Project 2

Analysis of Fourier Transforms on Audio Signals Through Carrier Waves

ECE 3793-001

Russell Kenny

Ged Miller

Introduction:

When sending a low frequency signal, the signal must be combined with a high frequency "carrier wave signal" to combat the physical limitations of creating an array large enough to send very low frequency signals. The signal must the be filtered, to filter out extraneous signals and other signals on the carrier wave. This report covers the process in which an input signal is filtered, modulated with a carrier signal, then converted back to the original signal using the properties of Fourier Transforms and trigonometric functions mathematically, then in a real world scenario using MATLAB software.

Part 1 a): Derivation of the Transmitter

Time Domain Signal:

$$x(t) = cos(2\boldsymbol{\pi}500t)$$

Fourier Transform:

$$x(t) = \cos(2\pi F_0 t) = \cos(\omega_0 t) \qquad X(F) = \frac{1}{2}\left[\delta(F - F_0) + \delta(F + F_0)\right]$$

Frequency Domain Signal:

$$X(F) = \frac{1}{2}[\boldsymbol{\partial}(F - 500) + \boldsymbol{\partial}(F + 500)]$$

Carrier Signal Time Domain:

$$\mathrm{x_c}(t) = cos(2\boldsymbol{\pi}\mathrm{F_c}t)$$

Fourier Transform:

$$x(t) = \cos(2\pi F_0 t) = \cos(\omega_0 t) \qquad X(F) = \frac{1}{2}\left[\delta(F - F_0) + \delta(F + F_0)\right]$$

Frequency Domain Carrier Signal:

$$\mathrm{X_C}(F) = \frac{1}{2}[\boldsymbol{\delta}(F - \mathrm{F_C}) + \boldsymbol{\delta}(F + \mathrm{F_C})]$$

Time Domain Transmitter Signal:

$$\mathrm{x_{TX}}(t) = x(t)\mathrm{x_c}(t)$$

$$\mathrm{x_{TX}}(t) = \frac{1}{2}[cos(2\boldsymbol{\pi}500t + 2\boldsymbol{\pi}\mathrm{F_C}t) + cos(2\boldsymbol{\pi}500t - 2\boldsymbol{\pi}\mathrm{F_c}t)]$$

Fourier Property:

Multiplication	$x(t)y(t)$	$X(F)*Y(F)$

Frequency Domain Transmitter Signal:

$$\mathrm{X_{TX}}(F) = \frac{1}{4}[\boldsymbol{\partial}(F - 500 \ - \mathrm{F_C}) + \boldsymbol{\partial}(F + 500 \ + \ \mathrm{F_C}) + \boldsymbol{\partial}(F - 500 + \mathrm{F_C}) + \boldsymbol{\partial}(F + 500 - \mathrm{F_C})$$

The properties and transforms displayed above demonstrate how signals can be converted to the frequency domain and back to the time domain. Manipulation of Fourier transforms allows signals to be combined for different purposes such as creating new signals in either the frequency or time domains. **X_{TX}(F) is shown to be related to X(F) since it is a scaled and shifted signal of X(F).**

Audio Waveform Signal Time Domain:

$$\mathrm{x_a}(t) = cos(2\boldsymbol{\pi}22000t)$$

Audio Waveform Signal Frequency Domain:

$$\mathrm{X_a}(F) = \frac{1}{2}[\boldsymbol{\delta}(F - 22000) + \boldsymbol{\delta}(F + 22000)]$$

Time Domain Transmitter Signal:

$$\mathrm{x_{TX}}(t) = \mathrm{x_a}(t)\mathrm{x_c}(t)$$

$$\mathrm{x_{TX}}(t) = \frac{1}{2}[cos(2\boldsymbol{\pi}22000t + 2\boldsymbol{\pi}\mathrm{F_C}t) + cos(2\boldsymbol{\pi}22000t - 2\boldsymbol{\pi}\mathrm{F_c}t)]$$

