DIGITAL SIGNAL PROCESSING LAB

Debagnik Kar
Roll: 1804373, ETC -06, EC - 3096

Index Page

Experiment Number	06
Date of Experiment	17/02/2021
Date of Submission	24/02/2021
Name of the Student	Debagnik Kar
Roll Number	1804373
Section	ETC - 06

Aim of The Experiment :-

To design FIR, LPF, HPF & BPF using windowing techniques

Software Required :-

- MATLAB R2018a

Theory :-

a) FIR:
- Finite Impulse Response
- It is a filter whose impulse response is of finite period, as a result of it settles to zero in finite time
- Used to implement almost any sort of frequency digitally.

b) LPF:
- Low pass filter
- It passes signals with a frequency lower than a selected cutoff frequency and attenuates signals with a frequency higher than cutoff frequency
- Filters noise from a circuit.

c) HPF:
- High pass filter
- It allows the signals with frequency higher than a certain cutoff frequency and attenuates signals with a lower than the cut off frequency.

d) BPF:
- Band pass filter
- A device that passes frequency within a certain range and alternative frequency outside that range.

Cut-off frequencies of the filters ($\times \pi \, rad/samples$)

	Rectangular Window	Triangular Window
Low Pass	0.3	0.3
High Pass	0.6	0.6
Band Pass	0.2, 0.5	0.2, 0.5

Code :-

```
%Main code for designing FIR Filter
%written by Debagnik Kar 1804373
clear all
close all
clc
prompt0 = input('Select window [Rect(1)/tri(2)]:  ')
prompt1 = input('select filter type [LP(1)/HP(2)/BP(3)]:  ')
N = 50
switch prompt0
    case 1
        win = rectwin(N+1)
    case 2
        win = triang(N+1)
    otherwise
        win = []
        disp('error')
end
switch prompt1
    case 1
        fil = "low"
        cut = 0.3
    case 2
        fil = "high"
        cut = 0.5
    case 3
        fil = "bpf"
        cut = [0.3 0.5]
    otherwise
        disp('error 2')
end
fs = 1000
N1 = 300
n = 0:N1-1
%generation and adding noise
xn = sin(20*pi*n/fs)
figure(1)
subplot 211
plot(n,xn)
title("original sine")
rn = xn+rand(size(xn))
subplot 212
plot(n,rn)
title('Noisy sine')
%Filtering
fxn = fir1(N,cut,fil,win)
yn = filter(fxn,1,rn)
figure(2)
subplot 313
freqz(fxn)
figure(3)
plot(n,yn)
```

