LAB REPORT

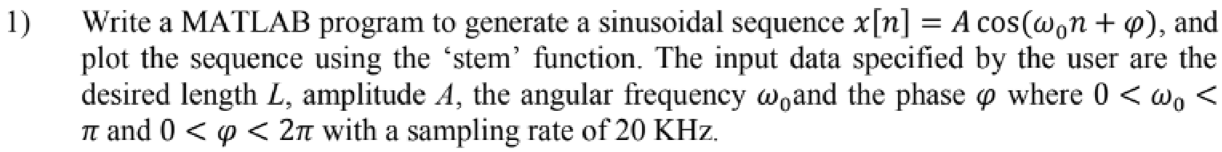
DSP LAB

HIMANSHU GUPTA
(510719033)

ETC-6th SEM

Assignment 1

Problem 1:

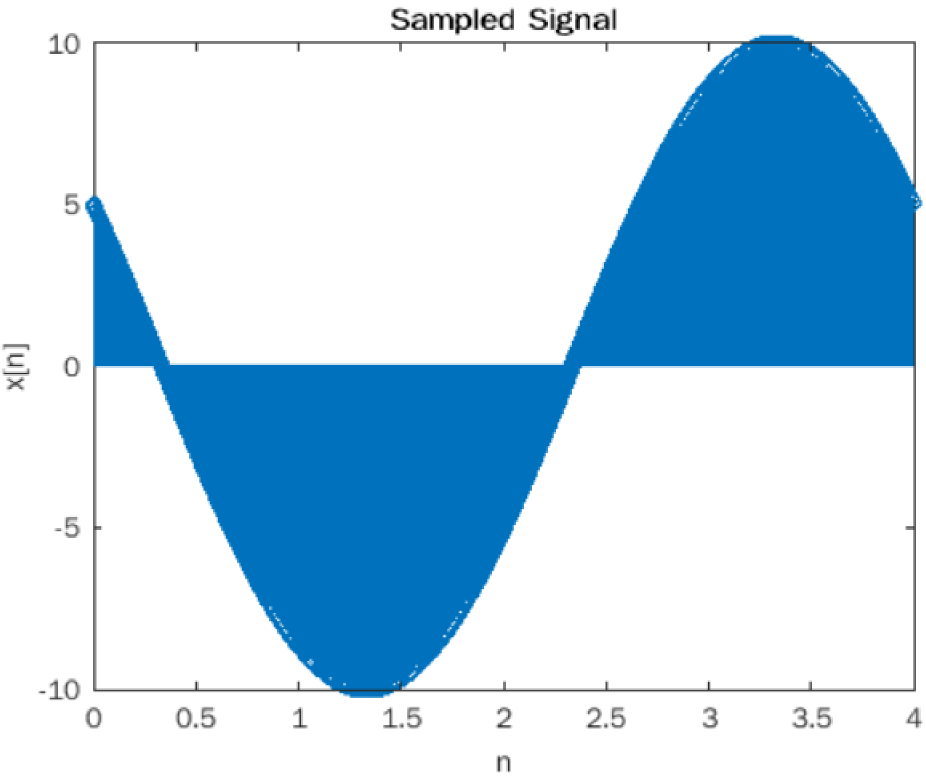


1) Write a MATLAB program to generate a sinusoidal sequence $x[n] = A\cos(\omega_0 n + \varphi)$, and plot the sequence using the 'stem' function. The input data specified by the user are the desired length L, amplitude A, the angular frequency $\omega_0$and the phase φ where $0 < \omega_0 < \pi$ and $0 < \varphi < 2\pi$ with a sampling rate of 20 KHz.

