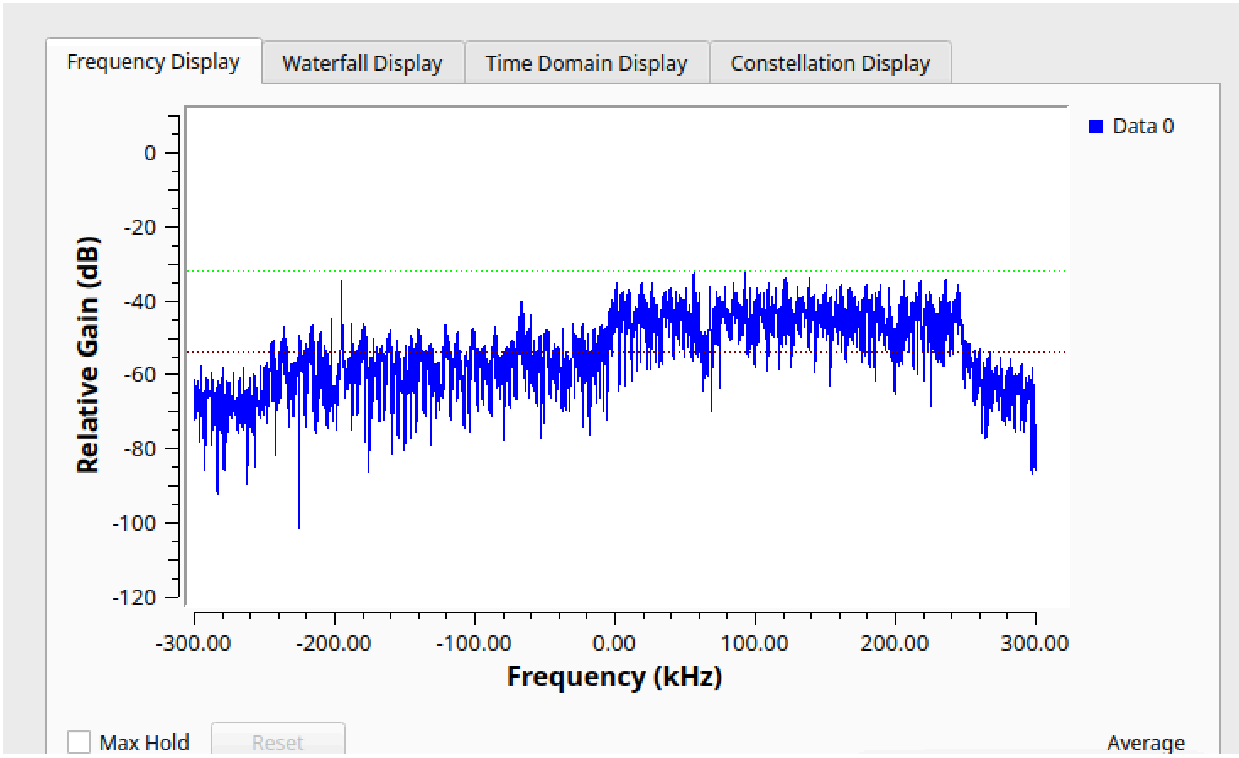
3. Now change the *occupied_carriers* in GNURadio GUI and rerun the flowgraph in both TX and RX PCs. You can observe the default occupied sub-carriers will range from -26 to 26 with some sub-carriers allocated for pilot symbols in-between.

   a) Remove the subcarriers from -26 to -1. Take a screenshot of the spectrum from the TX side and RX side. Measure the throughput.