Frequency Domain Transmitter Signal:

$$\mathrm{X_{TX}}(F) = \frac{1}{4}[\boldsymbol{\partial}(F - 22000 \ - \mathrm{F_C}) + \boldsymbol{\partial}(F + 22000 \ + \ \mathrm{F_C}) + \boldsymbol{\partial}(F - 22000 + \mathrm{F_C}) + \boldsymbol{\partial}(F + 22000 - \mathrm{F_C})$$

The new transmitter signal above, in both the time and frequency domains, is a shifted and scaled version of the audio waveform signal limited to 22,000 Hz. Since the signal of X_a(F) is limited by 22,000 Hz, if the signal was only in the right plane, the difference between the smallest and largest frequency would be 22,000 Hz. For the X_{TX}(F) signal, the bandwidth would be 22,000 Hz if the initial condition F_C were set to 0. Thus, the bandwidth of X_a(F) and X_{TX}(F) would be the same when $F_c = 0$.

Part 1 b): Derivation of the Receiver

Time Domain Interference Signal:

$$\mathrm{x}_1(t) = cos(2\boldsymbol{\pi}\mathrm{F}_c t)$$

Frequency Domain Interference Signal:

$$\mathrm{X}_1(F) = \frac{1}{2}[\boldsymbol{\delta}(F - \mathrm{F}_\mathrm{C}) + \boldsymbol{\delta}(F + \mathrm{F}_\mathrm{C})]$$

Time Domain Receiver Signal:

$$\mathrm{x}_{\mathrm{RX}}(t) = \mathrm{x}_{\mathrm{TX}}(t) + \mathrm{x}_1(t)$$

$$\mathrm{x}_{\mathrm{RX}}(t) = 2cos\left(\frac{2\boldsymbol{\pi}22000\mathrm{t} - 2\boldsymbol{\pi}\mathrm{F}_c\mathrm{t}}{2}\right)cos\left(\frac{2\boldsymbol{\pi}22000\mathrm{t} + 2\boldsymbol{\pi}\mathrm{F}_c\mathrm{t}}{2}\right)$$

Fourier Property:

Linearity	$ax(t)+by(t)$	$aX(F)+bY(F)$

Frequency Domain Receiver Signal in terms of X_{TX}(F):

$$\mathrm{X}_{\mathrm{RX}}(\mathrm{F}) = \mathrm{X}_{\mathrm{TX}}(\mathrm{F}) + \mathrm{X}_1(\mathrm{F})$$

$$\mathrm{X}_{\mathrm{RX}}(F) = \left[\frac{1}{4}[\boldsymbol{\partial}(F - 22000 \ - \mathrm{F}_\mathrm{C}) + \boldsymbol{\partial}(F + 22000 \ + \ \mathrm{F}_\mathrm{C}) + \boldsymbol{\partial}(F - 22000 + \mathrm{F}_\mathrm{C}) + \boldsymbol{\partial}(F + 22000 - \mathrm{F}_\mathrm{C})]\right] + \left[\frac{1}{2}[\boldsymbol{\delta}(F - \mathrm{F}_\mathrm{C}) + \boldsymbol{\delta}(F + \mathrm{F}_\mathrm{C})]\right]$$

Frequency Domain Receiver Signal in terms of X_a(F):

$$\mathrm{X}_{\mathrm{RX}}(\mathrm{F}) = \mathrm{X}_\mathrm{a}(\mathrm{F}) + \mathrm{X}_1(\mathrm{F})$$

$$\mathrm{X}_{\mathrm{RX}}(F) = \left[\frac{1}{2}[\boldsymbol{\delta}(F - 22000) + \boldsymbol{\delta}(F + 22000)]\right] + \left[\frac{1}{2}[\boldsymbol{\delta}(F - \mathrm{F}_\mathrm{C}) + \boldsymbol{\delta}(F + \mathrm{F}_\mathrm{C})]\right]$$