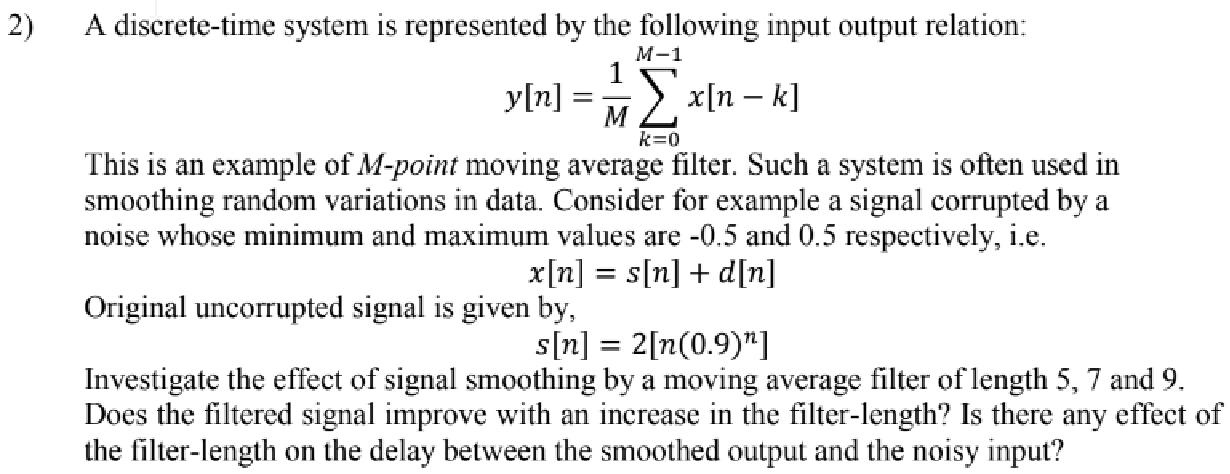
Code:

```
A=10;
L=80000;

w0=pi/2;
fi=pi/3;
Fs=20000;
n=0:1/Fs:(1/Fs)*(L-1);
x=A*cos(w0*n+ fi);

stem(n,x);
title('Sampled Signal');
xlabel('n');
ylabel('x[n]')
```

Graph:



Problem 2:



2) A discrete-time system is represented by the following input output relation:

$$y[n] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-k]$$

This is an example of *M-point* moving average filter. Such a system is often used in smoothing random variations in data. Consider for example a signal corrupted by a noise whose minimum and maximum values are -0.5 and 0.5 respectively, i.e.

$$x[n] = s[n] + d[n]$$

Original uncorrupted signal is given by,

$$s[n] = 2[n(0.9)^n]$$

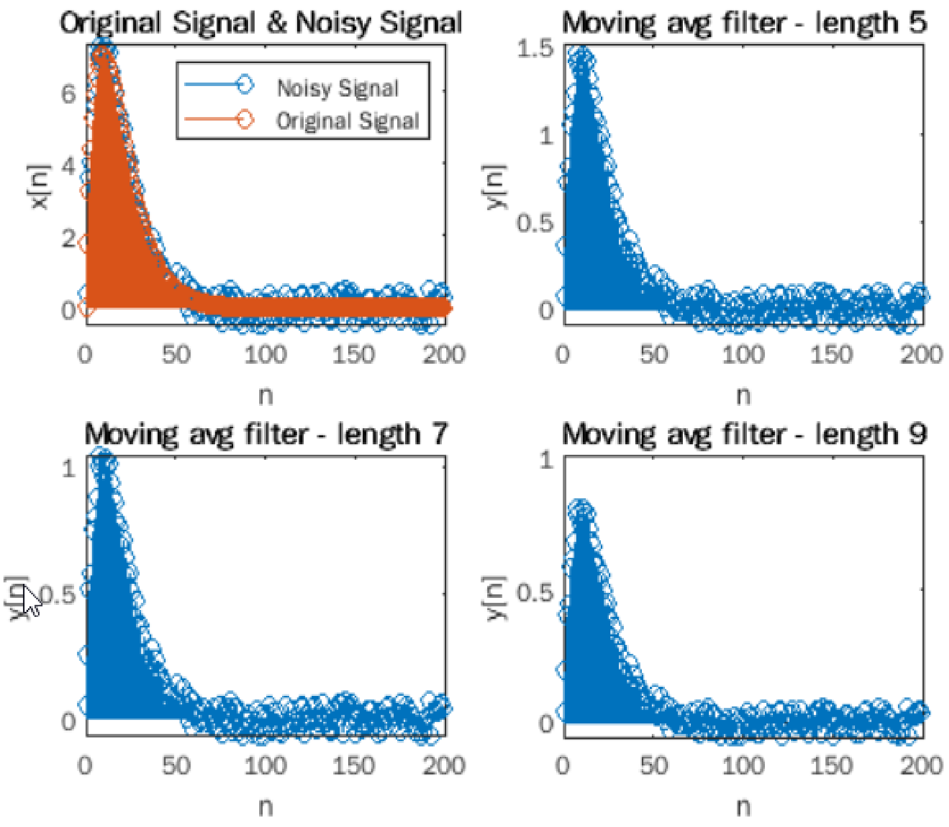
Investigate the effect of signal smoothing by a moving average filter of length 5, 7 and 9. Does the filtered signal improve with an increase in the filter-length? Is there any effect of the filter-length on the delay between the smoothed output and the noisy input?

