Experiment 2

Designing Low Pass Filter by Windowing Method

2.1 AIM:

To design FIR filters for various orders and cut-off Frequencies for five different window functions.

To determine how FIR filters are responding to the input signals contaminated by noise.

2.2 MATLAB FUNCTION USED:

```
Zeros, floor, plot, freqz
```

2.3THEORY:

FIR filters are digital filters with finite impulse response. They are also known as non-recursive digital filters as they do not have the feedback. The window method is most commonly used method for designing FIR filters. The simplicity of design process makes this method very popular. A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.

When designing digital FIR filters using window functions it is necessary to specify:

- A window function to be used; and
- The filter order according to the required specifications (selectivity and stopband attenuation).

These two requirements are interrelated. Each function is a kind of compromise between the two following requirements:

- The higher the selectivity, i.e. the narrower the transition region; and
- The higher suppression of undesirable spectrum, i.e. the higher the stopband attenuation.

2.4 PROCEDURE: First we define a low pass filter in time domain as,

$$W(n) = \frac{sin(w_c(n-k))}{(\pi(n-k))} forn \neq k$$

$$\frac{w_c}{\pi} forn = k$$

Where N= number of samples

k= (N-1)/2

We then a window is chosen out of the following window types, for three different values of N:

Rectangular window	$W(n) = 1; n = 0,1 \dots N-1$ $0; otherwise$
Triangular window	$W(n) = 1 - 2\left(\frac{n-(N-1)/2}{N-1}\right); n = 0,1 \dots N-1$ $0; otherwise$
Hanning window	$W(n) = 0.5 - 0.5cos\left(\frac{2\pi n}{N-1}\right); n = 0,1 \dots N-1$ $0; otherwise$
Hamming window	$W(n) = 0.54 - 0.46cos\left(\frac{2\pi n}{N-1}\right); n = 0,1 \dots N-1$ $0; otherwise$
Blackmann window	$W(n) = 0.42 - 0.5cos\left(\frac{2\pi n}{N-1}\right)$ $+0.08cos\left(\frac{4\pi n}{N-1}\right); n = 0,1..N-1$ $0; otherwise$

- The window and low pass filter are multiplied in time domain. Ie. h(n)=h_d *w(n)
- Filter characteristics are plotted using the MATLAB function *freqz(B, A, w)*, where B and A are respectively the numerator and denominator of the transfer function. Here the transfer function is FIR low pass filter.
- A function realised in time domain such that one frequency lies in pass band and other frequency lies in stop band of the filter and all the performance of the filters was observed.
- In the second part, white noise was added to the signal and SNR was calculated.

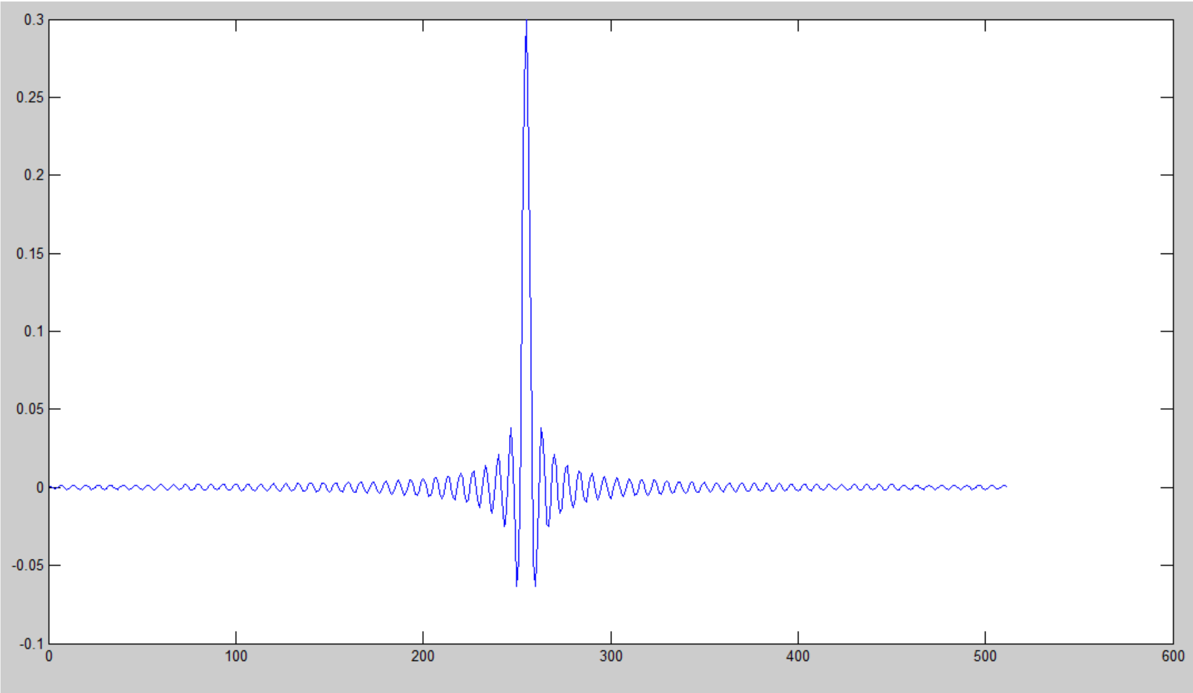
2.5 SOURCE CODE:

```
close all;clear;

fc=.15;
wc=2*pi*fc;
hd=zeros(5,512);H=zeros(5,512);
N=[8 64 512];
for il=1:3
    figure;
    h=zeros(1,512); n=1:1:512;size_n=size(n);
    w=-pi:pi/N(1,il):pi-pi/N(1,il);
    k1 =floor((N(1,il)-1)/2);
    for k=1:N(1,il)-1
        if k==k1
            h(1,k)=wc/pi;
        else
            h(1,k)= sin(wc*(n(1,k)-k1))/(pi*(n(1,k)-
k1));title('Sinc Function');
        end
    end
    plot(h);
    for count=1:5
        if count==1
            for i=1:size_n(1,2)
                if n(1,i)<N(1,il)-1
                    H(count,i)=1;
                else
                    H(count,i)=0;
                end
            end
            figure;
            hd(count,:)= h.*H(count,:);
            freqz(hd(count,:),1,w);title('Rectangular Window');
        end
        if count==2
            for i=1:size_n(1,2)
                if n(1,i)<N(1,il)-1
                    H(count,i)=1-2*(n(1,i)-(N-1)/2)/(N-1);
                else
                    H(count,i)=0;
                end
            end
            figure;
            hd(count,:)= h.*H(count,:);
            freqz(hd(count,:),1,w);title('Triangular Window');
        end
        if count==3
            for i=1:size_n(1,2)
                if n(1,i)<N(1,il)-1
                    H(count,i)=0.5-0.5*cos(2*pi*n(1,i)/(N(1,il)-
1));
                else
                    H(count,i)=0;
                end
```

```
            end
            figure;
            hd(count,:)= h.*H(count,:);
            freqz(hd(count,:),1,w);title('Hanning Window');
        end
        if count==4
            for i=1:size_n(1,2)
                if n(1,i)<N(1,il)-1
                    H(count,i)=0.54-.46*cos(2*pi*n(1,i)/(N(1,il)-
1));
                else
                    H(count,i)=0;
                end
            end
            figure;
            hd(count,:)= h.*H(count,:);
            freqz(hd(count,:),1,w);title('Hamming Window');
        end
        if count==5
            for i=1:size_n(1,2)
                if n(1,i)<N(1,il)-1
                    H(count,i)=0.42-0.5*cos(2*pi*n(1,i)/(N(1,il)-
1))+0.08*cos(4*pi*n(1,i)/(N(1,il)-1));
                else
                    H(count,i)=0;
                end
            end
            figure;
            hd(count,:)= h.*H(count,:);
            freqz(hd(count,:),1,w);title('Blackman Window');
        end
    end
end
tm= 1:512;
sig= sin(0.05*2*pi*tm)+ sin(0.2*2*pi*tm);
sig1= sin(0.05*2*pi*tm);
fsig1= fft(sig1);
fsig= fft(sig);
figure;
plot(-pi:pi/256:pi*255/256,fftshift(abs(fsig)));
title('Frequency spectrum of Signal');
xlabel('Frequency ( in radian )');
ylabel('AMPLITUDE');
ar= zeros(1,512);
ns= 0.5-rand(1,512);
nwsig = sig+ns;
nwsig1 = sig1+(0.5-rand(1,512));
fnsig1= fft(nwsig1);
fnwsig= fft(nwsig);
rs=ar;
nrs=ar;
inrs=nrs;
inp_snr= (rms(sig1)/rms(ns))^2;
op_snr=zeros(1,5);

irs=rs;
```

```
for i=1:5
    ar=fft(hd(i,:));
    rs= ar.*fsig;
    nrs1= fnsig1.*ar;
    nrs= fnwsig.*ar;
    inrs=ifft(nrs1);
    resl=fsig1-nrs1;
    op_snr(1,i)= (rms(fsig1)/rms(resl))^2;
    figure;
    subplot(2,1,1);
    plot(-pi:pi/256:pi*255/256,fftshift(abs(rs)));
    if(i==1) title(' Frequency spectrum of Output of Rectangular
window');
    end
    if(i==2) title('Frequency spectrum of Output of Triangular
window');
    end
    if(i==3) title('Frequency spectrum of Output of Hanning
window');
    end
    if(i==4) title('Frequency spectrum of Output of Hamming
window');
    end
    if(i==5) title('Frequency spectrum of Output of Blackman
window');
    end
    xlabel('Frequency ( in radian )');
    ylabel('AMPLITUDE');
    irs= ifft(rs);
    subplot(2,1,2);

    plot(-pi:pi/256:pi*255/256,fftshift(abs(nrs)));

    if(i==1) title(' Frequency spectrum of Output of Rectangular
window of Noisy Signal');
    end
    if(i==2) title('Frequency spectrum of Output of Triangular
window of Noisy Signal');
    end
    if(i==3) title('Frequency spectrum of Output of Hanning
window of Noisy Signal');
    end
    if(i==4) title('Frequency spectrum of Output of Hamming
window of Noisy Signal');
    end
    if(i==5) title('Frequency spectrum of Output of Blackman
window of Noisy Signal');
    end
    xlabel('Frequency ( in radian )');
    ylabel('AMPLITUDE');
end
disp('Input SNR is');
disp(10*log10(inp_snr));
disp('Output snr is');
  disp(10*log10(op_snr));
```