Down Converted Time Domain Signal:

$$\mathrm{x}_\mathrm{d}(t) = \mathrm{x}_{\mathrm{RX}}(t)\mathrm{x}_\mathrm{c}(t)$$

$$\mathrm{x}_\mathrm{d}(t) = \left[2cos\left(\frac{2\boldsymbol{\pi}22000\mathrm{t} - 2\boldsymbol{\pi}\mathrm{F}_c\mathrm{t}}{2}\right)cos\left(\frac{2\boldsymbol{\pi}22000\mathrm{t} + 2\boldsymbol{\pi}\mathrm{F}_c\mathrm{t}}{2}\right)\right][cos(2\boldsymbol{\pi}\mathrm{F}_c t)]$$

Frequency Domain of Down Converted Signal:

$$\mathrm{X_d(F) = X_{RX}(F)*X_c(F)}$$

$$\mathrm{X_d}(F) = \left[\left[\frac{1}{2}[\boldsymbol{\delta}(F-22000)+\boldsymbol{\delta}(F+22000)]\right]+\left[\frac{1}{2}[\boldsymbol{\delta}(F-\mathrm{F_C})+\boldsymbol{\delta}(F+\mathrm{F_C})]\right]\right] * \left[\frac{1}{2}[\boldsymbol{\delta}(F-\mathrm{F_C})+\boldsymbol{\delta}(F+\mathrm{F_C})]\right]$$

Frequency Domain Filter Signal:

$$H(F) = rect\left(\frac{\mathrm{F}}{44000}\right)$$

Fourier Transform:

$$x(t) = W\mathrm{sinc}(Wt) \qquad\qquad X(F) = \mathrm{rect}(F/W)$$

Time Domain Filter Signal:

$$h(t) = 44000sinc(44000t)$$

Time Domain Output Audio Signal:

$$\mathrm{x_{out}}(t) = \mathrm{x_d}(t)h(t)$$

$$\mathrm{x_{out}}(t) = \left[\left[2cos\left(\frac{2\boldsymbol{\pi}22000\mathrm{t}-2\boldsymbol{\pi}\mathrm{F_c t}}{2}\right)cos\left(\frac{2\boldsymbol{\pi}22000\mathrm{t}+2\boldsymbol{\pi}\mathrm{F_c t}}{2}\right)\right][cos(2\boldsymbol{\pi}\mathrm{F_c}t)]\right][44000sinc(44000t)]$$

Filtered Frequency Domain of Output Audio Signal:

$$\mathrm{X_{out}}(F) = \frac{1}{4}[\boldsymbol{\delta}(F-22000\mathrm{F_c})+\boldsymbol{\delta}(F+22000\mathrm{F_c})]$$

Final Time Domain Output Audio Signal:

$$\mathrm{x_{out}}(t) = \frac{1}{2}cos(2\boldsymbol{\pi}22000\mathrm{F_c}t)$$

The final output audio signal has been filtered through the function h(t) and has resulted in the output above considering the $x_1(t)$ is within the bounds of the filter function. The final audio output is a scaled version of $x_a(t)$ and thus it is evident that the filtering process has worked.