Code:

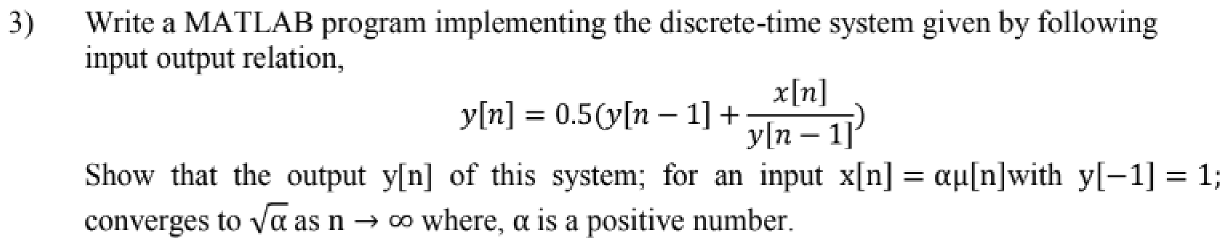
```
n=0:1:200-1;
s=2*(n.*(0.9).^n);
d = -0.5 + rand(1,200);
x=s+d;

subplot(2,2,1);
stem(n,x);
hold on
stem(n,s);
hold off
title('Original Signal & Noisy Signal ');
xlabel('n');
ylabel('x[n]');
legend('Noisy Signal','Original Signal');

m1=5;
y1=zeros(1,200);
for index=0:1:m1-1;
   y1=y1+delayseq(x,index);
end
y1=y1/m1;
subplot(2,2,2);
stem(n,y1);
title('Moving avg filter - length 5');
xlabel('n');
ylabel('y[n]');

m2=7;
y2=zeros(1,200);
for index=0:1:m2-1;
   y2=y2+delayseq(x,index);
end
y2=y2/m2;
subplot(2,2,3);
stem(n,y2);
title('Moving avg filter - length 7');
xlabel('n');
ylabel('y[n]');
```

```
m3=9;
y3=zeros(1,200);
for index=0:1:m3-1;
   y3=y3+delayseq(x,index);
end
y3=y3/m3;
subplot(2,2,4);
stem(n,y3);

title('Moving avg filter - length 9');
xlabel('n');
ylabel('y[n]');
```

Graph:



Problem 3:



3) Write a MATLAB program implementing the discrete-time system given by following input output relation,

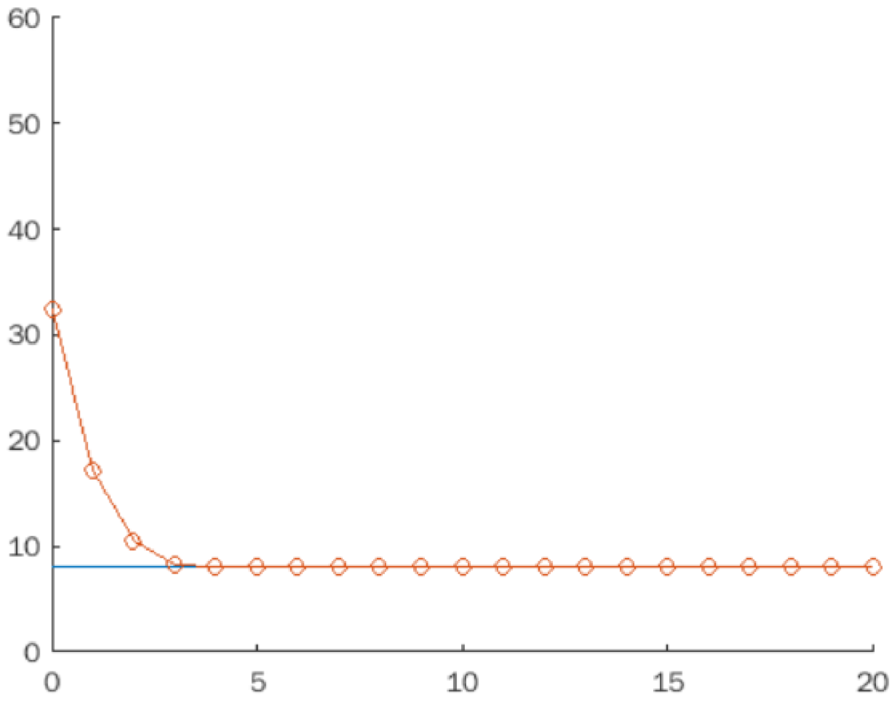
$$y[n] = 0.5(y[n-1] + \frac{x[n]}{y[n-1]})$$

Show that the output y[n] of this system; for an input $x[n] = \alpha\mu[n]$with $y[-1] = 1$; converges to $\sqrt{\alpha}$ as $n \rightarrow \infty$ where, α is a positive number.

