EC 303P - PRINCIPLES OF COMMUNICATION SYSTEMS LAB
PROJECT

Anti Jamming Techniques

Anurag Pendyala – IMT2017504
K. Sailesh – IMT2017524

Contents

1. ABSTRACT

Wireless networks have transformed the way people communicate. They have helped things like broadcasting, long distance communication become a reality. Wireless networks are more vulnerable to attacks/distortion because the medium of transport is air. One does not have much control on this aspect as anyone with a basic system like a Raspberry Pi can start jamming the network

Jamming Techniques include basic disrupting the original signal to complete manipulation and take over of the network. Advanced Jamming Techniques include some famous ones like follow-on and channel hop jammers. Follow on hopping jammer hops over all the available channels and jams each for a short time. Channel Hop jammer proactively jams multiple channels at the same time.

Thus adding an extra layer of security to the network helps improve the comunication.

2. PROBLEM STATEMENT

Objective : Design an FM modulator and demodulator system that resists noise and jamming.

Steps :

- Choose a wav file. An instrumental file is preferred.
- Modulate it and then demodulate it and the play it.
- Handle any kind of noise being added after modulation.
- Now add a FM signal instead of a normal noise signal to test the anti jamming.
- Test the capture effect and explain the approaches on how to avoid it.
- Play the received signal again.

The report includes the following details :

- Different trials with you approaches on how to handle noise and jamming signals.
- Difficulties faced in all trials.
- Ideas and Approaches for every problem face.

The Matlab Code is given in the Appendix.

3. Approaches

It is important to avoid any additional noise being added through the channel. There may be a lot of data loss if the noise is not handled carefully. Someone might find the frequency in which the user is transmitting and then send a high power signal at that frequency. This signal is known as a jamming signal.

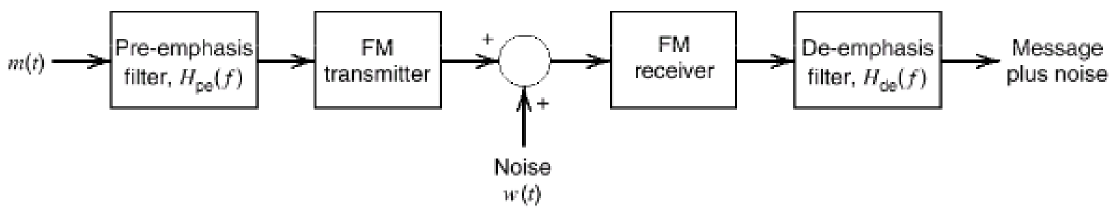
The approaches to avoid the problems and their observations have been explained clearly in the respective subsections.

3.1. Noise Reduction

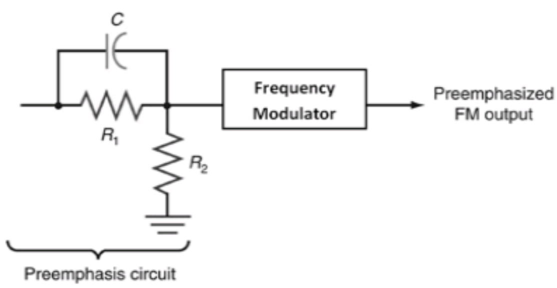
Emphasis in telecommunication refers to enhancement/boosting of frequencies in signal which are most susceptible to noise in a medium. The noise suppression ability of FM decreases with the increase in the frequencies.

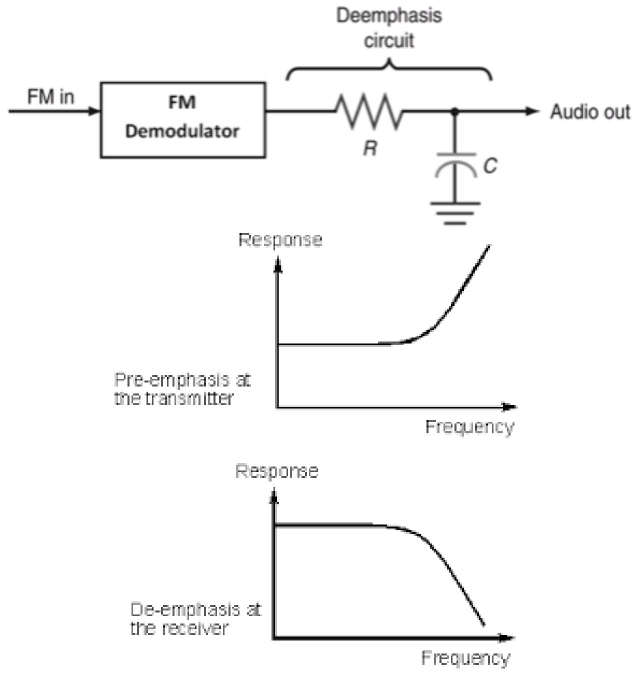
Pre-emphasis and De-emphasis Techniques refer to the operations done on the signal before transmission and after transmission. These help to avoid distortions and also do not cause any major changes to the original signal.