Graph/Output :-

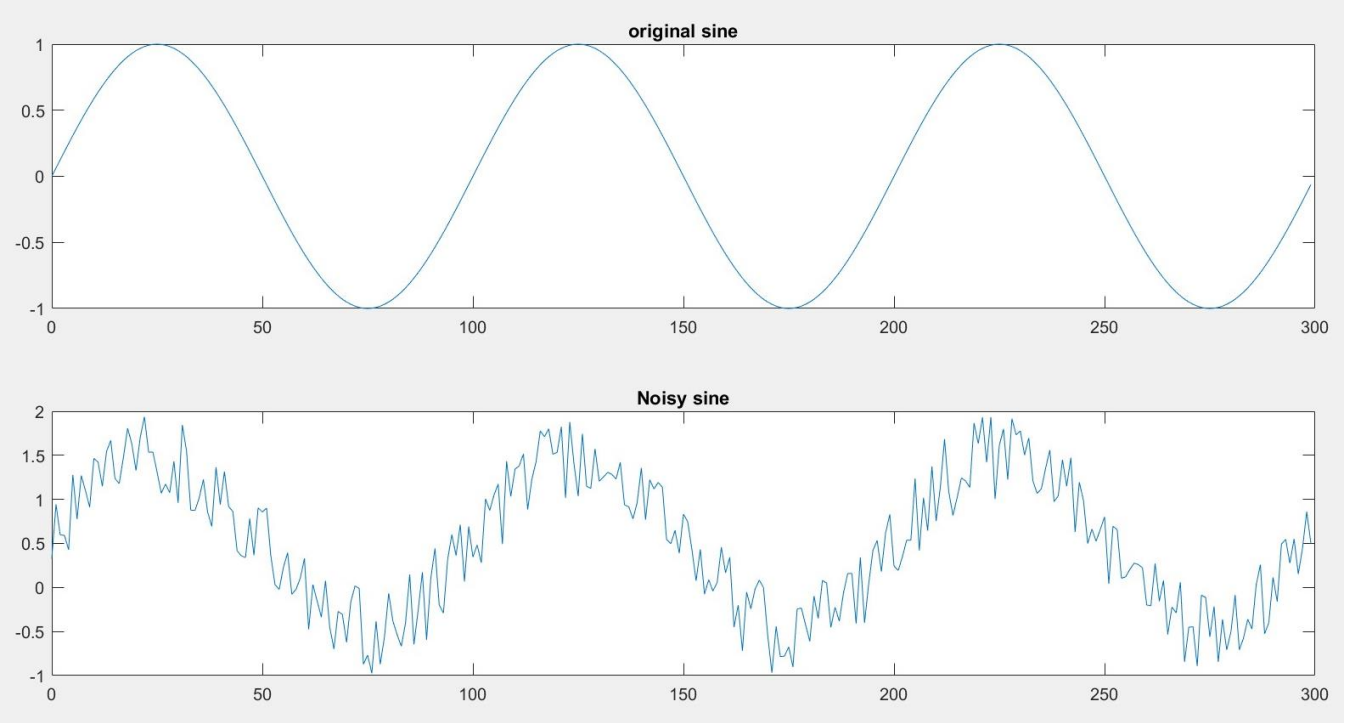

Fig 6.1: Input signals

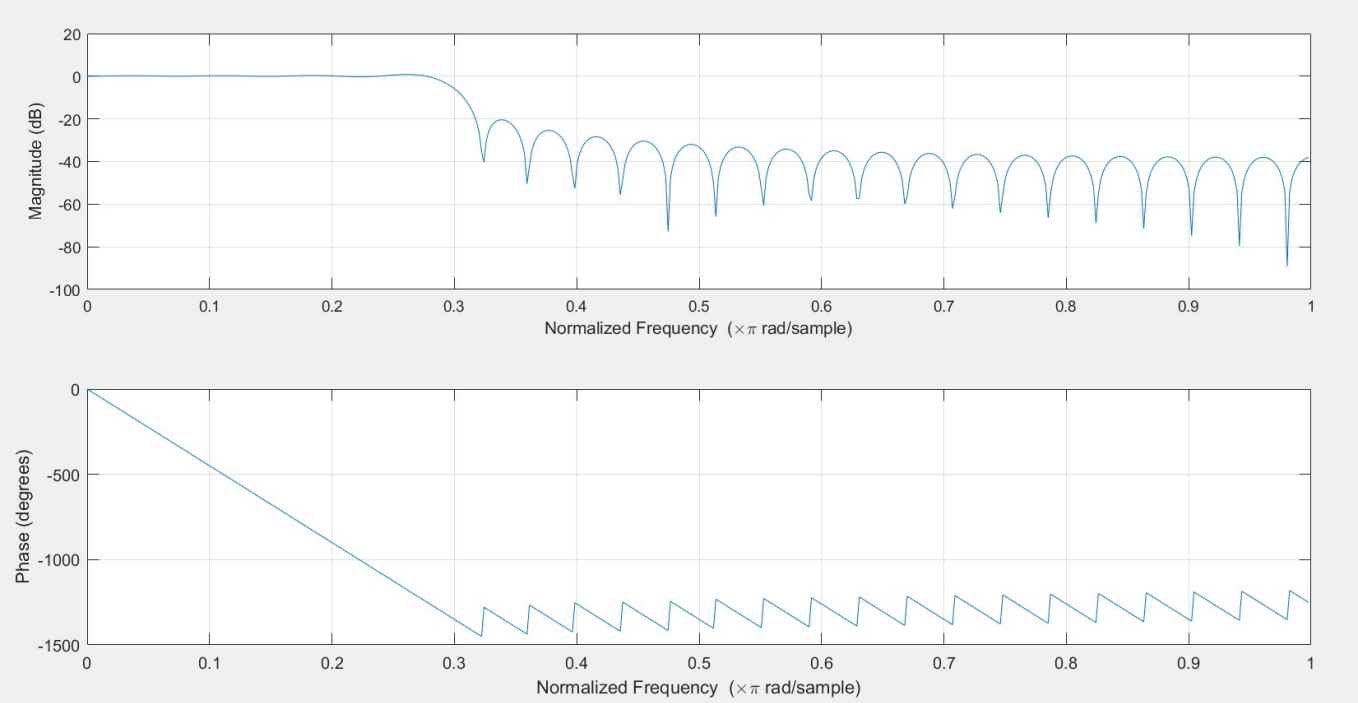

Fig 6.2: frequency response