Code:

```
close all;
clear;
clc;

L = 20;
u = @(n) 1*(n>=0);

n = (-1:L);
a = 64;
x = a*u(n);

y = zeros(1,length(x));

y(1) = 1;

for index = 2:length(x)
y(index) = 0.5*(y(index-1)+(x(index)/y(index-1)));
end

figure;
hold on;
plot([0 n(end)], [sqrt(a) sqrt(a)]);
plot(n(2:end),y(2:end), '-o');
ylim([0 60]);
hold off;
```

Graph:



Problem 4:

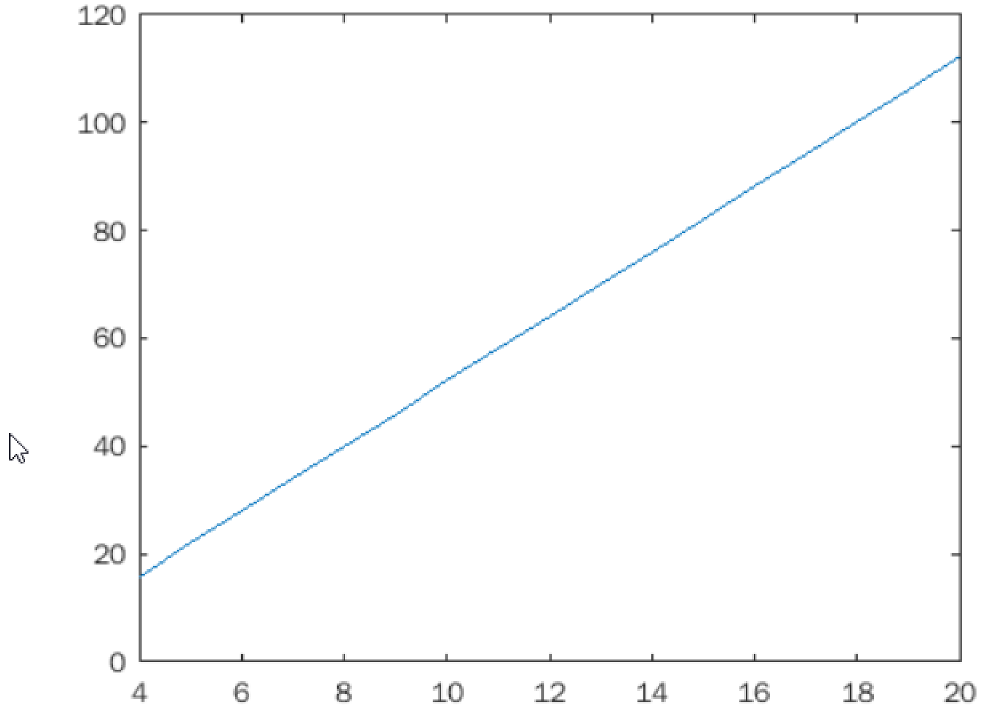
4) Plot the given input speech file in MATLAB and write a program to implement a quantizer for the given speech file. Plot the variation of signal to quantization noise ratio (SNR_q)against variation of number of *bits/sample*.

Code:

```
close all;
clear;
clc;
load mtlb;
t = (0:length(mtlb)-1)/Fs;
u = max(mtlb);
l = min(mtlb);
b = 4:20;
s = zeros(1,length(b));
i = 1;
for nb = b
    del = (u - l)/(2^nb);
    par = l:del:u;
    cb = l-del/2:del:u+del/2;
    [index,quants] = quantiz(mtlb,par,cb);
    N = rms(mtlb - quants')^2;
```

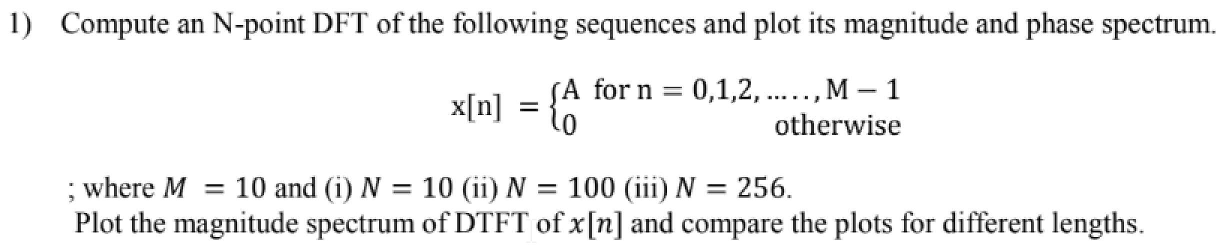
```
        S = rms(mtlb)^2;
        s(i) = 10*log10(S/N);
        i = i+1;
    end

    figure;
    plot(b, s);
```