Part 2: Demodulating Modulated CW Signals

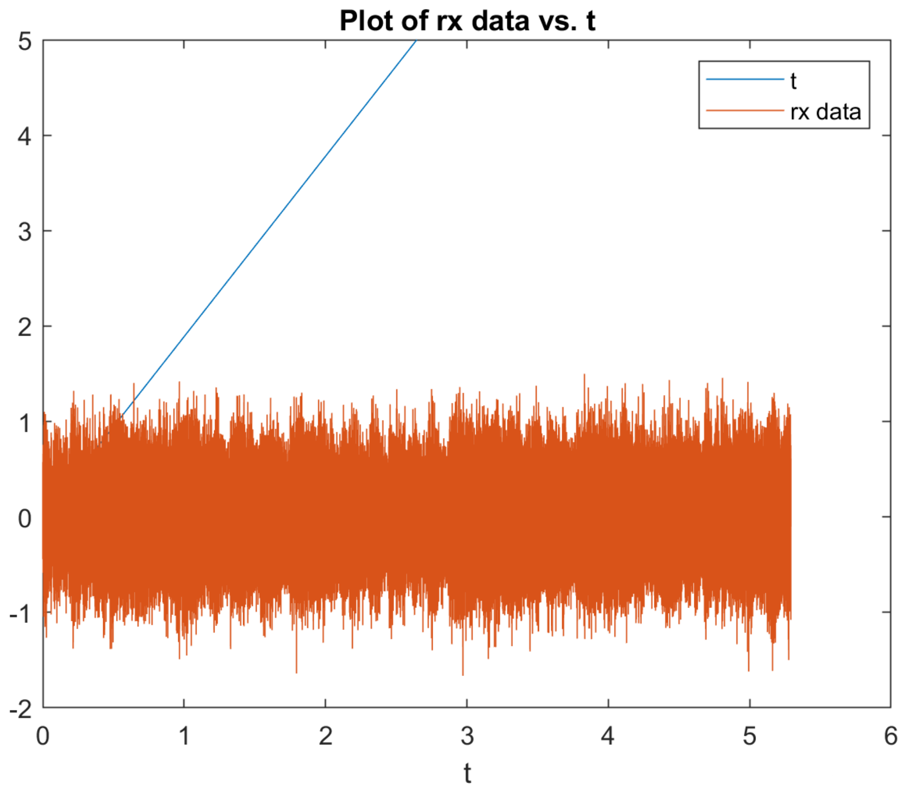

Figure 1: vector rx_data vs. vector t showing audio data over about 5 seconds.

After plotting the provided audio data vs a vector of the size of the data, no useful data can be derived from the relation between the two vectors, however, the plot itself shows the length of the audio stored within the vector which appears to be about 5.29 seconds of audio.