- **Pre-Emphasis** : Increasing the relative strength or amplitude of the high frequency components of the message signal before modulation is termed as Pre-emphasis.
- **De-Emphasis** : The backward transformation which restores the original signal after demodulation is termed as De-emphasis.



Pre emphasis and De emphasis can be easily implemented using a simple RC circuit.





This idea of increasing the weightage of higher components while transmitting can be extended to increase the weightage for lower frequency components as well. The range can be tweaked and a 'passband' pre-emphasis filter can also be applied.

3.2. Anti Jamming

A basic way in which a signal/network can be jammed by transmitting a more stronger signal at the same frequency as that of the message signal.

A rudimentary way to solve this could be to transmit not at one particular frequency but at multiple frequencies so that, even if one channel is jammed, the message can still be properly decoded. But bandwidth is an expensive commodity. The whole purpose of cheap FM Broadcast is defeated.

Building on the above idea, instead of multiple bandwidth channels, one could use the technique of frequency hopping. Frequency hopping involves transmitting a signal in parts across a range of frequencies centered around F_c. The sequence of the frequencies at which message is being transmitted is pre-shared between the receiver and transmitter.

Frequency hopping helps overcome jamming of a particular channel. But advanced jamming techniques like follow-on jamming and channel-hop jamming jam not a single channel but multiple at small intervals of time. To handle this the pattern of frequency hops at receiver and transmitter should be generated randomly such that the transmitting and receiving frequency are the same. Also the time interval during which signal is being transmitted should be made as small as possible to be on a safer side.

4. Observations

4.1. Noise Reduction

The noise added was AWGN. First the signal was passed through a Pre-Emphasis filter and then modulated. Later some noise was added. Finally, the signal is demodulated to get the required message signal back. This approach adds more weightage to the higher frequency components. This reduces the impact of noise on higher frequency components.

This process was used using different types of Pre-Emphasis and De-Emphasis filters. The filters and their respective results have been described in the respective subsections.

4.1.1. Linear Filter

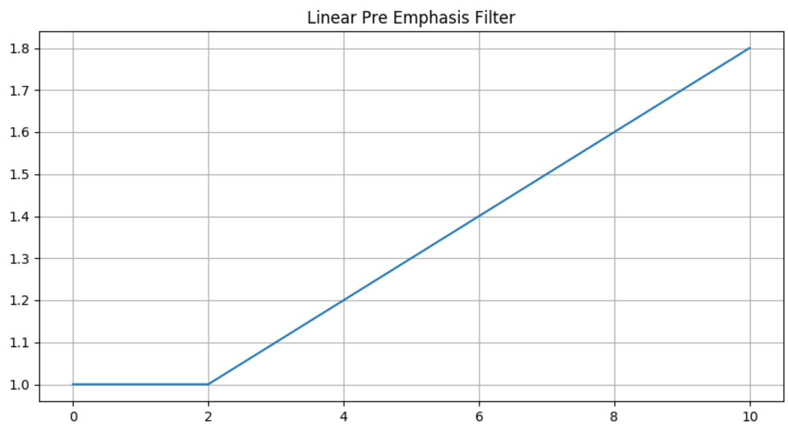
Linear Filters for pre-emphasising and de-emphasinzing have been chosen.
For pre-emphasis the transfer function is of the form

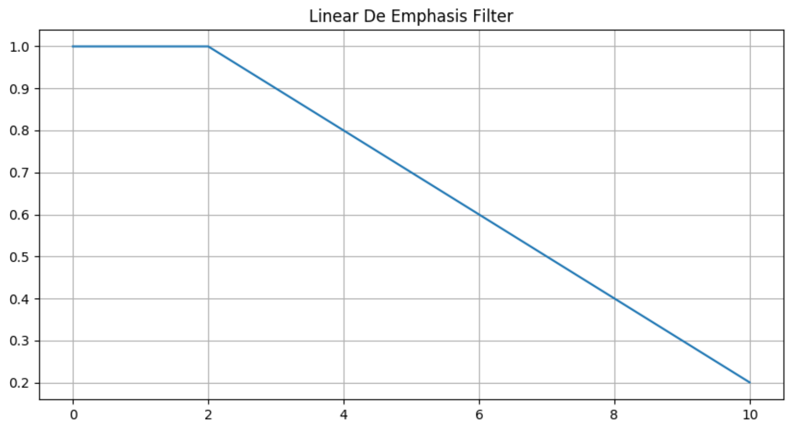
$$H_{pre}(f) = 1 + \alpha(f - f_0)$$

where α is the slope or the scaling factor and f_0 is the starting frequency.
Transfer function of a de-emphasis filter is the reciprocal of the pre-emphasis filter.