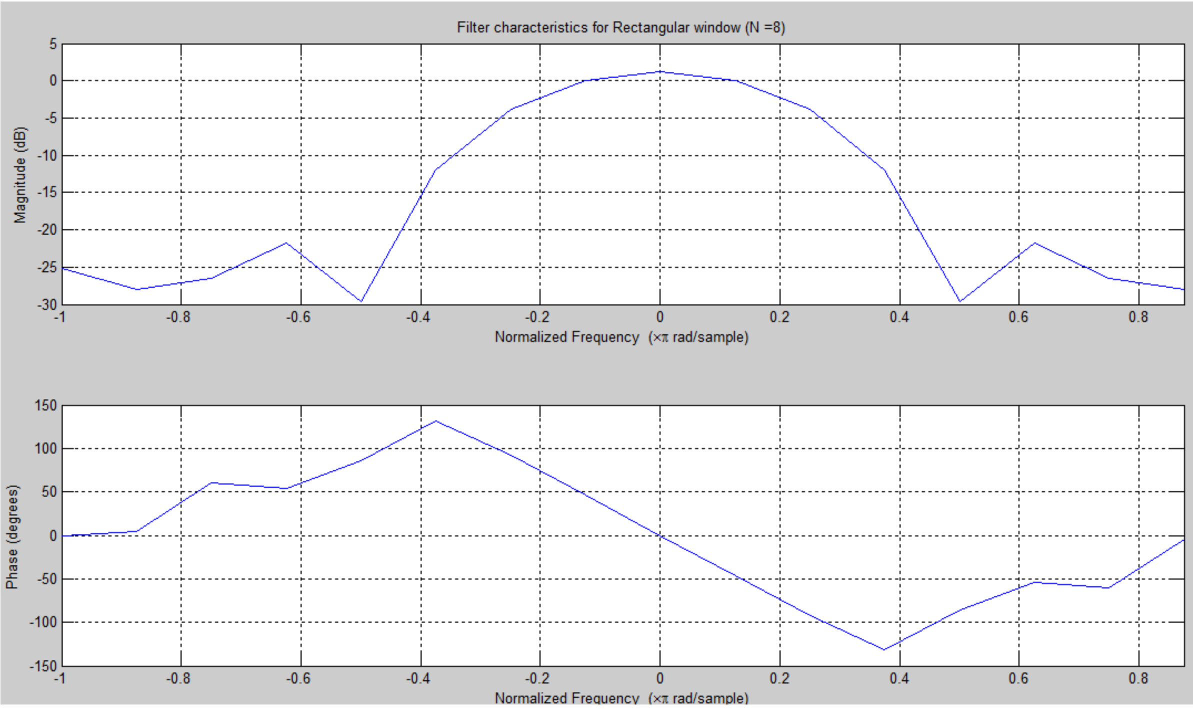
152.2 RESULT: Sinc function for N=512



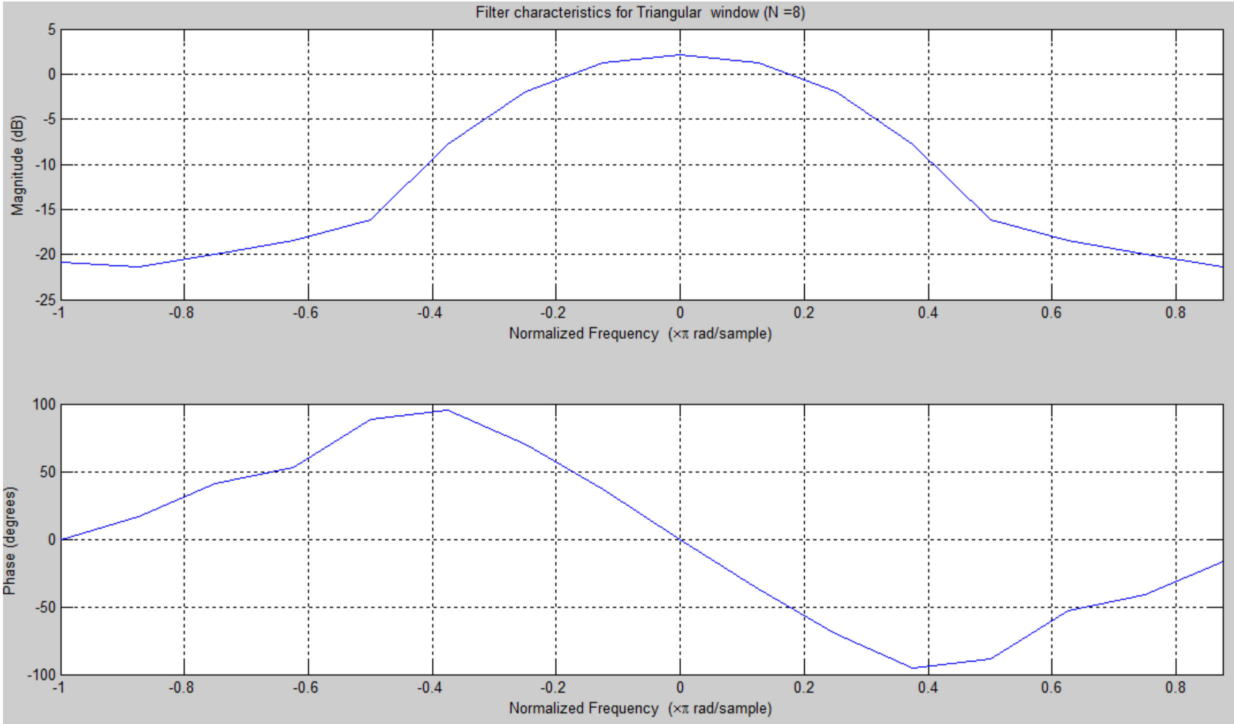
FILTER CHARACTERISTICS

A. N=8

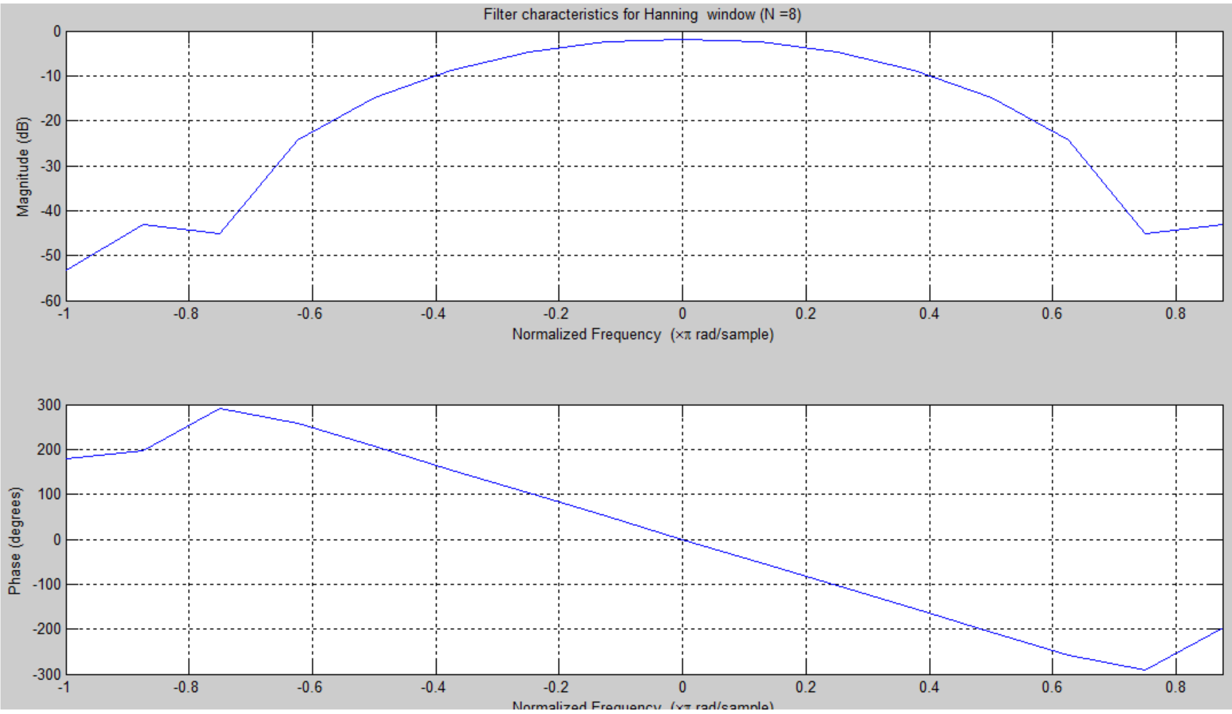
- **RECTANGULAR WINDOW**



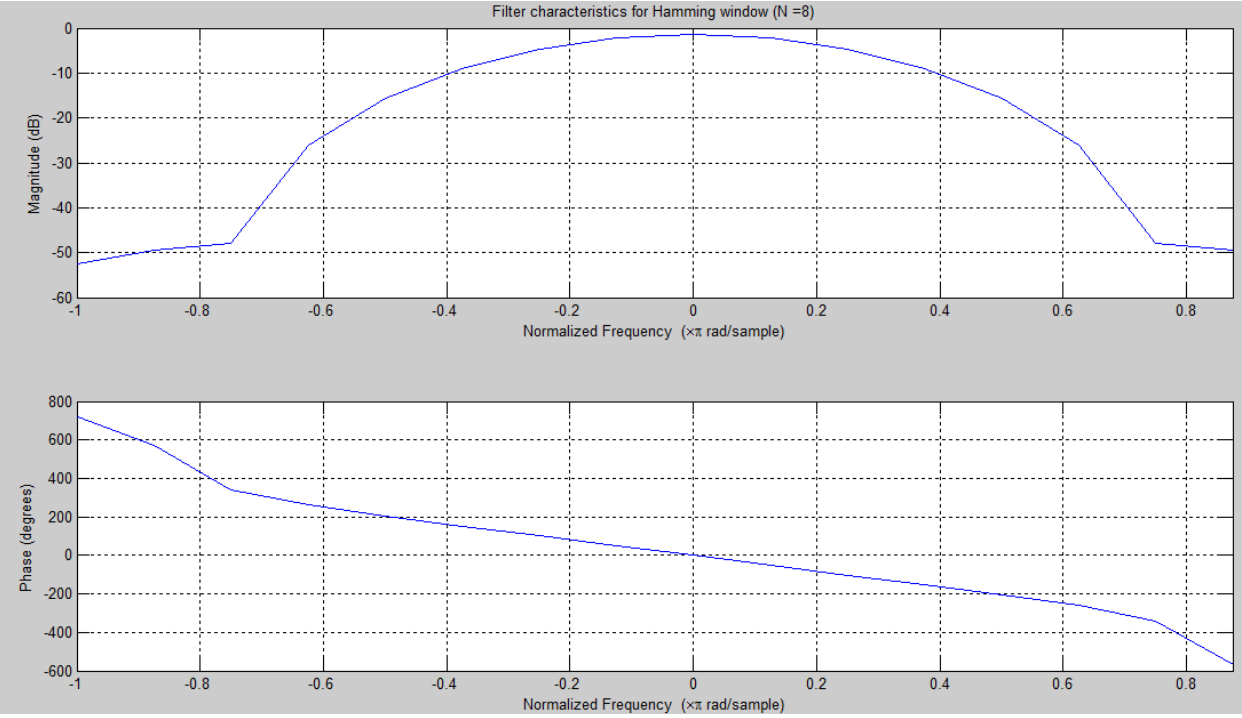
- **Triangular Window**



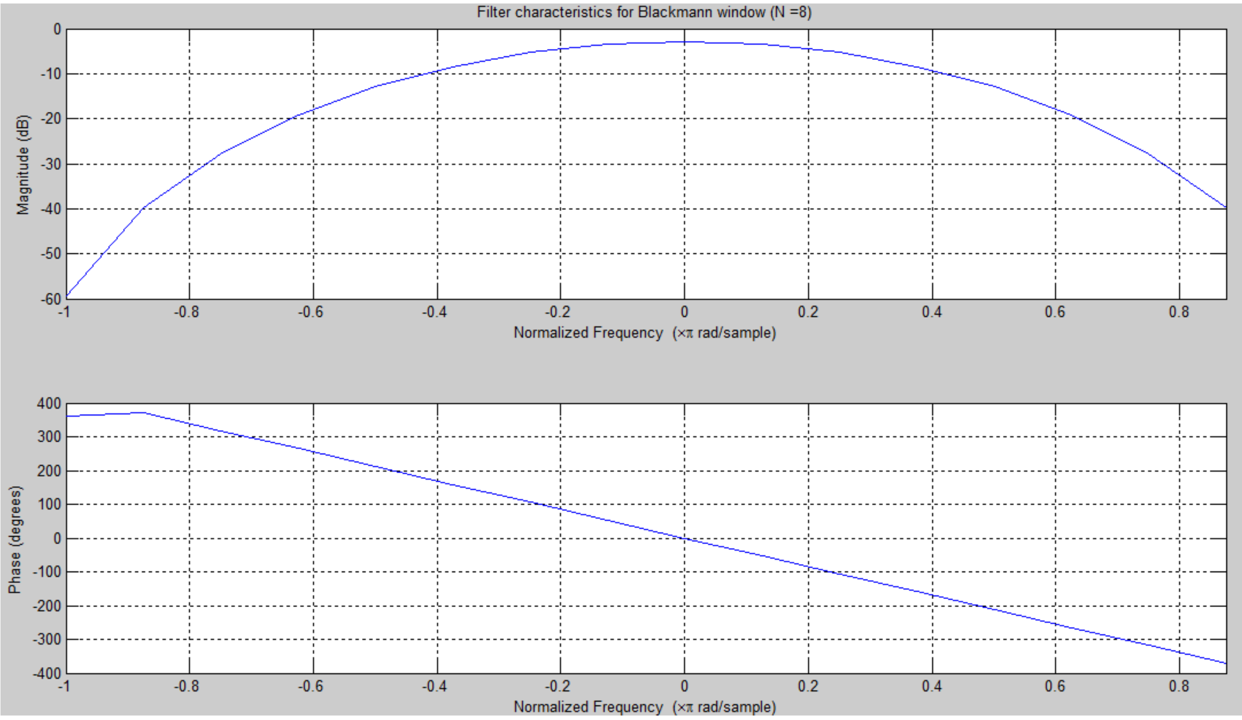