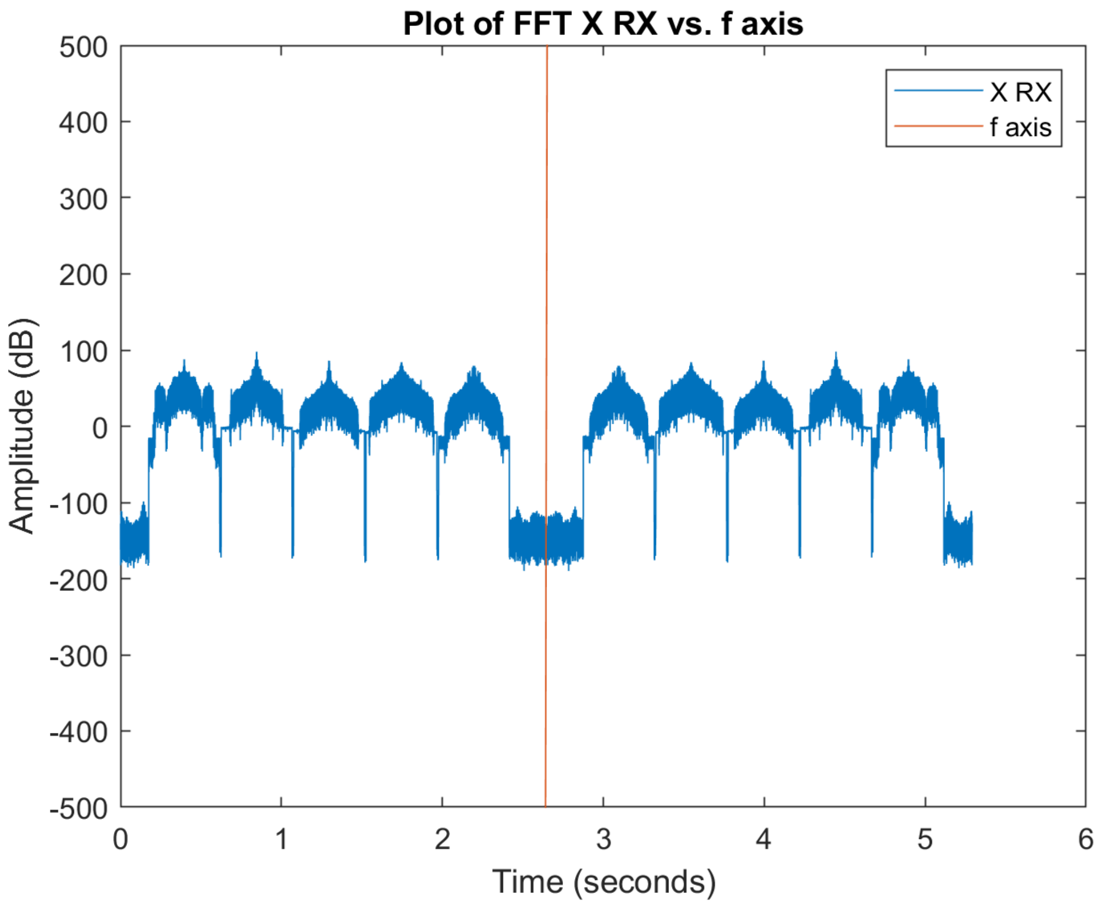

Figure 2: Plot of the FT of X_XR vs. f_axis depicting 5 signal humps on either side of the f_axis.

The plotted signal does not have any meaningful signals within 20kHz of the f_axis at F=0, after, signal humps appear on either side of the f_axis. They are symmetric and reach almost 100 dB. Each hump appears to be centered upon frequencies at an interval of 45000 Hz away from the f_axis. These values appear to 45000 Hz, 90000 Hz, 135000, 180000 Hz, and 225000 Hz on each side of the f_axis respectively. These humps correspond to the carrier frequencies given because they represent the audio data stored in at each frequency increment as described.