of the low-pass FIR filter with rectangular window

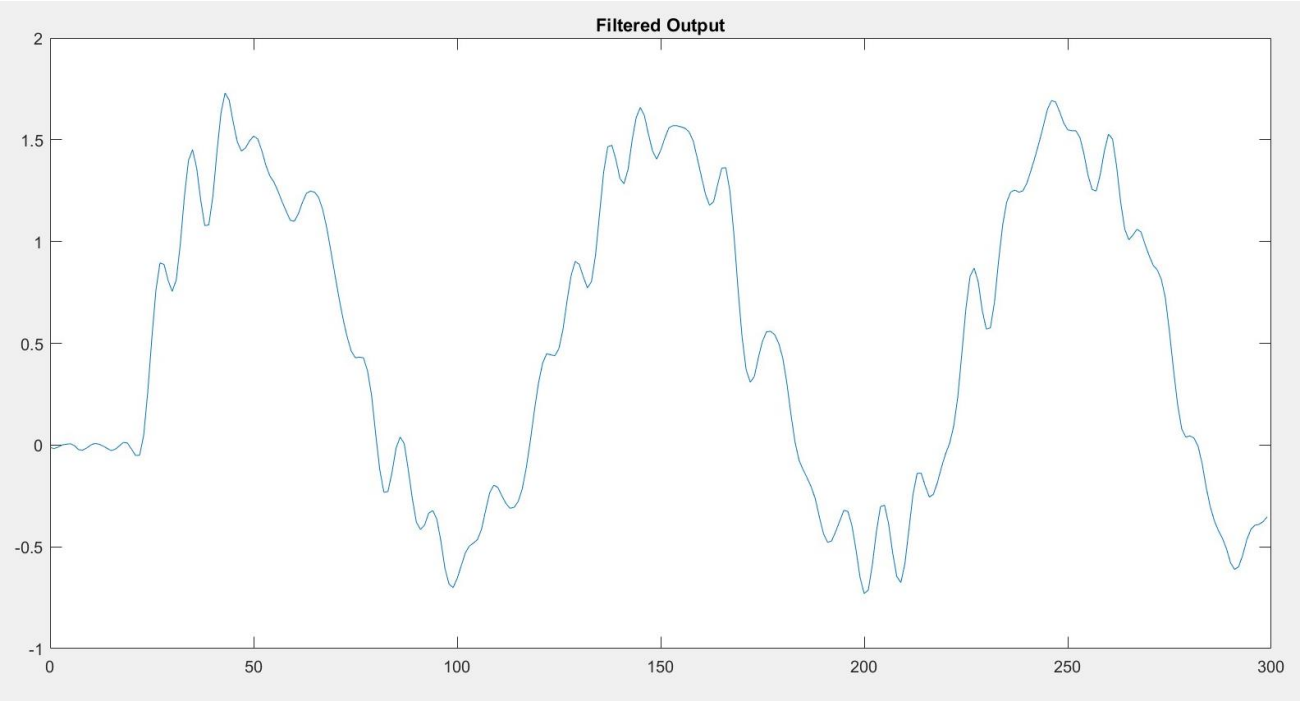

Fig 6.3: filtered output. (Low pass – rectangular window)