Graph:



Assignment 2

Problem 1:



1) Compute an N-point DFT of the following sequences and plot its magnitude and phase spectrum.

$$x[n] = \begin{cases} A & \text{for } n = 0,1,2,\ldots\ldots,M-1 \\ 0 & \text{otherwise} \end{cases}$$

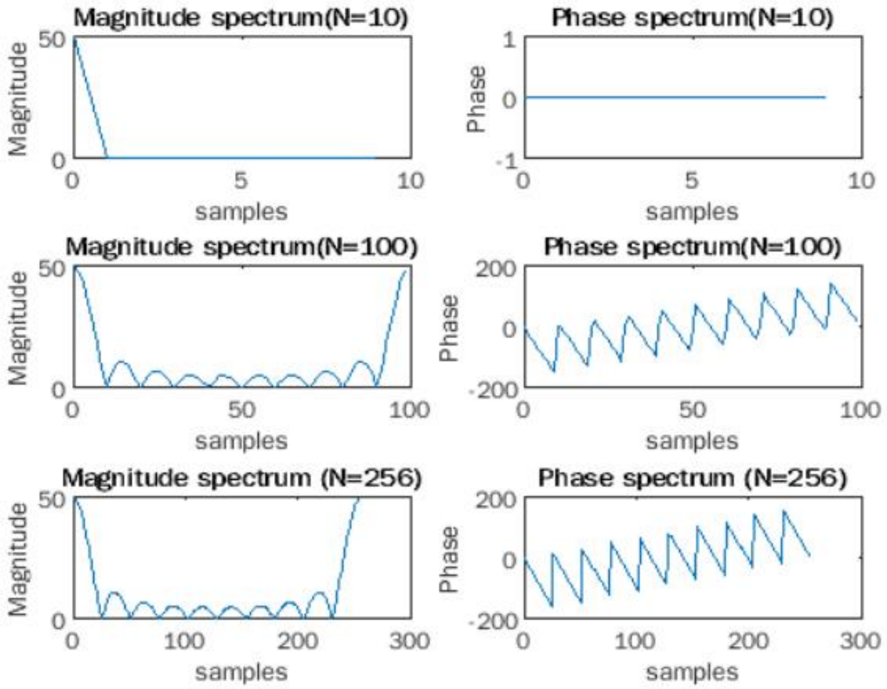
; where $M = 10$ and (i) $N = 10$ (ii) $N = 100$ (iii) $N = 256$.
Plot the magnitude spectrum of DTFT of $x[n]$ and compare the plots for different lengths.

Code:

```
M=10;
A=5;
x= A*ones(1,M);
n1=10;n2=100;n3=256;
y1=fft(x,n1);
y2=fft(x,n2);
y3=fft(x,n3);

a1=angle(y1);
a2=angle(y2);
a3=angle(y3);

subplot(3,2,1)
plot(0:n1-1,abs(y1))
title('Magnitude spectrum(N=10)')
xlabel('samples')
ylabel('Magnitude')

subplot(3,2,2)
plot(0:n1-1,rad2deg(a1))
title('Phase spectrum(N=10)')
xlabel('samples')
ylabel('Phase')

subplot(3,2,3)
plot(0:n2-1,abs(y2))
title('Magnitude spectrum(N=100)')
xlabel('samples')
ylabel('Magnitude')

subplot(3,2,4)
plot(0:n2-1,rad2deg(a2))
title('Phase spectrum(N=100)')
xlabel('samples')
ylabel('Phase')

subplot(3,2,5)
plot(0:n3-1,abs(y3))
title('Magnitude spectrum (N=256)')
xlabel('samples')
```

```
ylabel('Magnitude')

subplot(3,2,6)
plot(0:n3-1,rad2deg(a3))
title('Phase spectrum (N=256)')
xlabel('samples')
ylabel('Phase')
```

Graph:



Magnitude spectrum(N=10)

Phase spectrum(N=10)

Magnitude spectrum(N=100)

Phase spectrum(N=100)

Magnitude spectrum (N=256)

Phase spectrum (N=256)

Problem 2:

2) Write a program to plot the magnitude and phase response of discrete-time system characterized by its impulse response:

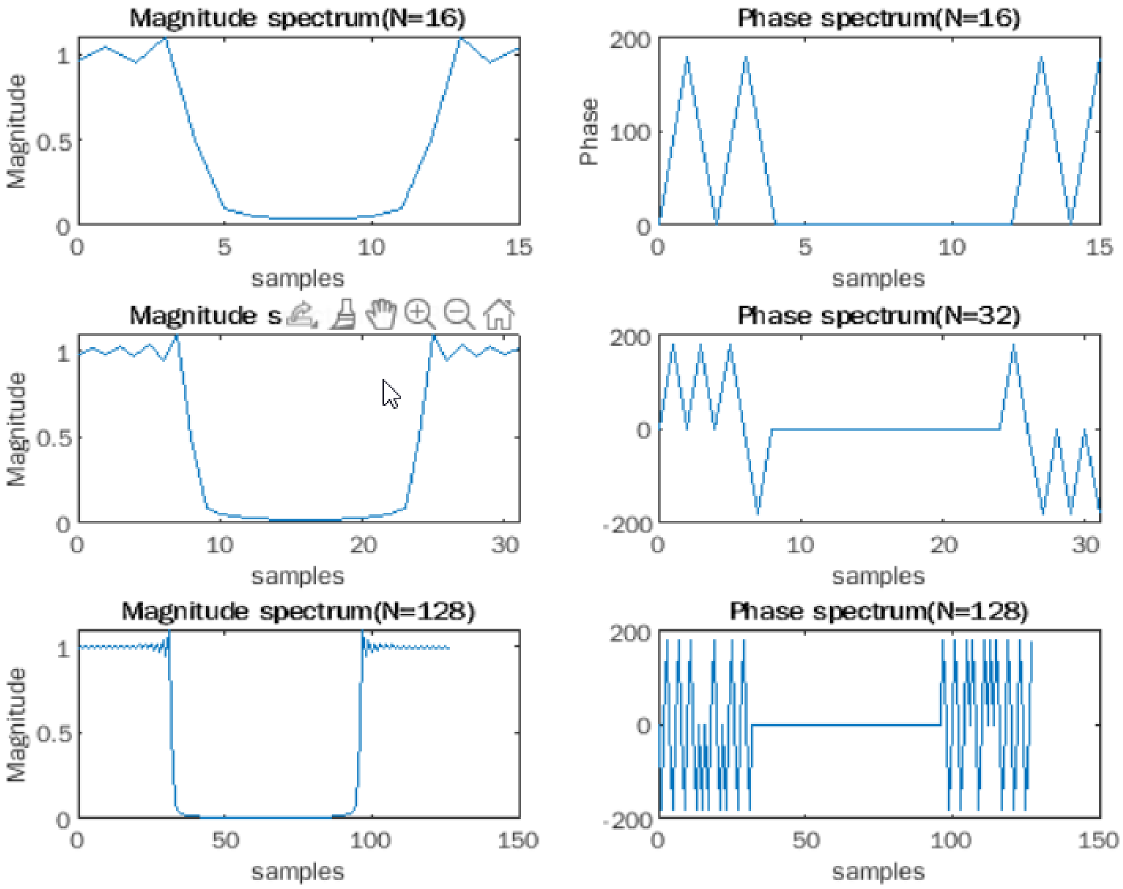
$$h[n] = \begin{cases} 0.5 & \text{for } n = 0 \\ \dfrac{\sin 0.5\pi n}{\pi n} & \text{otherwise} \end{cases}$$

(i) $n = -8:7$
(ii) $n = -16:15$
(iii) $n = -64:63$

Code:

```
clear;

% spectrum with N=16
n1=-8:7;
for i=n1
h1(i+9)= sin(0.5*pi*i)/(pi*i);
end
h1(9)=0.5;
y1=fft(h1,length(n1));
a1=angle(y1);
subplot(3,2,1)
plot(0:length(n1)-1,abs(y1))
title('Magnitude spectrum(N=16)')
xlabel('samples')
ylabel('Magnitude')
subplot(3,2,2)
plot(0:length(n1)-1,rad2deg(a1))
title('Phase spectrum(N=16)')
xlabel('samples')
ylabel('Phase')

% spectrum with N=32
n2=-16:15;
for i=n2
h2(i+17)= sin(0.5*pi*i)/(pi*i);
end
h2(17)=0.5;
y2=fft(h2,length(n2));
arg2=angle(y2);
subplot(3,2,3)
plot(0:length(n2)-1,abs(y2))
title('Magnitude spectrum(N=32)')
xlabel('samples')
ylabel('Magnitude')
subplot(3,2,4)
plot(0:length(n2)-1,rad2deg(arg2))
title('Phase spectrum(N=32)')
xlabel('samples')

% spectrum with N=128
n3=-64:63;
for i=n3
h3(i+65)= sin(0.5*pi*i)/(pi*i);
end
h3(65)=0.5;
y3=fft(h3,length(n3));
arg3=angle(y3);
subplot(3,2,5)
plot(0:length(n3)-1,abs(y3))
title('Magnitude spectrum(N=128)')
xlabel('samples')
ylabel('Magnitude')
subplot(3,2,6)
plot(0:length(n3)-1,rad2deg(arg3))
title('Phase spectrum(N=128)')
xlabel('samples')
```