$$H_{de}(f) = \frac{1}{H_{pre}(f)}$$

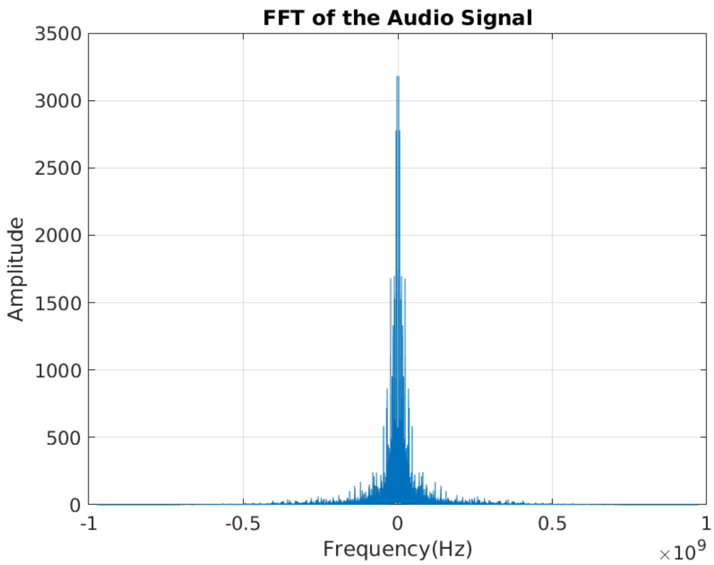
The frequency responses for the filters are given as follows :

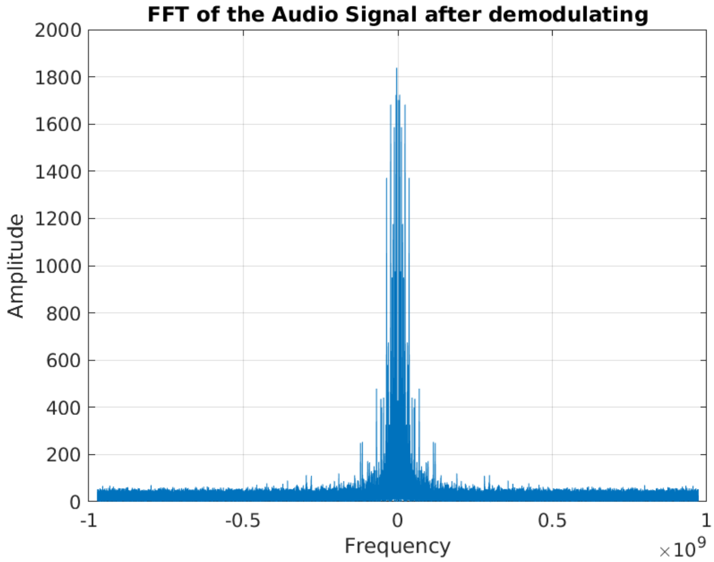




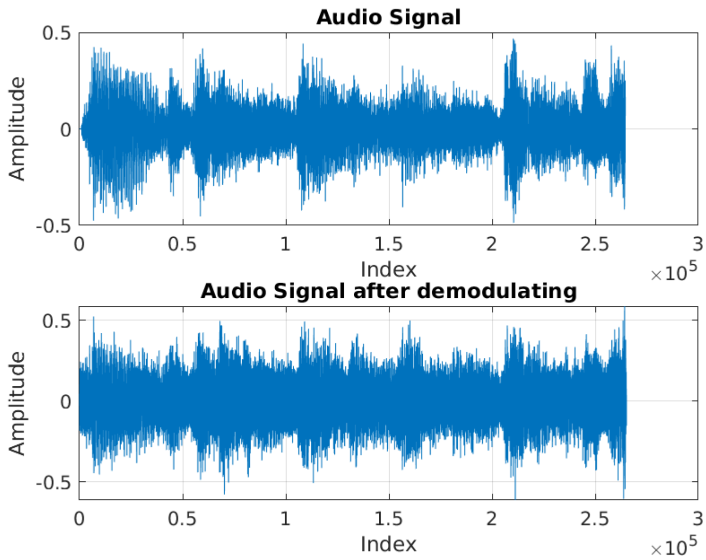
The audio signal has most of its information in the lower frequency components only. Therefore increasing the weightage for higher frequency components is redundant. The AWGN noise is present throughout the frequency spectrum. Hence it is affecting the lower frequency component of the audio signal which has the same weightage as the one without the filter. Therefore the output obtained has a lot of noise in it's lower frequency components as well. The higher frequency components can be removed using a low pass filter, but there is always some data loss in that case. Most of the higher frequency components like sound of plucking a guitar string is lost. Therefore using a linear filter for pre-emphasis and de-emphasis did not contribute to remove any noise from the received signal.