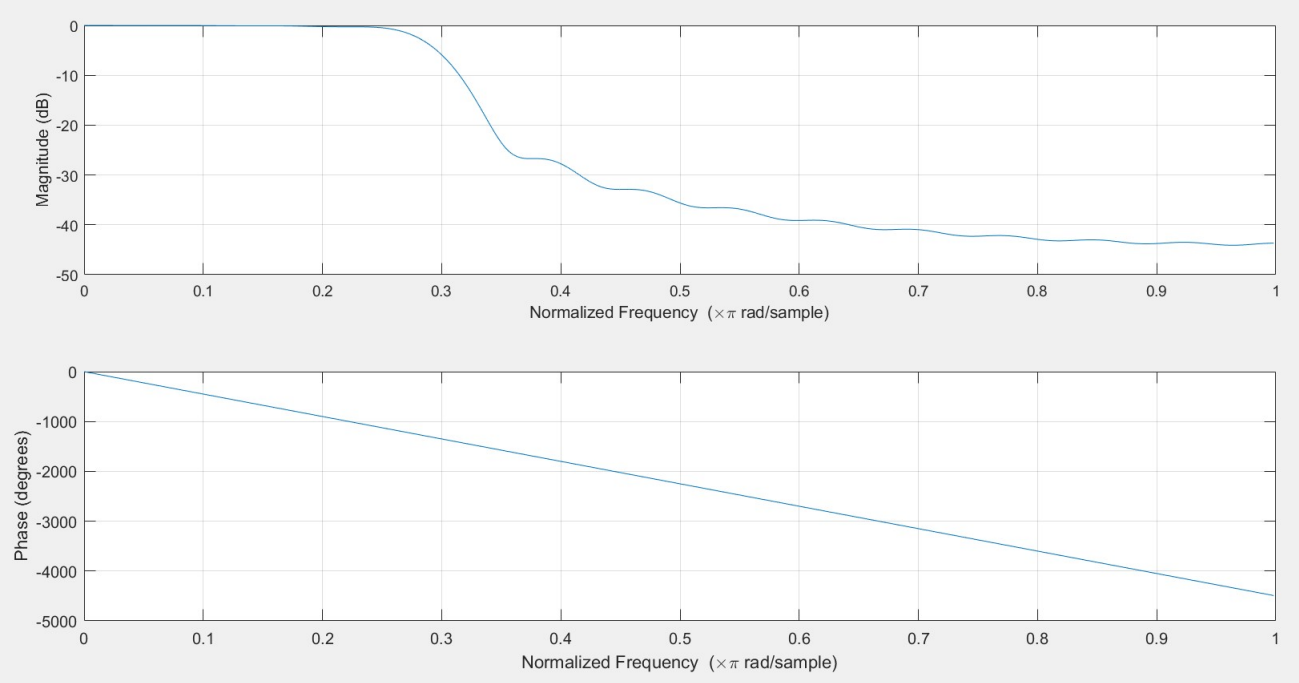

Fig 6.4: frequency response of lowpass FIR filter with triangular window

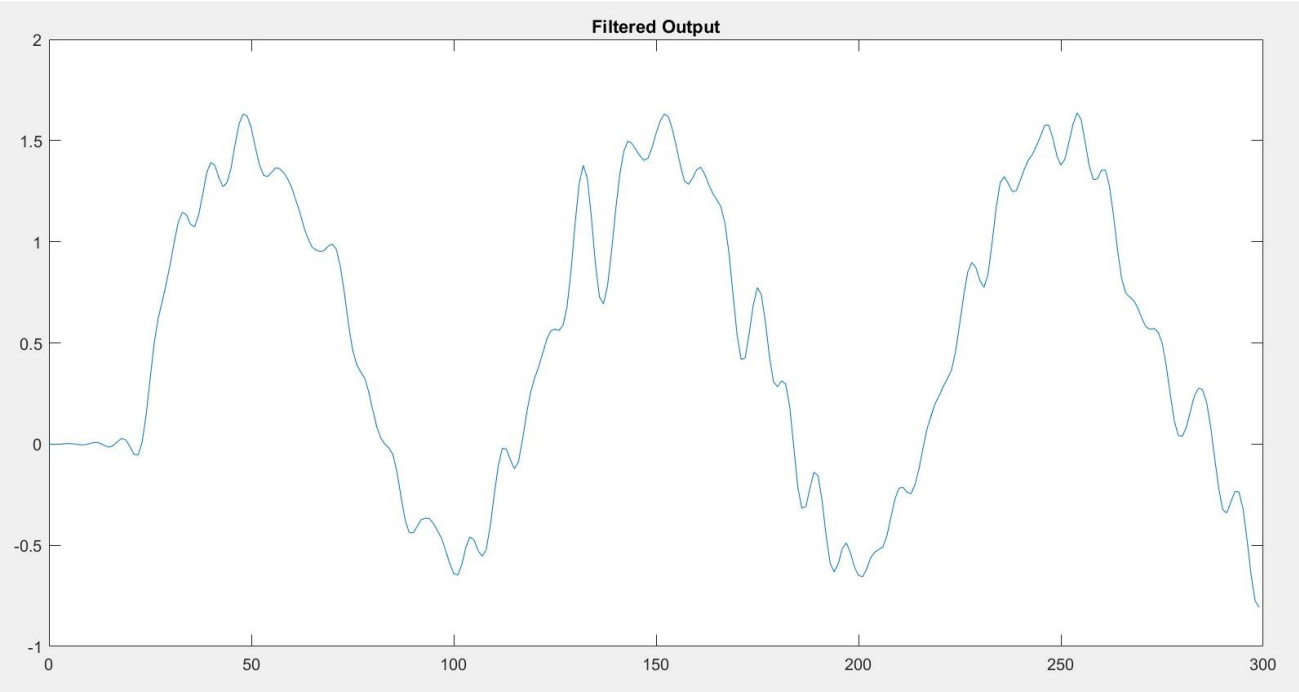

Fig 6.5: filtered output. (Low pass – triangular window)