- **Hanning Window**



- ## Hamming Window

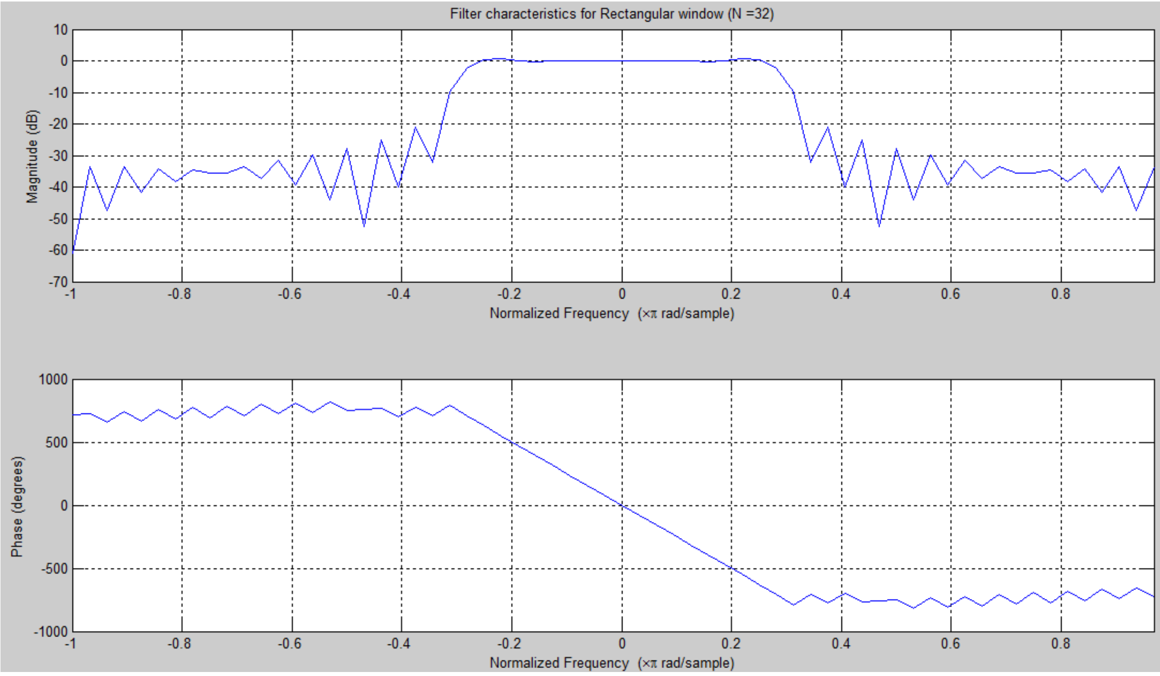


- ## Blackmann Window

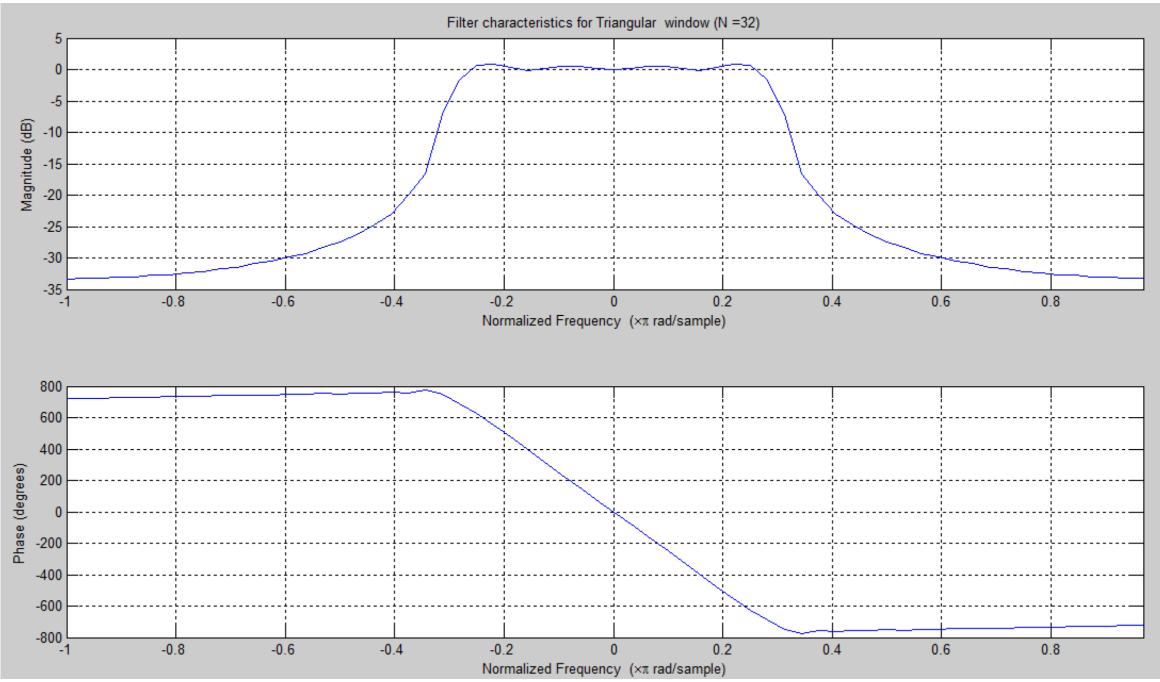


A. N=32

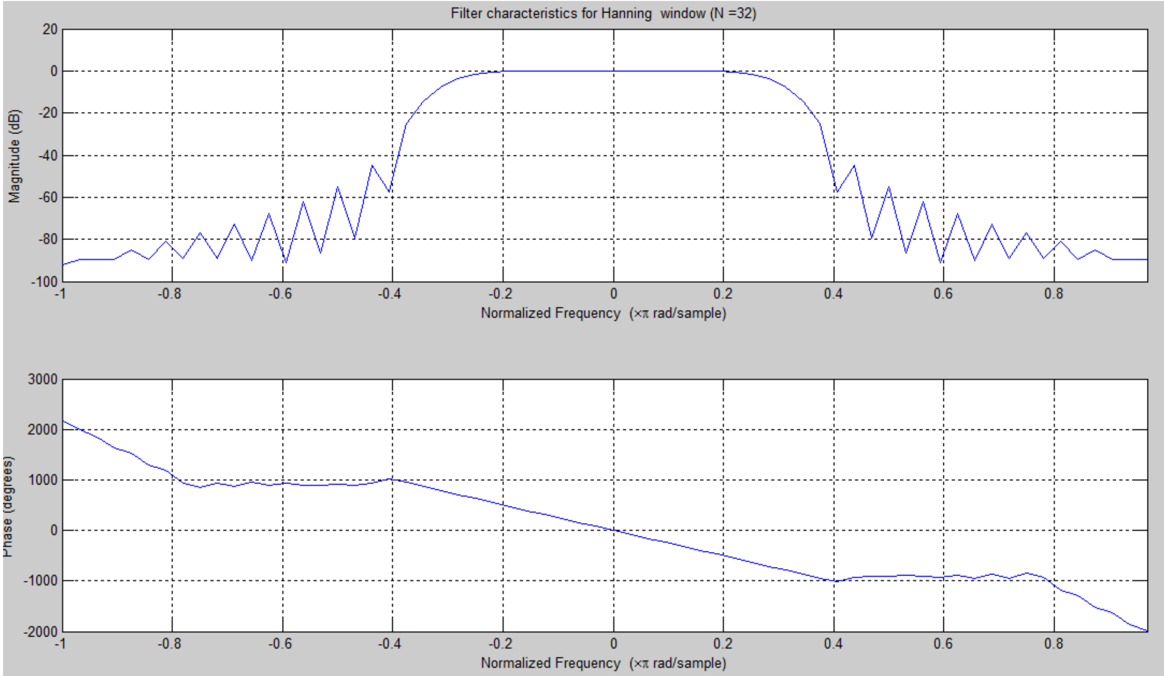
- **Rectangular Window**



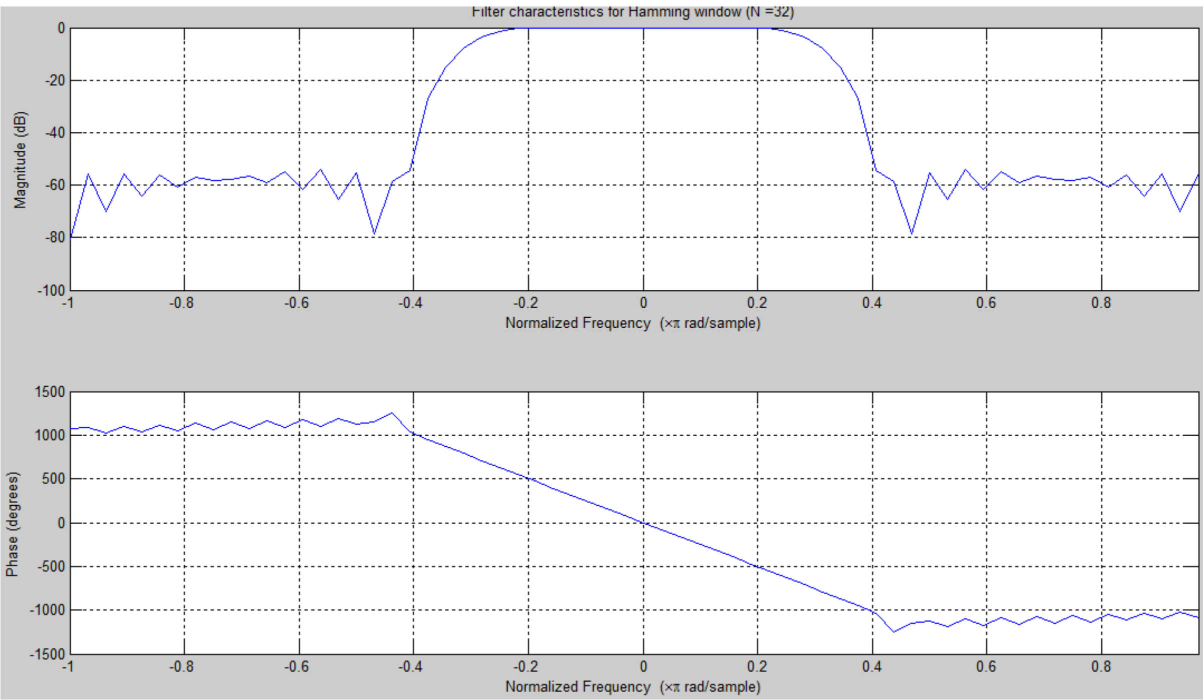
- **Triangular Window**