The FFT plots of the actual signal and the demodulated signal are as shown below :





The Final audio comparison is as follows :



It can be observed that there is still a lot of noise when compared to the original signal.

4.1.2. Exponential Filter

Although there was only little or no information in the higher frequency components, the Linear filters gave high importance to the higher frequency components. Therefore there was almost no change while trying to remove the noise.

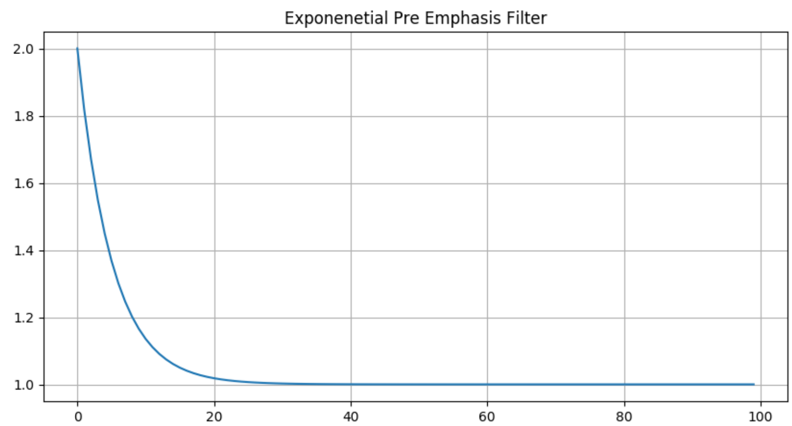
Now the aim was to add more importance to the lower frequency components. To do that an exponential filter was considered. For pre-emphasis the transfer function is of the form

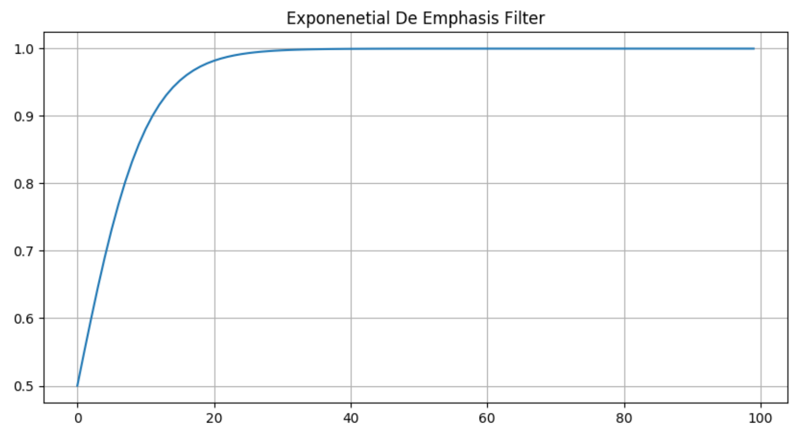
$$H_{pre}(f) = e^{-\alpha f} + 1$$

where α is the scaling factor.
Transfer function of a de-emphasis filter is the reciprocal of the pre-emphasis filter.

$$H_{de}(f) = \frac{1}{H_{pre}(f)}$$

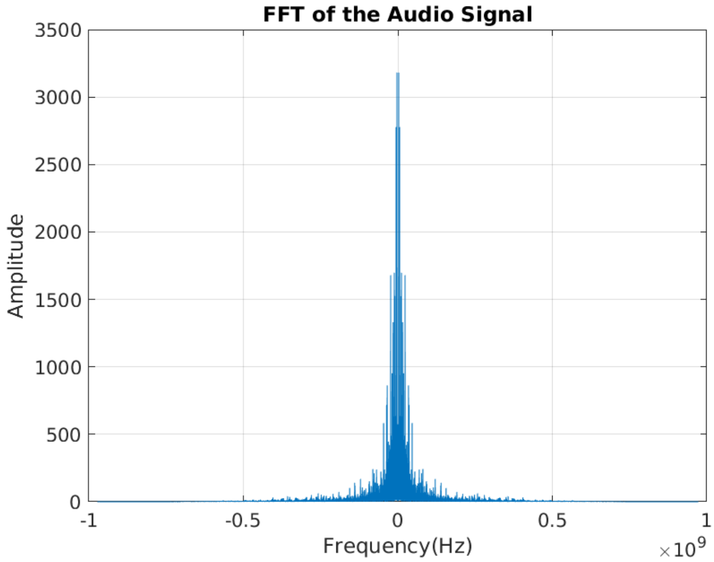
The frequency responses for the filters are given as follows :