   rx



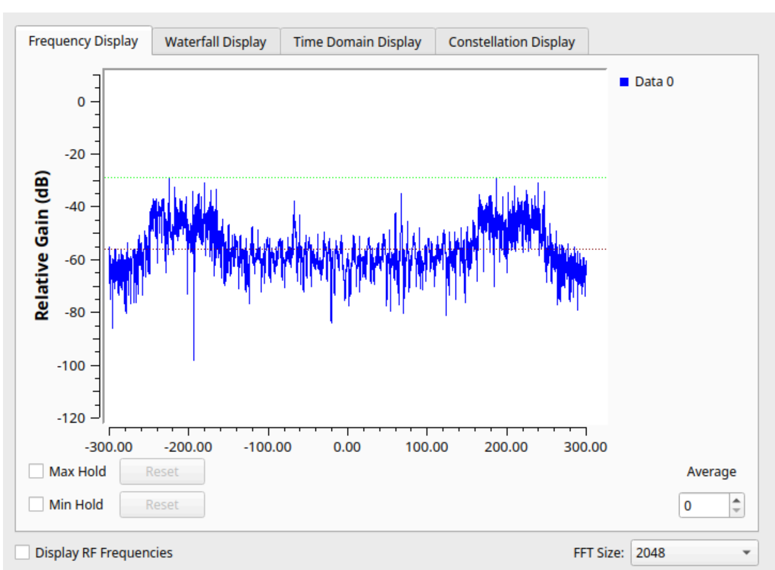
tx



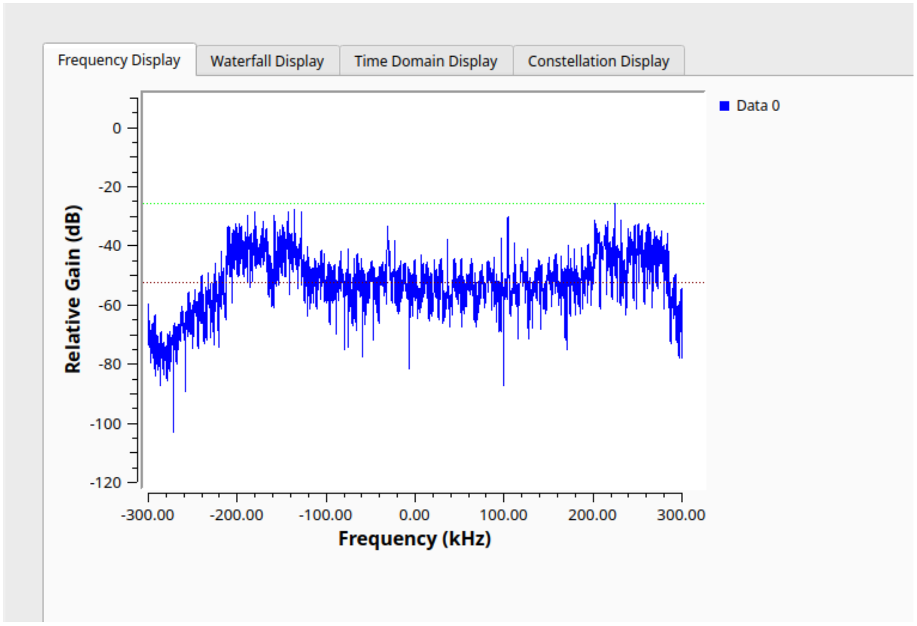
6.7kB / 20 s = .335 kB/s

b) Remove the subcarriers -17 to -1 and 1 to 17 and keep the remaining, from -26 to -18 and 18 to 26. Note, -21 and 21 still belong to *pilot_carriers*, so don't include them in the list as well. Take a screenshot of the spectrum from the TX and RX side. Measure the throughput.