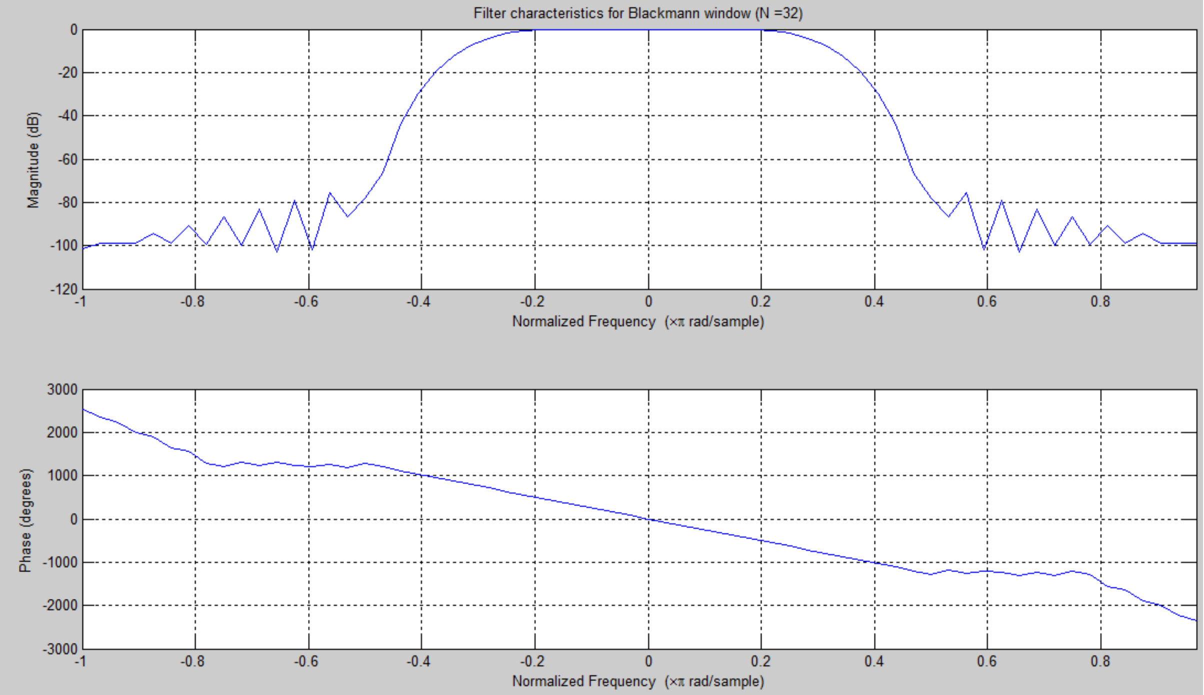
• Hanning Window



• Hamming Window

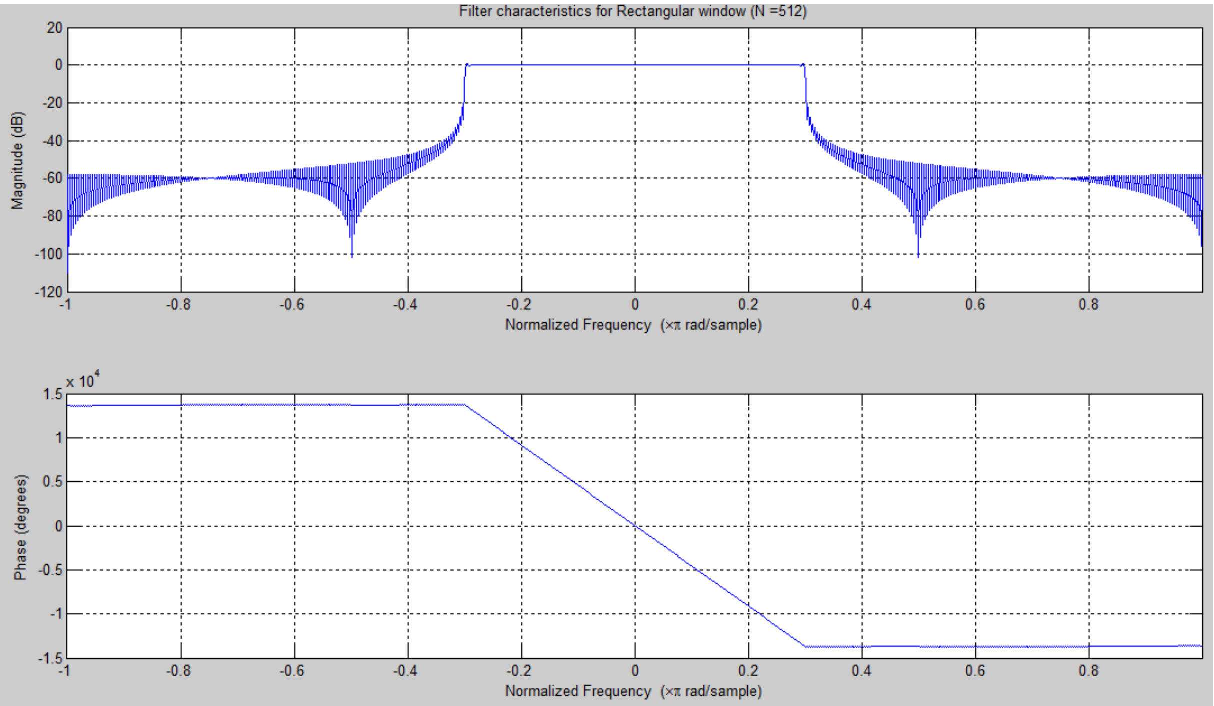


- **Blackmann Window**

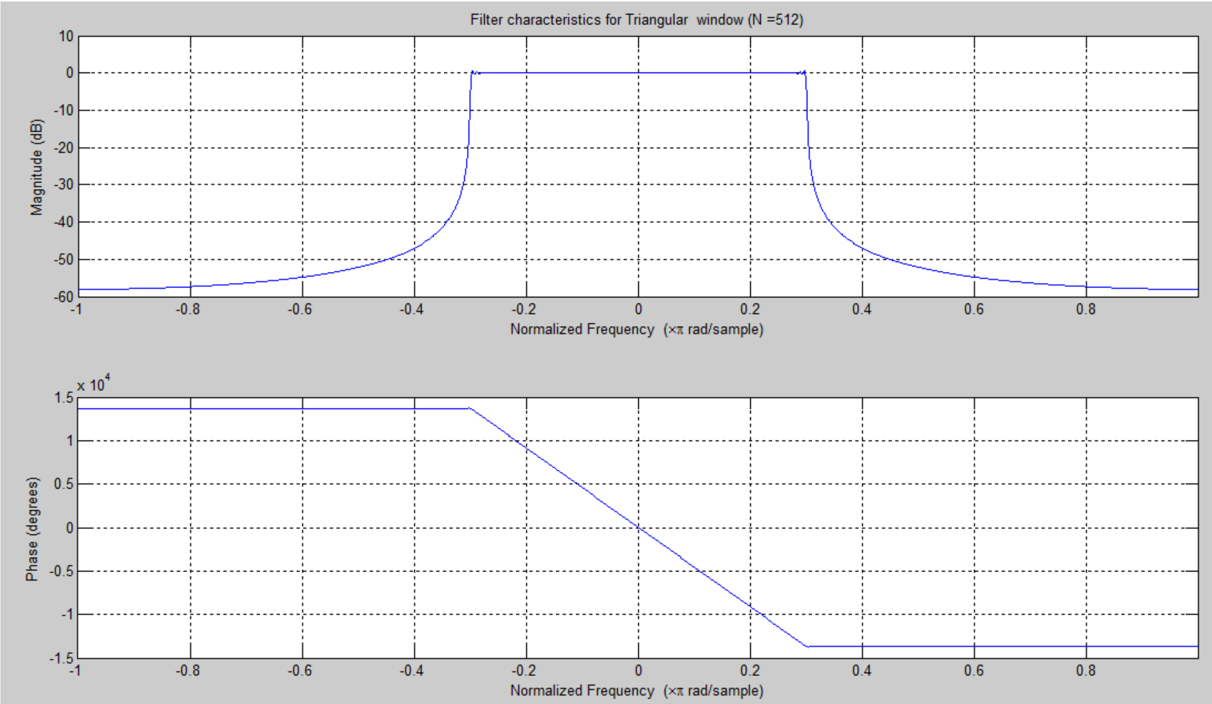


C. N=512

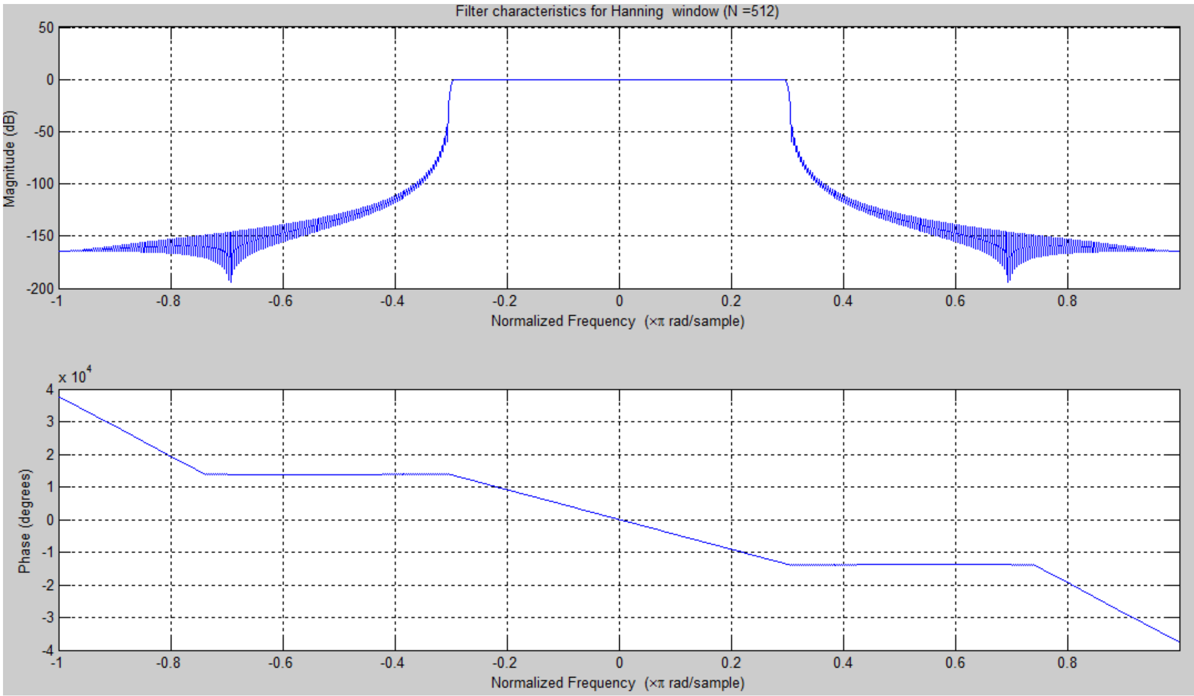
- **Rectangular Window**



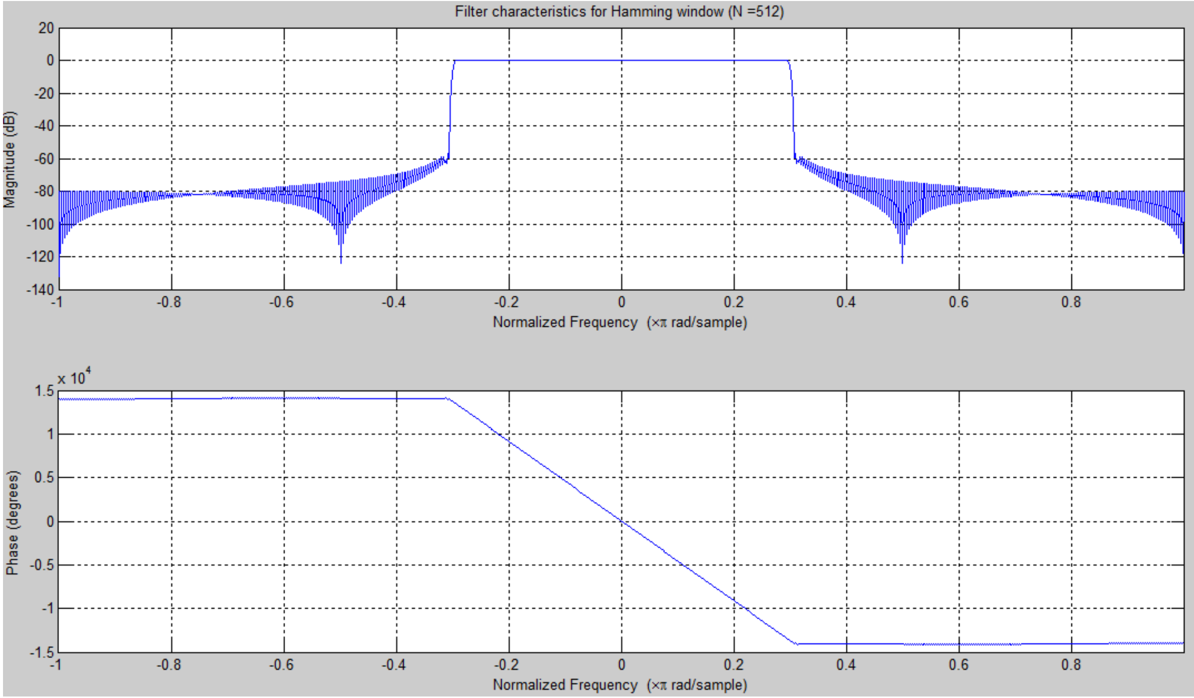
- ## Triangular Window