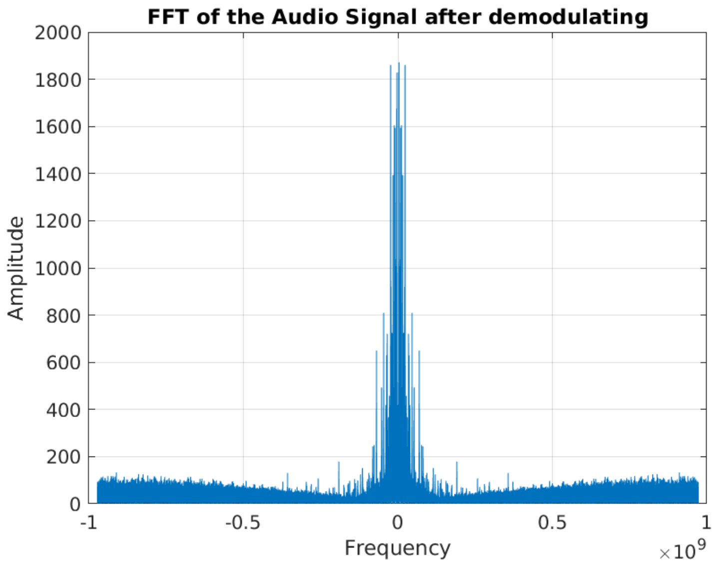




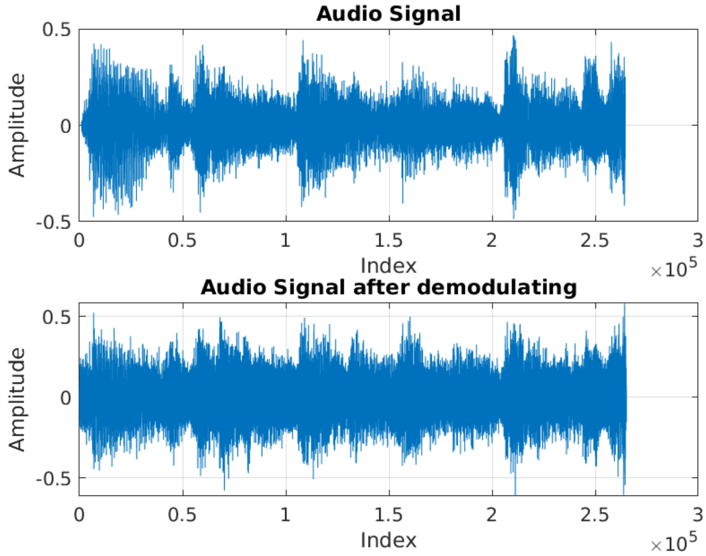
This method gave lower frequency component more importance. But there is still some noise in the higher frequency components. To remove that a low pass filter can be used. But then the same problem arises. The data in higher frequency components are lost.

The FFT plots of the actual signal and the demodulated signal are as shown below :





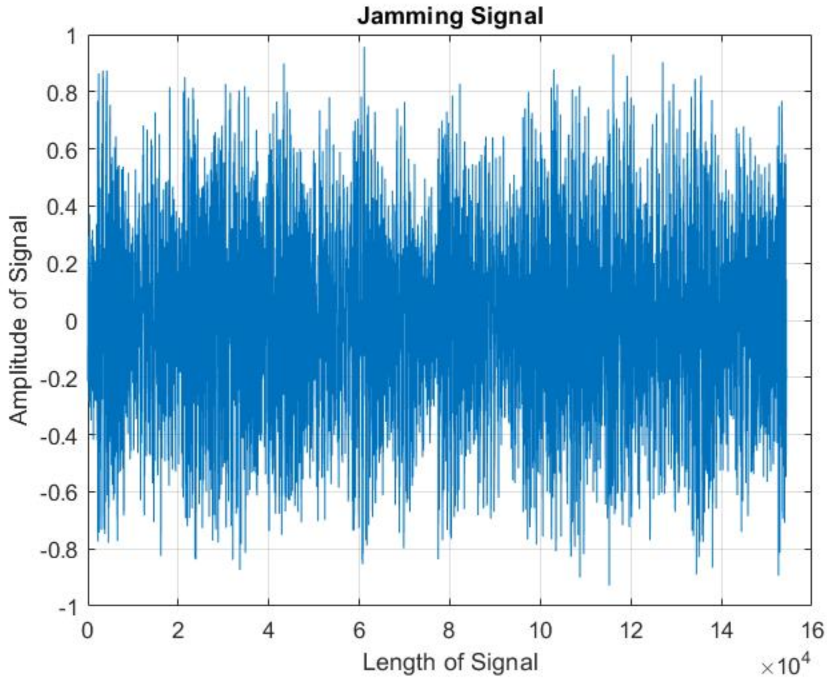
The Final audio comparison is as follows :