tx



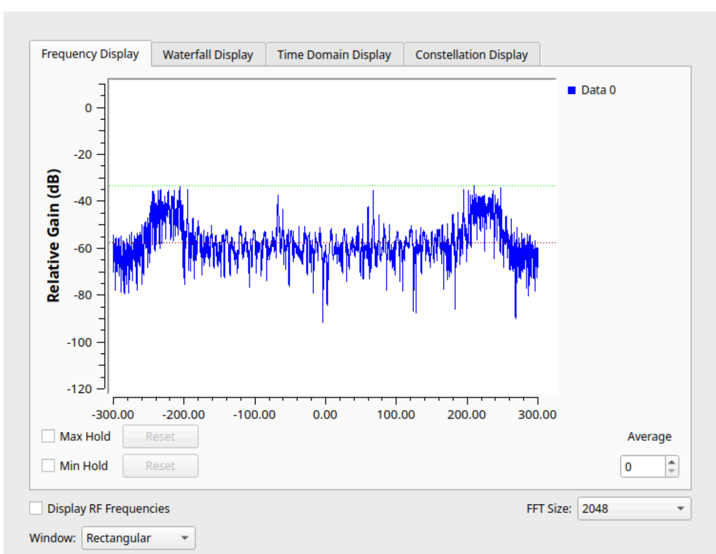
rx



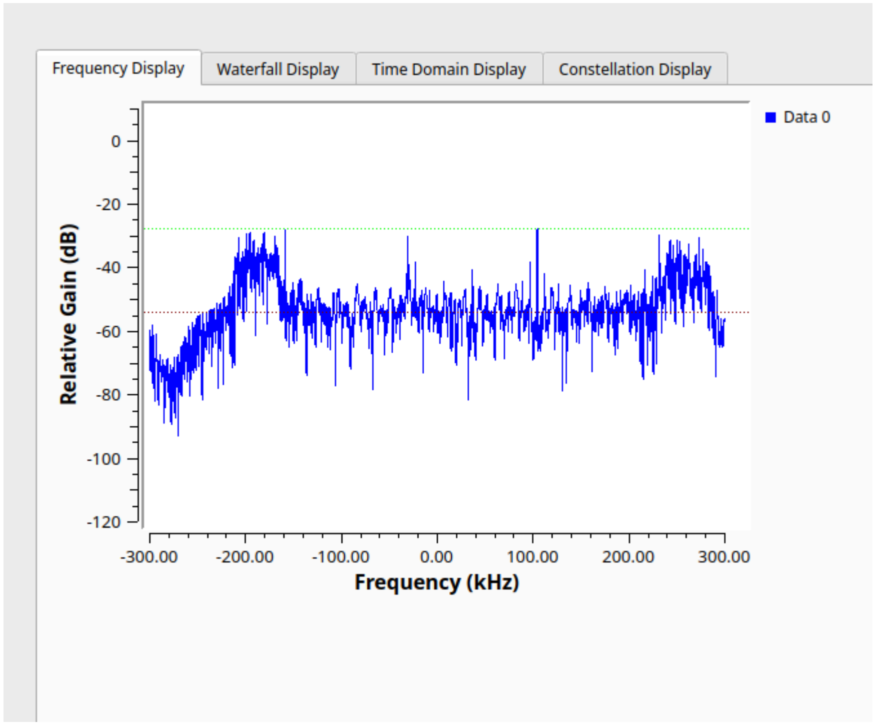
428kB / 20s = 21.4 kB/s

c) Remove the subcarriers -20 to 20 and keep the remaining, from -26 to -22 and 22 to 26. Take a screenshot of the spectrum from the TX and RX side. Measure the throughput.

tx



rx



291.7kB / 20 = 14.585 kB/s

Size of Rx file/20 seconds

Write-up for section 2.2

1. Report the screenshots for the three sampling rates mentioned above in step 2. Find the relation between bandwidth and sampling rate you observed from step 2.

   As the bandwidth of the signal increases, the sampling rate required to adequately sample the signal would also increase.

2. Report three screenshots from the three subparts of step 3. How do you relate the changes in subcarriers and the spectrum that you observe?

   More subcarriers should result in a more densely populated spectrum, while fewer subcarriers should lead to a sparser spectrum with fewer frequency components.

3. Report the throughput obtained in each of the measurements. What can you infer from these measurements?

   Less subcarriers = faster throughput (affects the efficiency of data transmission)

   With fewer subcarriers, each subcarrier carries more data, potentially leading to higher throughput per subcarrier (but less robustness).