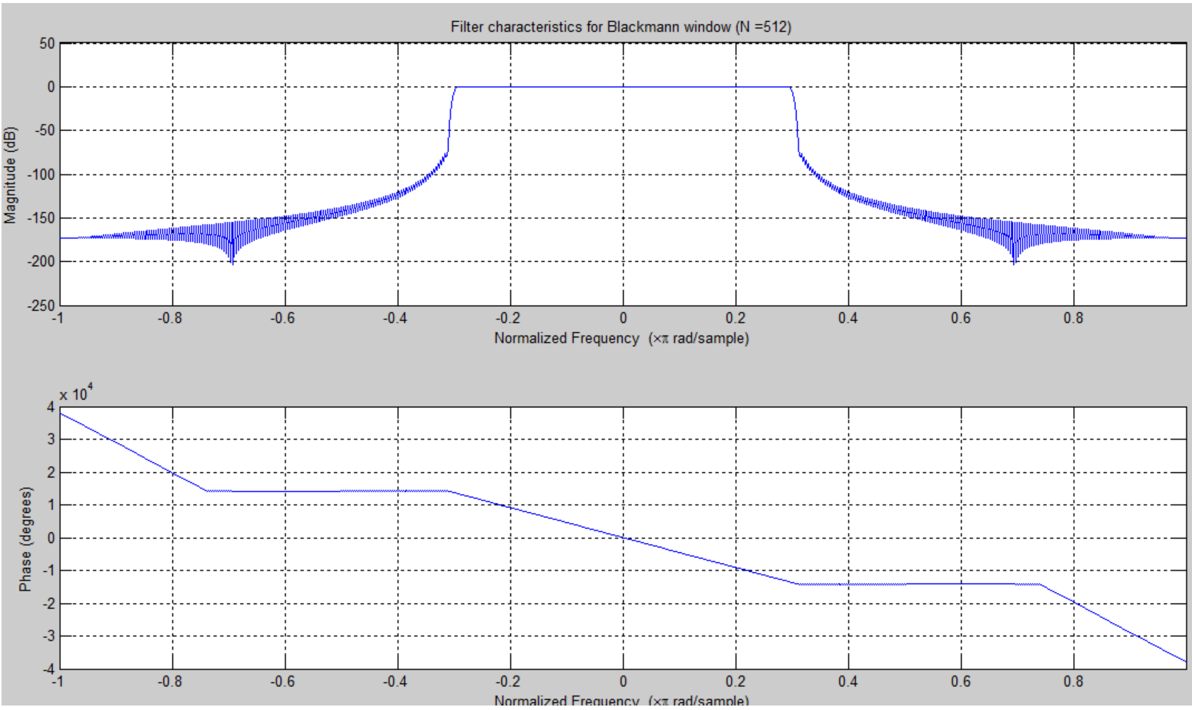
- ## Hanning Window



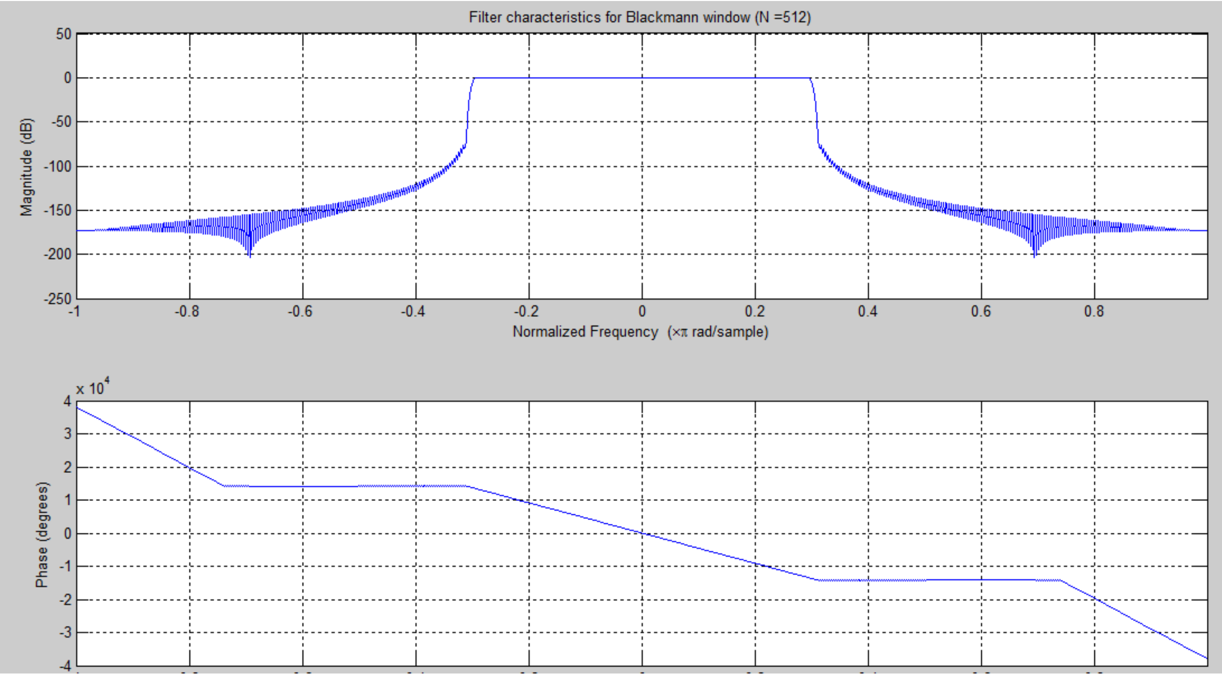
- ## Hamming Window



- ## Blackmann Window



- ## Blackmann Window



Filter Parameters

Table 1 Rectangular window

N	Transition width	Peak of first lobe	Maximum stop band attenuation
8	0.2	-22	-30
64	0.1	-33	-50
512	0.06	-40	-100

Table 2 Triangular window

N	Transition width	Peak of first lobe	Maximum stop band attenuation
8	0.3	-15	-22
64	0.16	-28	-45
512	0.07	-40	-60

Table 3 Hanning window

N	Transition width	Peak of first lobe	Maximum stop band attenuation
8	0.5	-42	-52
64	0.3	-80	-120
512	0.1	-100	-200