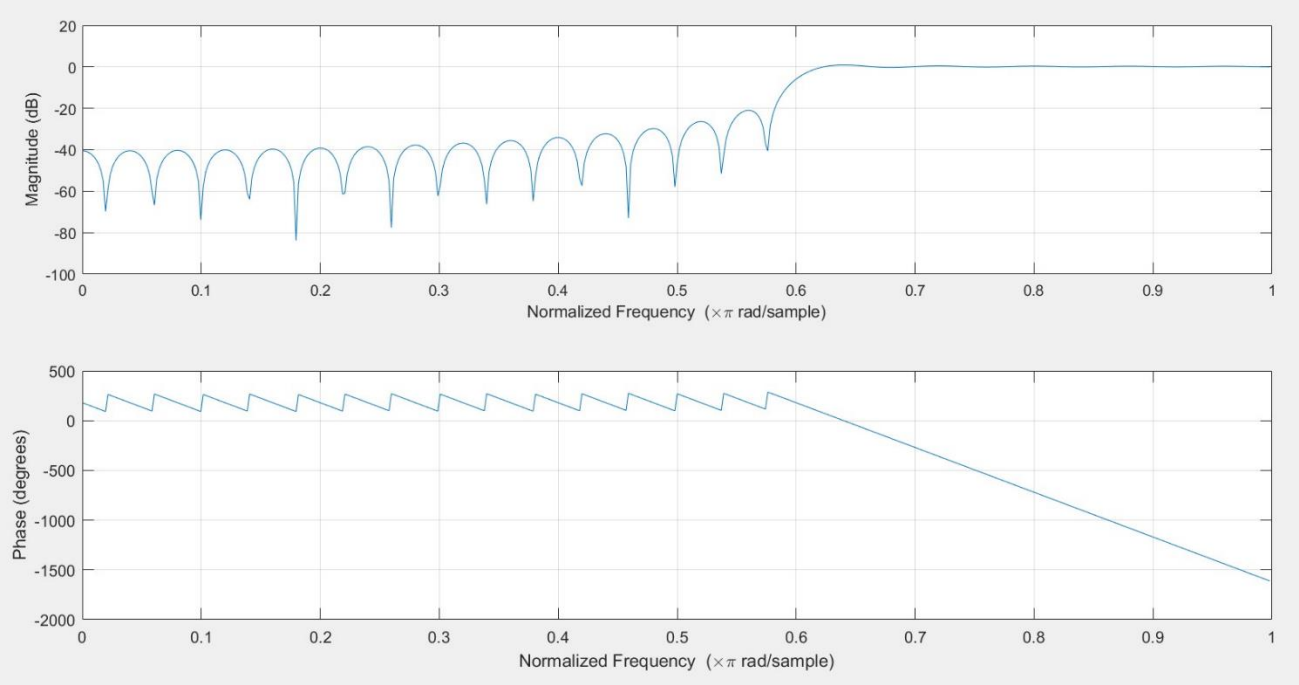

Fig 6.6: frequency response of high pass FIR filter with rectangular window

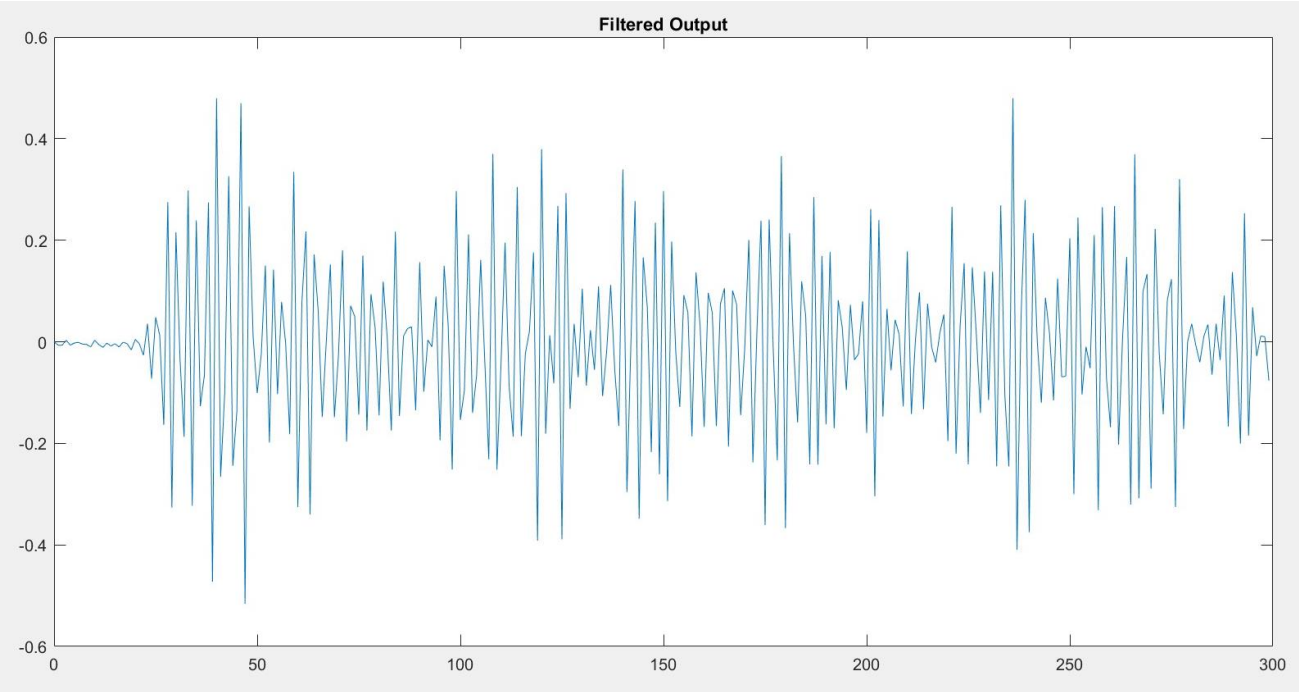

Fig 6.7: filtered output for high pass FIR filter with rectangular window

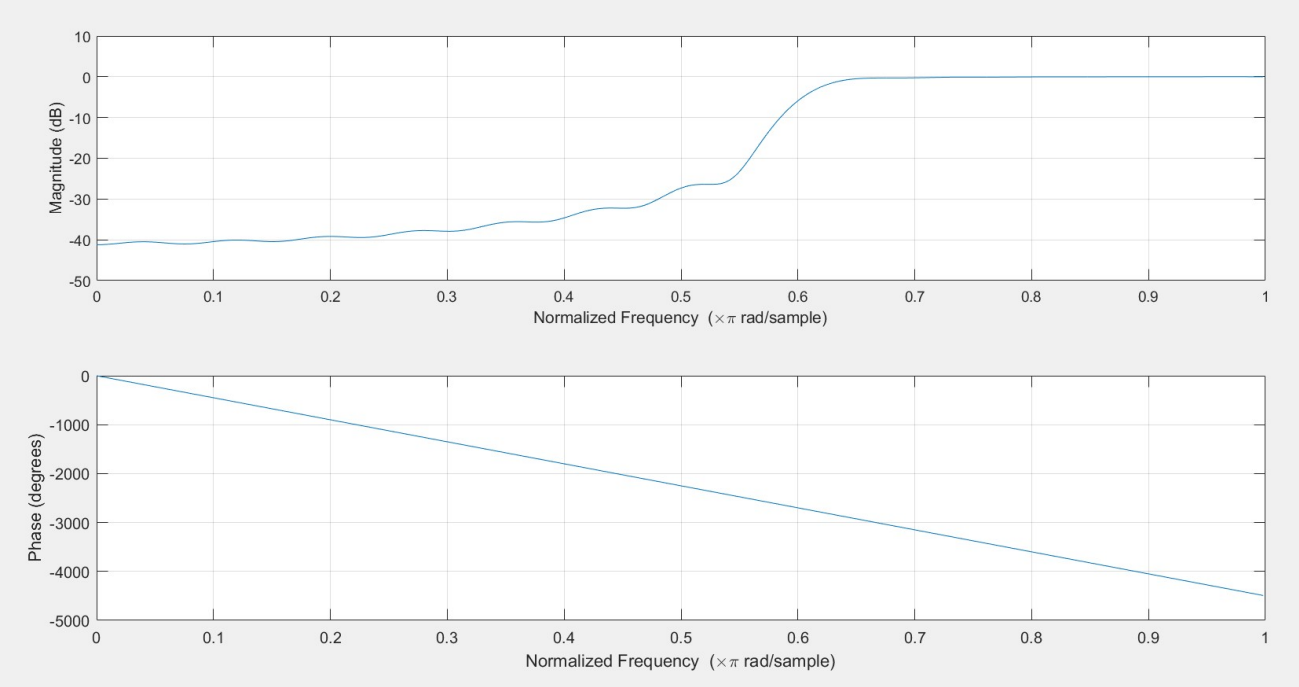

Fig 6.8: frequency response of high pass FIR filter with triangular window

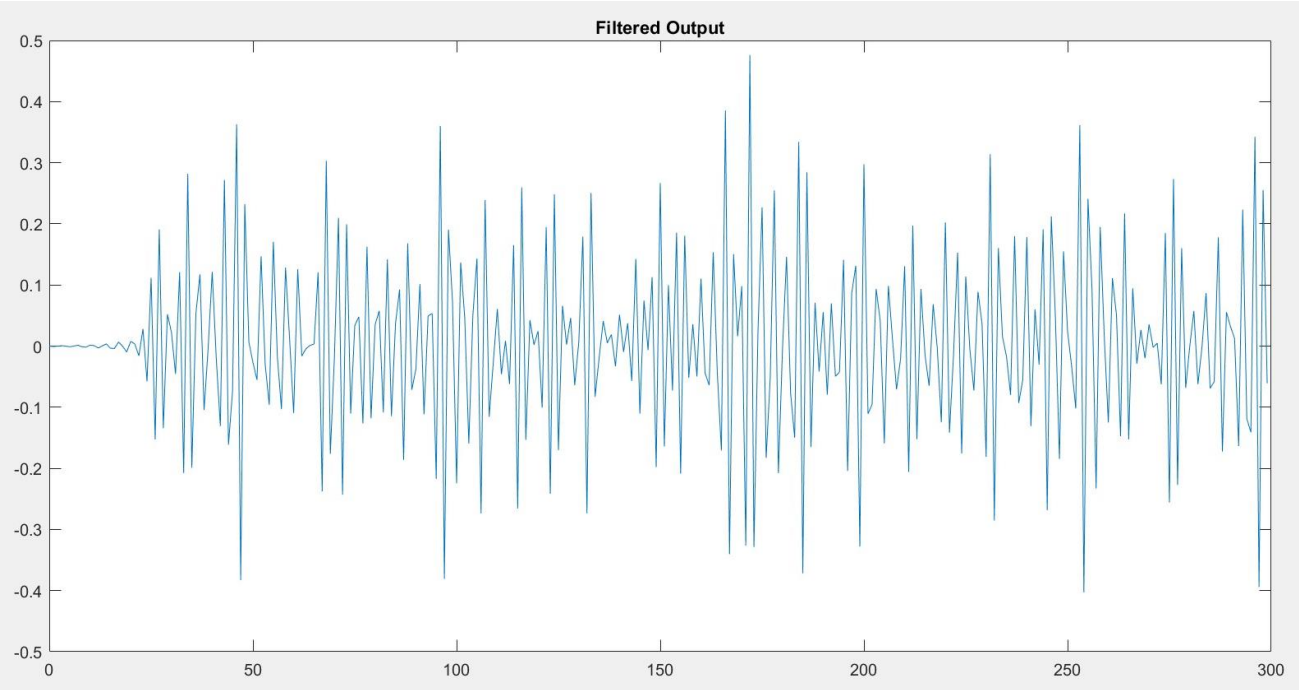