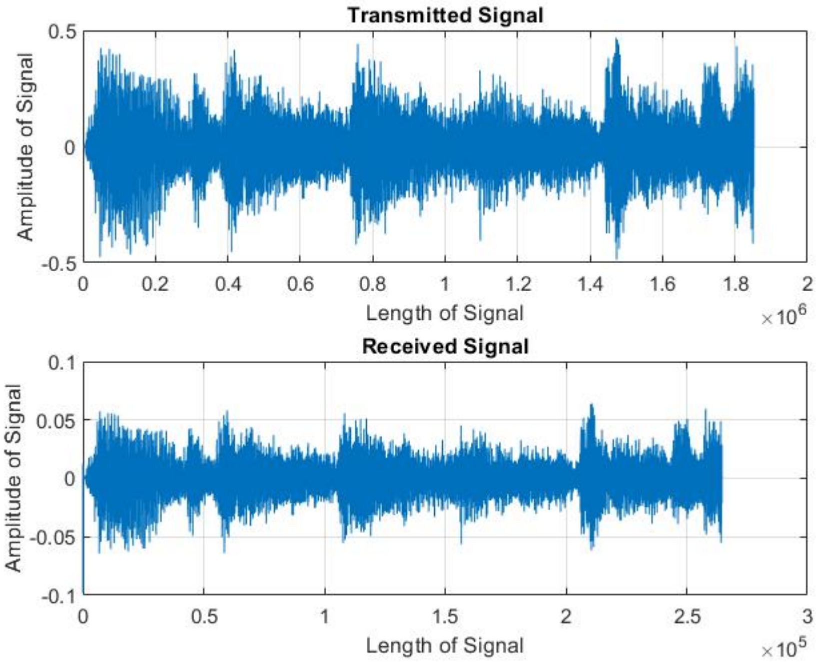
It can be observed that there is still a lot of noise when compared to the original signal. Most of the noise comes from the higher frequency component itself.

4.2. Anti Jamming

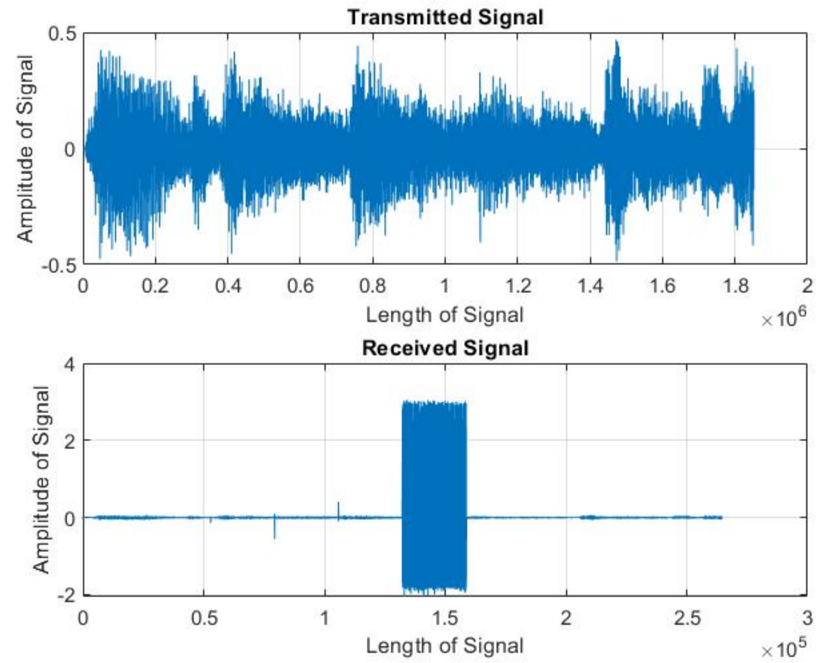
The signal being used to jam the message signal is:



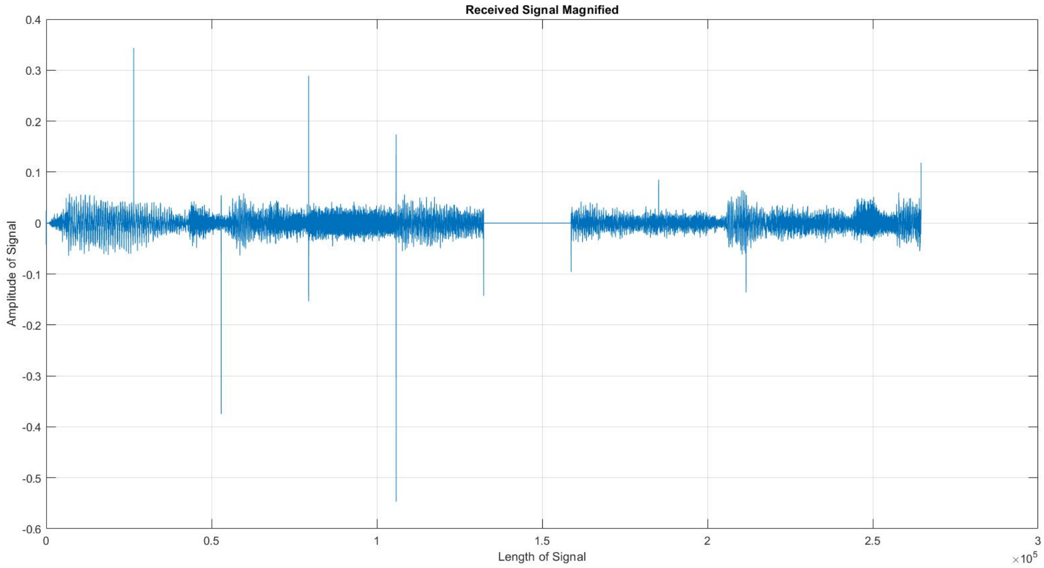
The transmitted and received signal over a channel with single frequency jammer is:



The transmitted and received signal over a channel with multi frequency jammer is:



The signal after removing the noise is:



Capture effect can be seen in the above image. When a receiver receives more than one signal at the same frequency it demodulates the one which has higher power. The output signal has been passed through a low pass filter to remove spikes which occur at the place of joining of reconstructed parts. This leads to slight loss of data. This loss can be avoided by using windows of overlapping messages rather than just sending independent message parts. This helps in reconstructing the signal properly with minimal loss.

5. CONCLUSION

Although using an exponential filter gave a better result, there is still some data loss in higher frequency components. Therefore pre-emphasis and de-emphasis may not be a preferred way to transmit signals that have data in higher frequency components as well. Phase Locked Loop(PLL) at the demodulator seem

s to be a better method to avoid noise interference.

Anti jamming with frequency channel hop is an effective and a cost effective method. In case of FM where major portion of the data being transmitted is audio slight loss is acceptable(Loss occurs when the frequency of the jammer and that of the message are same and message has higher power leading to it getting demodulated because of capture effect.). The amount of loss can be optimized by estimating the parameters of the frequency hop(Like range of frequencies, time interval between change in frequencies) over the channel in which the message is being transmitted.

6. References

- Signal Enhancement by Science Direct
- Pre-Emphasis and De-Emphasis of signals by DAEnotes
- FM Pre-Emphasis and De-Emphasis from RadioMuseum
- Jamming and Anti jamming Techniques
- Communication Systems by Dr. Cong Ling from Imperial College, London
- Short Notes on Pre-Emphasis and De-Emphasis by Ques10

A. Appendix A – Noise Reduction

A.1. Modulator

```
function [theta, time] = modulator(os_factor, kf, nsymbols,
   pulse)
    os_factor = ceil(max(kf, 1)*os_factor);
    ts = 1/os_factor;
    nsamples = ceil(1/ts);
    symbols = sign(rand(nsymbols,1)-0.5);
    nsymbols_upsampled=1+(nsymbols-1)*nsamples;
    symbols_upsampled=zeros(nsymbols_upsampled,1);
    symbols_upsampled(1:nsamples:nsymbols_upsampled)=symbols;
    message = conv(symbols_upsampled,pulse);
    theta = 2*pi*kf*ts*cumsum(message);
    time=transpose(0:length(theta)-1)*ts;
end
```

A.2. Demodulator

```
function message_estimate = demodulate(signal)
    I_comp = real(signal);
    Q_comp= imag(signal);
    Iderivative = [0; diff(I_comp)];
    Qderivative = [0; diff(Q_comp)];
    message_estimate = (I_comp.*Qderivative - Q_comp.*
       Iderivative);
    message_estimate = message_estimate./(I_comp.*I_comp +
       Q_comp.*Q_comp);
end
```