Table 4 Hamming window

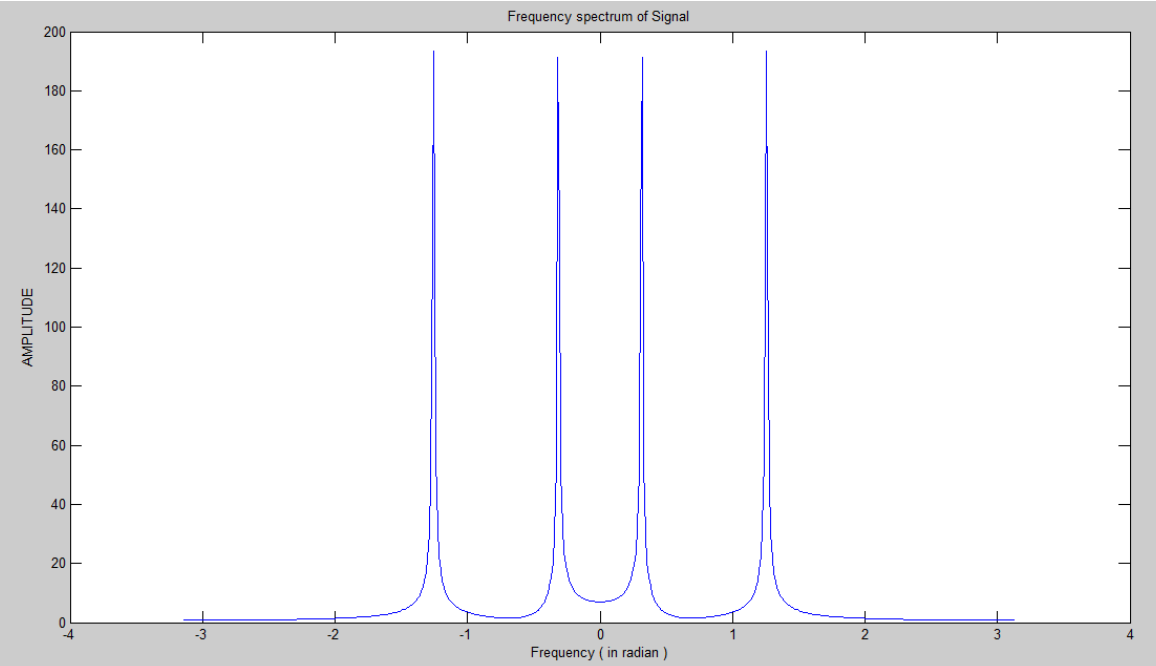
N	Transition width	Peak of first lobe	Maximum stop band attenuation
8	0.54	-48	-52
64	0.1	-58	-72
512	0.02	-60	-120

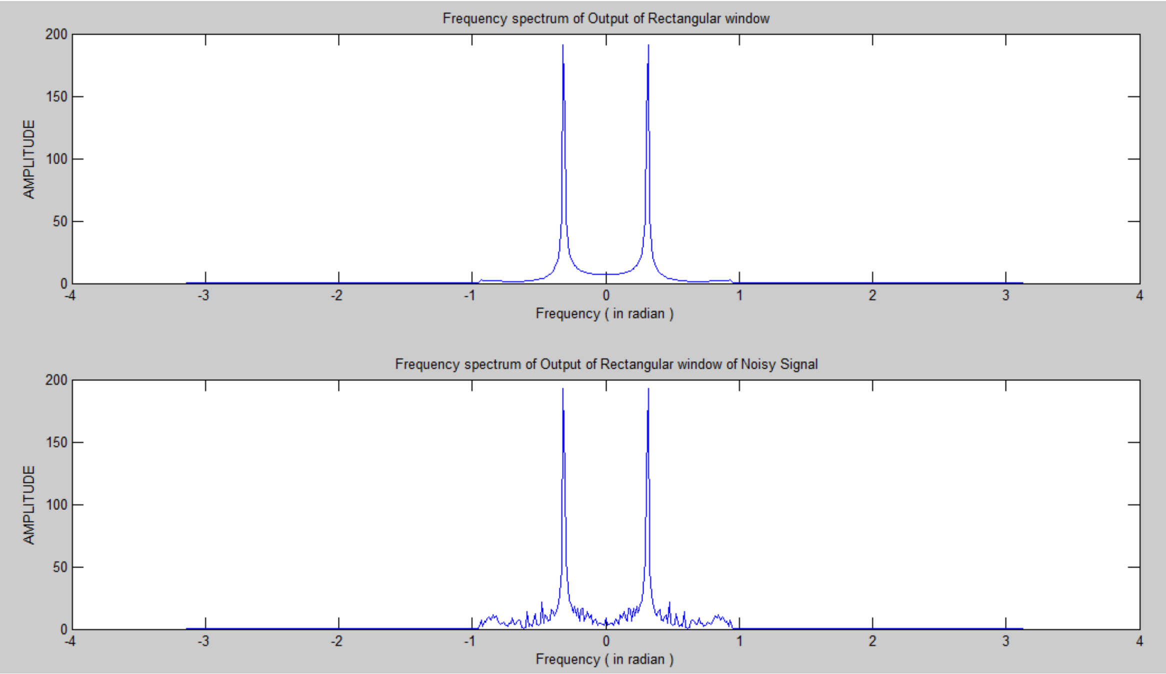
Table 5 Blackmann window

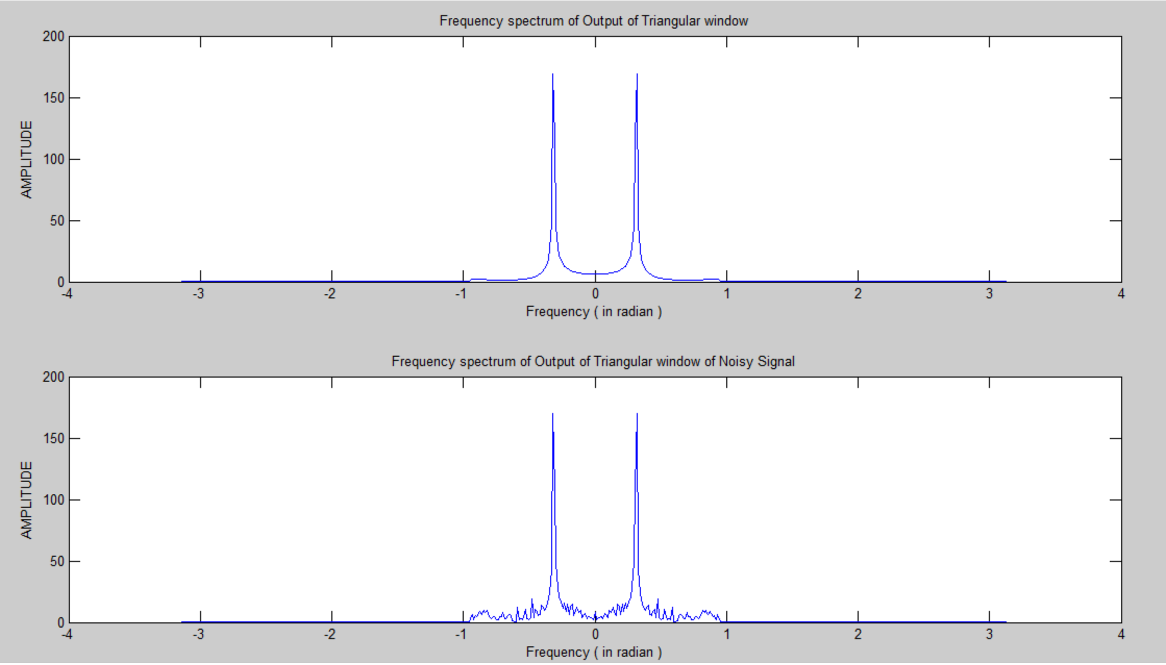
N	Transition width	Peak of first lobe	Maximum stop band attenuation
8	0.4	-26	-60
64	0.05	-76	-130
512	0.01	-80	-200