Fig 6.9: filtered output for high pass FIR filter with triangular window

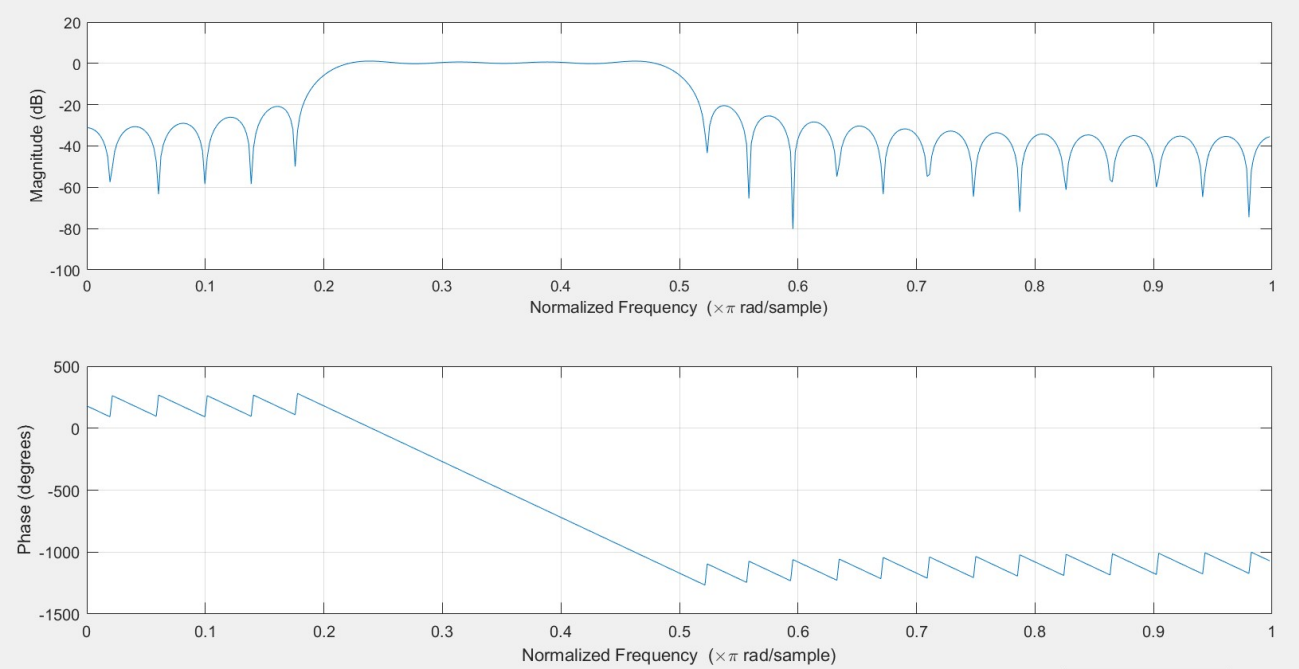

Fig 6.10: frequency response of band pass FIR filter with rectangular window

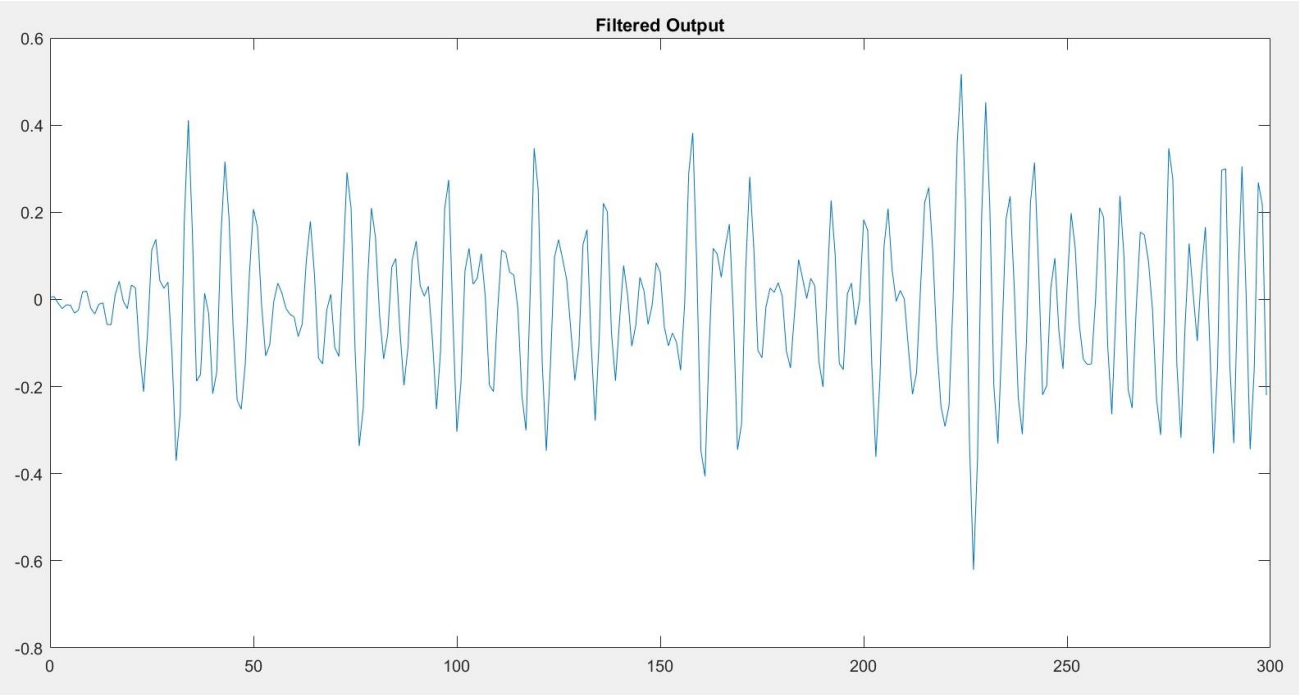

Fig 6.11: filtered output for band pass FIR filter with rectangular window

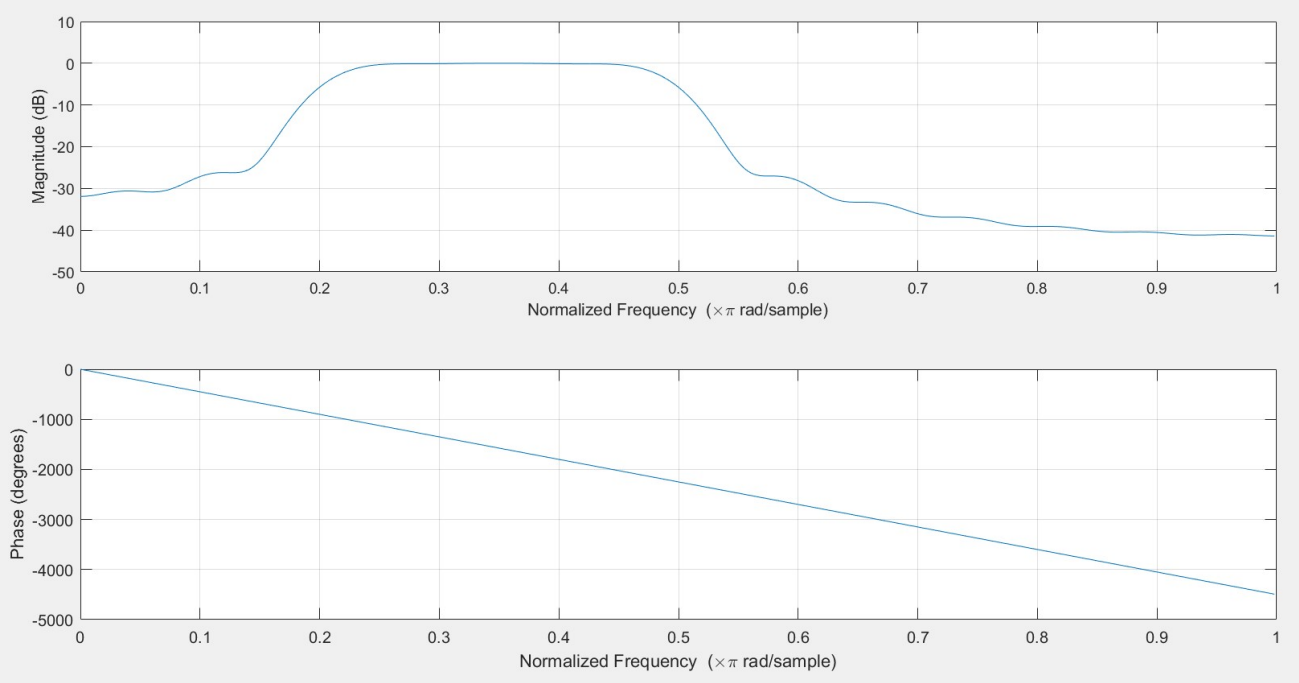

Fig 6.12: frequency response of band pass FIR filter with triangular window