A.3. Main Code

```
[y, Fs] = audioread('sample.wav');
t = linspace(0, length(y)/Fs, length(y));
% H = [1 -0.95];
df = Fs/length(y);
x = -Fs/2 : df : Fs/2 - df;
x = Fs*x;
H = ifft(exp(-0.001*abs(x)) + 1);
y_pre = filter(H, 1, y);

figure(1000);
plot(abs(fft(H)));
```

```
Fc = 100 * 10^6;
% signal_sent = fmmod(y_pre, Fc, 2*Fc, 75*10^3);
[theta_y, mod_time] = modulator(16, 4, 10, y_pre);
signal_sent = exp(j*theta_y);
noisy = awgn(signal_sent, 25);
r = demodulate(noisy);
% H_de = H.^1;
H_de = ifft(1./fft(H));

deemp = filter(H_de, 1, r);

for i = 0:50
    df = fft(deemp);
    df(df == max(df)) = 0;
    deemp = ifft(df);
end

% deemp = lowpass(deemp, 0.25);
deemp = 1*deemp;

audiowrite('rec.wav', r, Fs);
audiowrite('deemp.wav', real(deemp), Fs);

figure(100);
subplot(4, 1, 1);
plot(fftshift(abs(fft(y))));
subplot(4, 1, 2);
plot(fftshift(abs(fft(y_pre))));
subplot(4, 1, 3);
plot(fftshift(abs(fft(r))));
subplot(4, 1, 4);
plot(fftshift(abs(fft(deemp))));

figure(1);
plot(Fs*(-Fs/2:Fs/length(y):Fs/2 - Fs/length(y)), fftshift(abs(
    fft(y))));
xlabel('Frequency(Hz)'), ylabel('Amplitude'), grid on
title('FFT of the Audio Signal')

figure(2);
plot(Fs*(-Fs/2:Fs/length(deemp):Fs/2 - Fs/length(deemp)),
    fftshift(abs(fft(deemp))));
xlabel('Frequency'), ylabel('Amplitude'), grid on
title('FFT of the Audio Signal after demodulating')
```

```
figure(200);
subplot(4, 1, 1);
plot(y);
subplot(4, 1, 2);
plot(real(y_pre));
subplot(4, 1, 3);
plot(r);
subplot(4, 1, 4);
plot(real(deemp));

figure(3);
plot(t, y);
xlabel('Time(s)'), ylabel('Amplitude'), grid on
title('Audio Signal');

figure(4);
plot(mod_time, real(deemp));
xlabel('Time(s)'), ylabel('Amplitude'), grid on
title('Audio Signal after demodulating');
```

B. Appendix B – Anti Jamming

```
s = rng;            %Random number generation seed

%Audio and noise input files
[y, Fsa] = audioread('sample_1.wav');
[cap,Fsa_c] = audioread('sample.wav');


%Upsampling the audio file to Fs frequency
y = upsample(y,7);

figure()
plot (cap)
title('Jamming Signal')
xlabel('Length of Signal')
ylabel('Amplitude of Signal')
grid on


%Upsampling the noise to Fs frequency
cap = upsample(cap,15);
cap_dist = cap(1:length(y)/10);
a=randperm(10,10)+(-10-1);


%dividing the signal into parts to transmit at different
   frequencies
y_dist = zeros(length(y)/10,1);
demod_signal_dist = zeros(length(y)/10,1);

for i = 0:9
     send = y(((i)*length(y)/10 + 1) :((i+1)*length(y)/10)) ;
     y_dist( :, i+1 ) = send ;
end


%Transmission and reception of the message signal
for j = 1:10

    Fc = 1*10^5 + 2000*a(j) ;
    Fs = 3*Fc;
```

```
    Fcn = 1.5*10^5 ;
    Fsn = 2*Fcn;

    freqdev = 800;
    y_dist(:,j) = lowpass(y_dist(:,j),800,Fs);
    cap_dist = lowpass(cap_dist,800,Fs);
    mod_signal = fmmod(y_dist(:,j),Fc,Fs,freqdev,0);
    %For frequency hopping of the jammer replace Fcn by
       10^5+2000*b(j)
    mod_signal = mod_signal + fmmod(cap_dist,Fcn,Fsn,freqdev,0)
       ;

    mod_signal = bandpass(mod_signal,[Fc-400,Fc+400],Fs);
    x1 = fmdemod(mod_signal,Fc,Fs,freqdev,0);
    demod_signal_dist(:,j)= x1;

end


 demod_signal = demod_signal_dist(:);


%Removing the noise spikes
 for l = 1:10
   demod_signal(length(y)/10*l-50:length(y)/10*l+50) = 0 ;
end

%Downsampling to audio frequency
demod_signal = downsample(demod_signal,7);
demod_signal = lowpass(demod_signal, 800, Fsa);
sound(demod_signal*10,Fsa)



figure()
grid on


subplot(2, 1, 1)
plot(y)
title('Transmitted Signal')
xlabel('Length of Signal')
ylabel('Amplitude of Signal')
grid on
```

```
subplot(2, 1, 2)
plot(demod_signal)
title('Received Signal')
xlabel('Length of Signal')
ylabel('Amplitude of Signal')
grid on
```