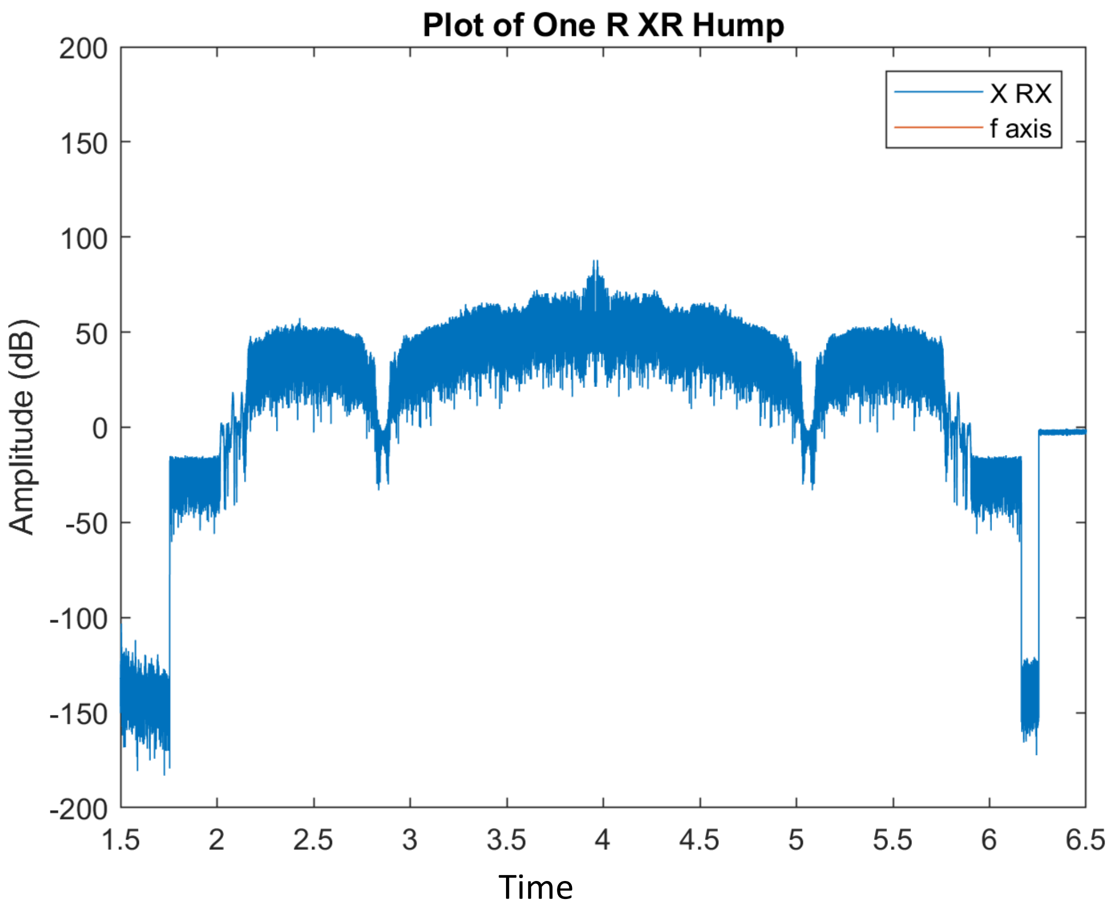

Figure 3: Zoomed in plot of FFT of rx_data on one hump of signal.

The hump displayed in Figure 3 appears to have three humps withing it that are symmetric to one another. This hump matches the filtered region as described in part one. It has a rectangular shape and matches the height and width as allowed by the filter function from part 1.