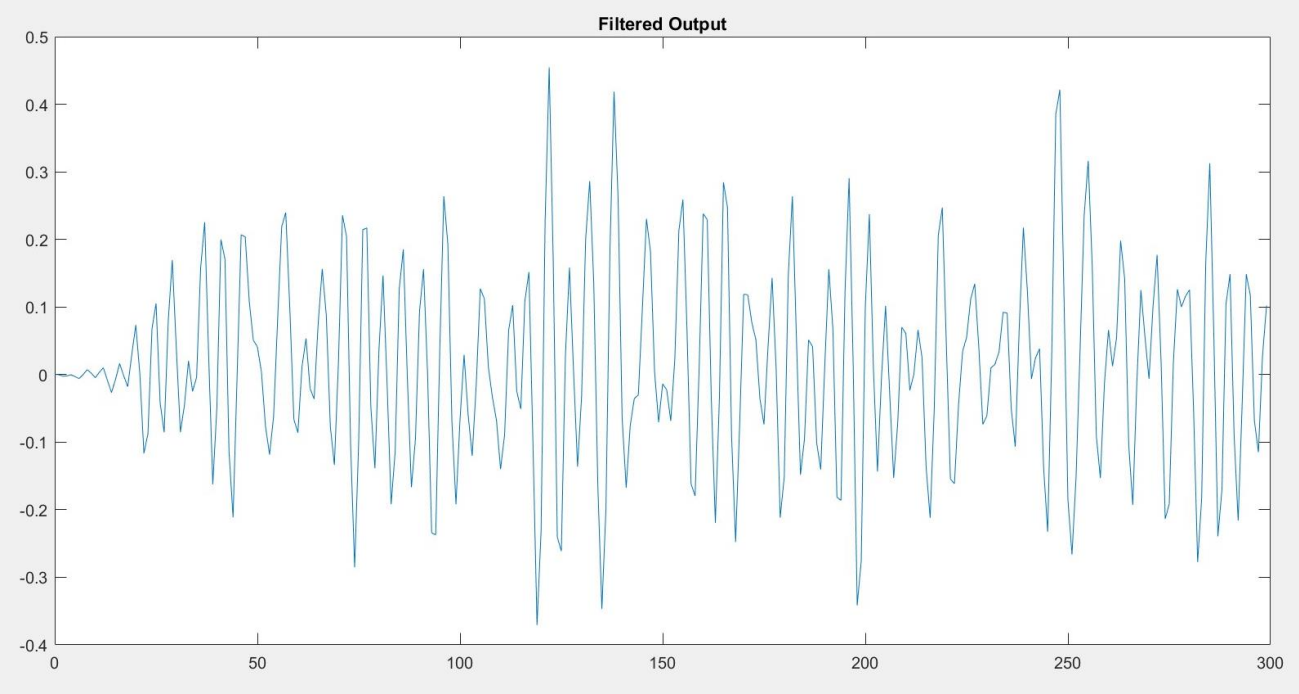

Fig 6.13: filtered output for band pass FIR filter with triangular window

Discussion/Inference of the experiment :-

In this experiment we designed FIR filters using rectangular and triangular window for lowpass, high pass, and bandpass by simulation in MATLAB.

For windows we used rectwin() and triang() functions. We generated a sinusoidal message signal to it. Then created the FIR filter by using the fir1() functions with the appropriate parameters for high, low or bandpass filters and also designed assigned the cut-off frequency accordingly. Then used the filter() function and the frequency responses using freqz().

Conclusion :-

We designed and simulated a lowpass/high pass/bandpass FIR filter successfully using MATLAB using windowing techniques.