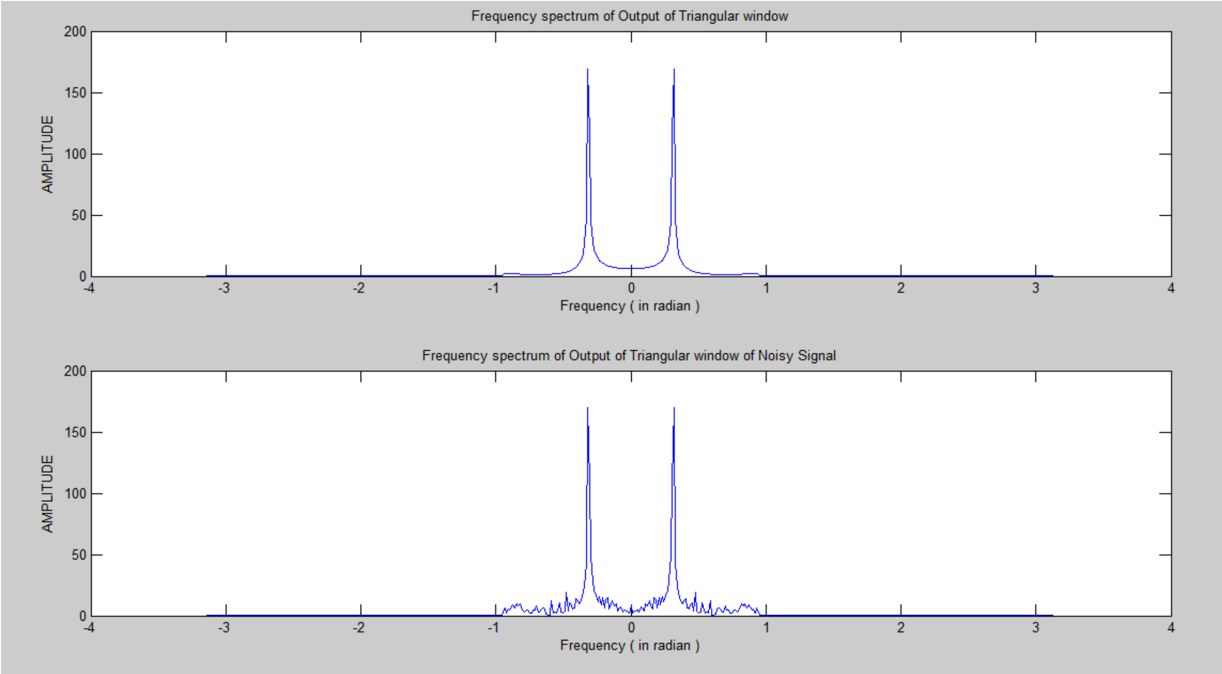
Filter Output

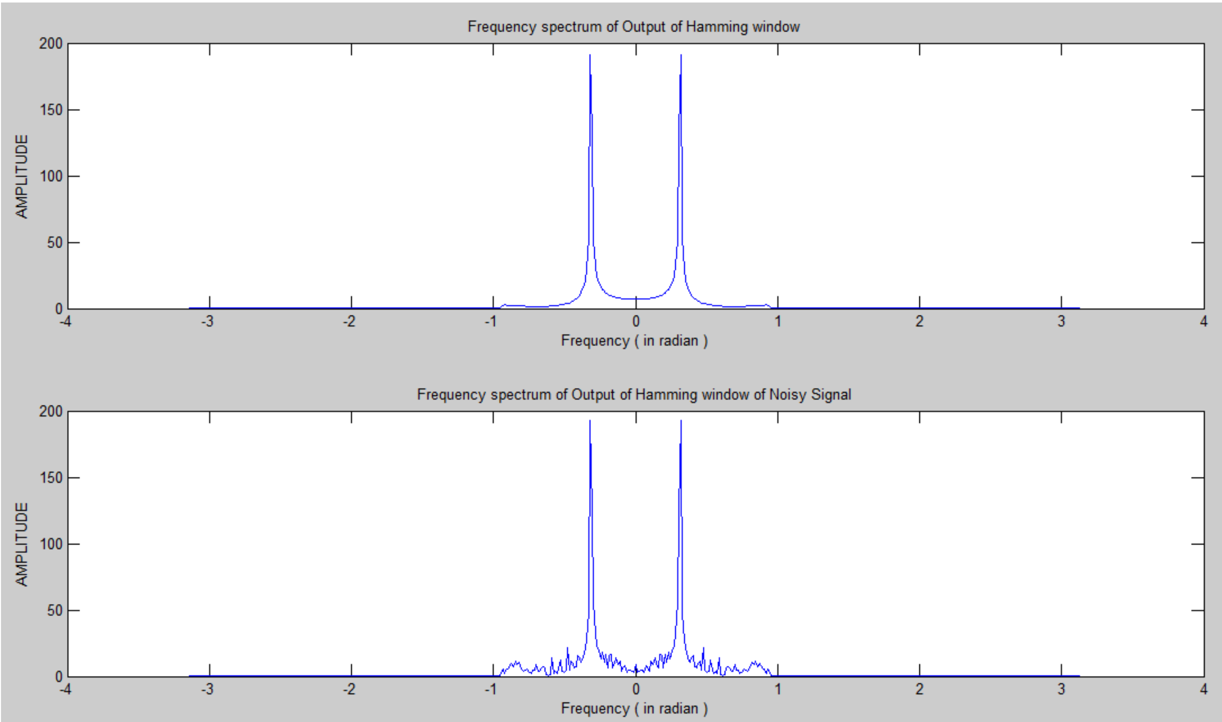
Frequency Spectrum of signal



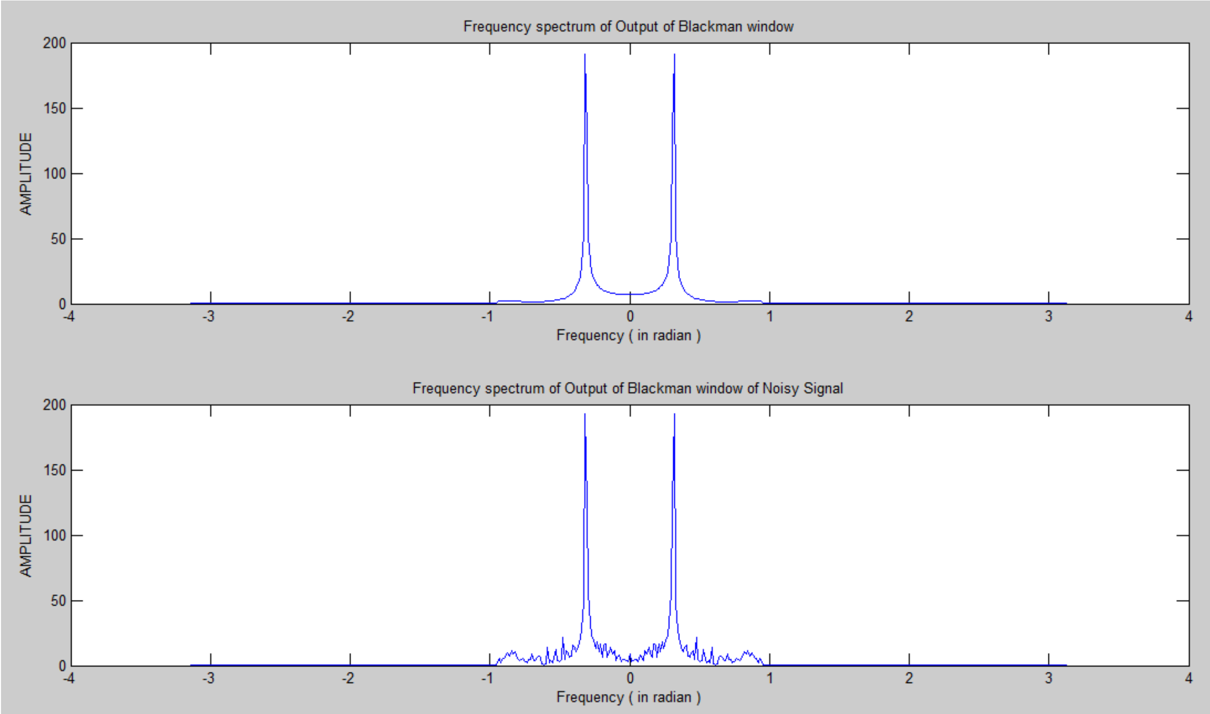

Frequency spectrum of Output of Rectangular window
AMPLITUDE
Frequency (in radian)
Frequency spectrum of Output of Rectangular window of Noisy Signal
AMPLITUDE
Frequency (in radian)


Frequency spectrum of Output of Triangular window
AMPLITUDE
Frequency (in radian)
Frequency spectrum of Output of Triangular window of Noisy Signal
AMPLITUDE
Frequency (in radian)


Frequency spectrum of Output of Triangular window
AMPLITUDE
Frequency (in radian)
Frequency spectrum of Output of Triangular window of Noisy Signal
AMPLITUDE
Frequency (in radian)


Frequency spectrum of Output of Hamming window
AMPLITUDE
Frequency (in radian)
Frequency spectrum of Output of Hamming window of Noisy Signal
AMPLITUDE
Frequency (in radian)

\



- ## Rectangular Window

Input Signal Amplitude (in dB)	Output Signal Amplitude (in dB)	Noise Amplitude (in dB)	SNR(input) (in dB)	SNR(output) (in dB)
0	-0.3	-7.75	7.75	4.9473

- ## Triangular Window

Input Signal Amplitude (in dB)	Output Signal Amplitude (in dB)	Noise Amplitude (in dB)	SNR(input) (in dB)	SNR(output) (in dB)
0	-0.27	-7.75	7.75	5.446

- ## Hanning Window

Input Signal Amplitude (in dB)	Output Signal Amplitude (in dB)	Noise Amplitude (in dB)	SNR(input) (in dB)	SNR(output) (in dB)
0	-0.28	-7.75	7.75	4.971

- ## Hamming Window

Input Signal Amplitude (in dB)	Output Signal Amplitude (in dB)	Noise Amplitude (in dB)	SNR(input) (in dB)	SNR(output) (in dB)
0	-0.29	-7.75	7.75	4.971

- ## Blackmann Window

Input Signal Amplitude (in dB)	Output Signal Amplitude (in dB)	Noise Amplitude (in dB)	SNR(input) (in dB)	SNR(output) (in dB)
0	-0.30	-7.75	7.75	4.94

Discussion:

Ved Prakash Tiwari
12EC10061

- Windowing helps in realizing causal low pass filter which cannot be realized in its ideal form because of its infinite duration impulse response.

- Different window coefficients meet different requirement of phase response and pass band and stop band attenuation.

- We observed that pass band maximum attenuation is more stable in higher coefficient filters. But as n increase number of side lobes also increases.

- Maximum stop-band attenuation is attained for Blackmann window with filter coefficient is equal to 512, while for triangular window we get maximally linear phase response.

DISCUSSION:

DEEP KIRAN SHROTI
12EC30042

- The output signal SNR was lower than input signal SNR because the noise after filtering gets spread in the entire frequency domain due to windowing and the output SNR was calculated with respect to the original input signal.
- Windowing functions are used because ideal filters have sharp cut-off which are not realizable in practice since their impulse response extend up to infinity.
- Five windowing functions were used and each function had a different frequency response.
- It was also observed that as the order N increases, we get a more stable stop band attenuation.
- When designing digital FIR filters using window functions it is necessary to specify:

  A window function to be used and the filter order according to the required specifications (selectivity and stopband attenuation).