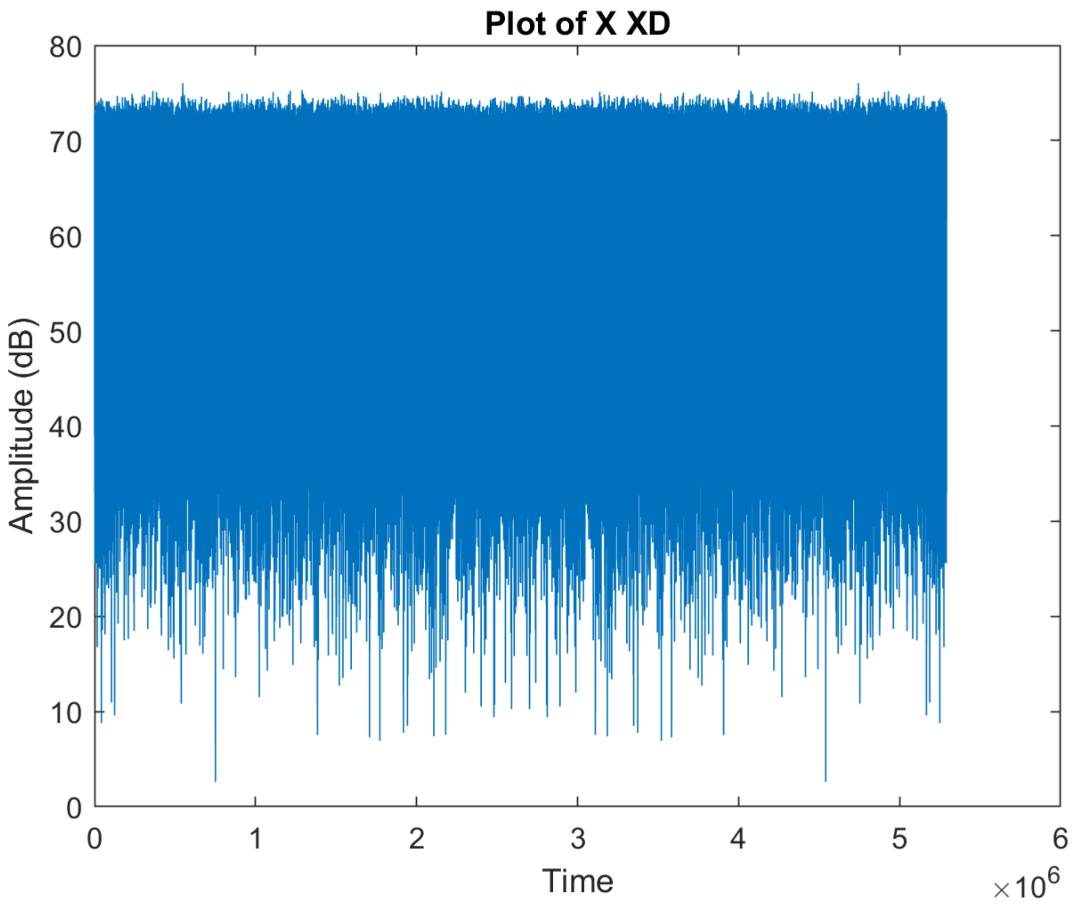

Figure 4: Plot of X_d(F) in db over time in microseconds.

The signal in figure 4 has a lot of static and is not centered at F = 0 as the $x_{RF}(t)$ signal was. When played, the signal sounds like static and no coherent audio can be heard. This faulty signal is likely researcher error and occurred when the deriving the $x_d(t)$ function in MatLab.

Conclusion:

This lab experiment covered the using carrier wave signals to transmit low frequency signals along distances. It also served to demonstrate how to apply filters to carrier waves and how to parse out the audio signals from a filter signal and carrier wave. Most obejectives were complete, however, the lab technicians failed to convert the calculations from part 1 of the lab to workable code in the second part of the lab. The technician was unable to get code to work that required the multiplication of functions. Specific operators such as (.*) were tried but did not work. This lab also demonstrated how to convert signals back and fourth from the time and frequency domains successfully.

Code Appendix

```
clear all

close all

clc


load('Project_2_Data.mat'); %loads provided matlab file


t = 0:(1/Fs):5; %% Set the end of the vector to be 5 since ending at 1 like rc_data does not show up on plot


%plot(t)


%hold on


%plot(rx_data)


%hold off


%legend('t','rx data')


%xlabel('t')


%title('Plot of rx data vs. t')


%sz = size(rx_data) used to find size of vector


%length(rx_data)


%down sampling given vector


rx_downsample = rx_data(1:interp_factor:end);


%soundsc(rx_downsample,44100); %plays signal... very pleasant indeed


X_RX = fftshift(fft(rx_data)); %computes FT of rx_data and shifts data to right axis


X_RX = mag2db(abs(X_RX));
```

```
%determines the frequency axis of the shifted X_RX vector
f_axis = -Fs/2 + (0:(length(X_RX)-1))*Fs/length(X_RX);

%plot(X_RX)

%hold on

%plot(f_axis)

%hold off

%ylim([-200 200])

%xlim([150000 650000])

%legend('X RX','f axis')

%xlabel('Time (seconds)')

%ylabel('Amplitude (dB)')

%title('Plot of One R XR Hump')

%title('Plot of FFT X RX vs. f axis')

Fc = 45000;

%I want to muliply rx_data and x_c but I can't for some reason
%I know this is wrong but it'll have to do.

x_d = cos(2*pi*22000*rx_data);



x_d_downsample = x_d(1:interp_factor:end);
```

```
%soundsc(x_d_downsample,44100)

X_XD = fftshift(fft(x_d));

X_XD = mag2db(abs(X_XD));

%plot(X_XD)

%xlabel('Time')

%ylabel('Amplitude (dB)')

%title('Plot of X XD')

L_h = 101;

t_h = -Ts*((L_h-1)/2):Ts:Ts*(L_h-1)/2; %this line is giving me an error and I don't know why

x_h = 44000*sinc(44000*101); %I can't get a function to work

plot(x_h)

xlabel('Time')

ylabel('Amplitude (dB)')

title('Plot of X XD')
```