Graph:



Problem 3:

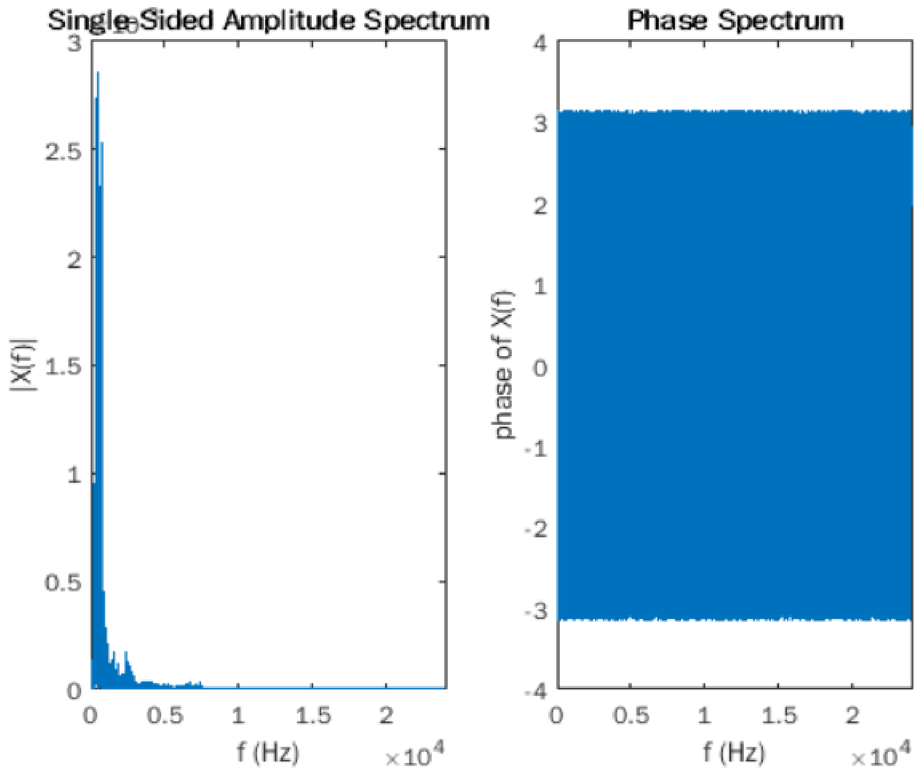
3) Evaluate and plot the spectrum of your own voice signal.

Code:

```
clear;
[x, Fs]= audioread("Recording (2).m4a");
sound(x,Fs)
length=length(x);
y = fft(x);
p2 = (y/length);
p1 = p2(1:(length+1)/2);
p1(2:end-1) = 2*p1(2:end-1);
f = Fs*(0:(length/2))/length;
subplot(1,2,1)
plot(f,abs(p1))
title('Single-Sided Amplitude Spectrum')
xlabel('f (Hz)')
ylabel('|X(f)|')
```

```
subplot(1,2,2)
plot(f,(angle(p1)))
title('Phase Spectrum')
xlabel('f (Hz)')
ylabel('phase of X(f)')
```

Graph:



Problem 4:

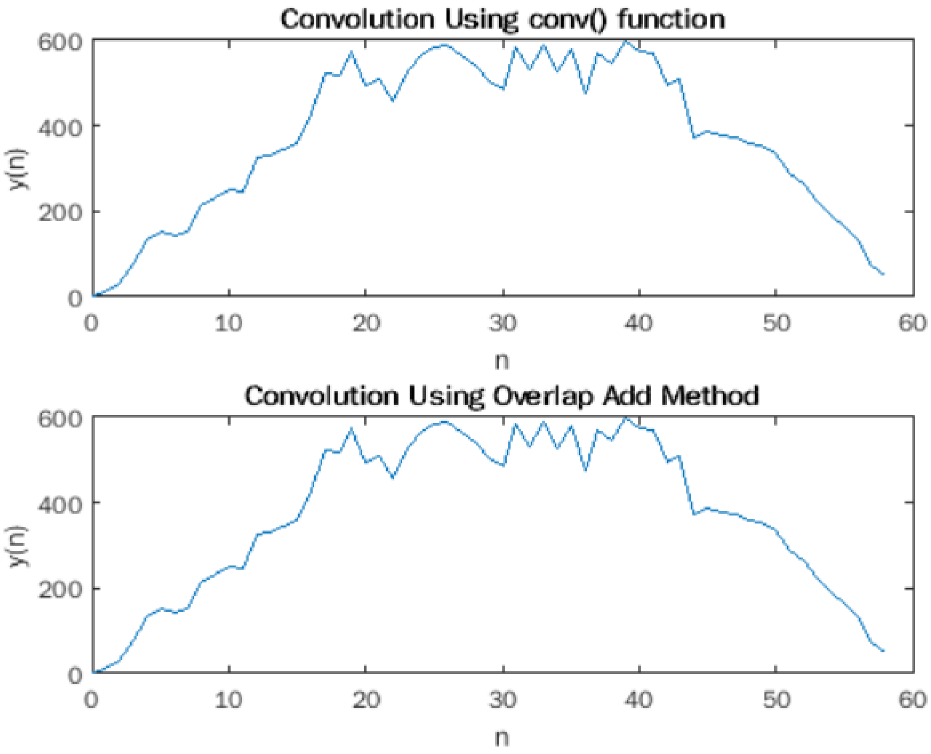
4) Write a program to implement linear convolution via DFT-based approach, and compare your results using direct linear convolution.

Code:

```
clear;
x=randi(10,[1,40]);
y=randi(10,[1,20]);
l=length(x)+length(y)-1;
x_fft=fft(x,l);
y_fft=fft(y,l);
coF=x_fft.*y_fft;
coT=conv(x,y);
coF2T=ifft(coF);
subplot(2,1,1)
plot(0:l-1,coT)
title('Convolution Using conv() function')
```

```
xlabel('n');
ylabel('y(n)');
subplot(2,1,2)
plot(0:l-1,coF2T)
title('Convolution Using Overlap Add Method')
xlabel('n');
ylabel('y(n)');
```

Graph:



Problem 5:

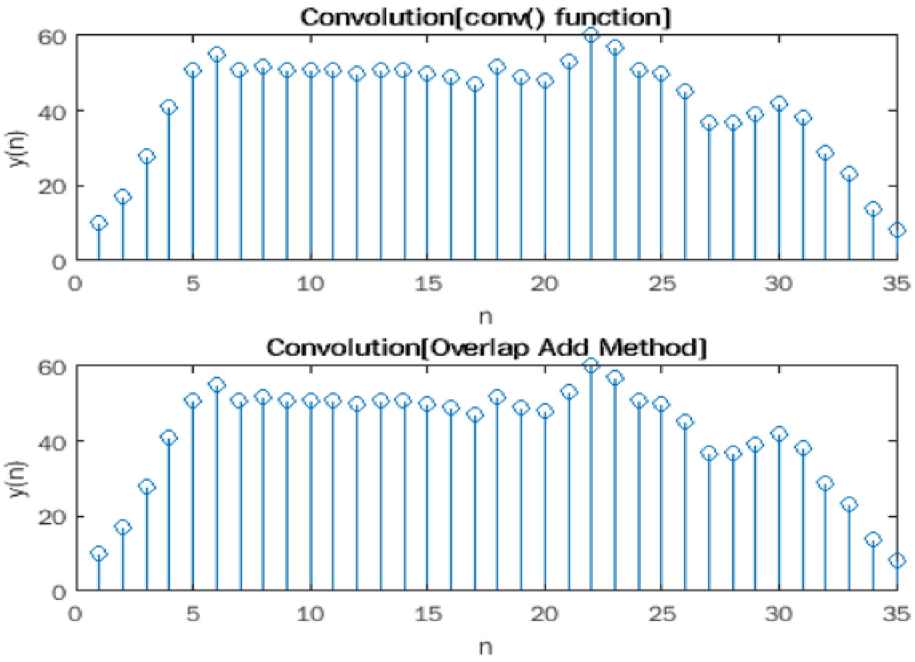
5) Write a program to realize linear convolution between a small and a long discrete-time sequence using overlap-add method. (Do not use 'fftfilter' function of MATLAB)

Code:

```
clear

x=randi(5,[1,30]);
h=randi(3,[1,6]);
len=3;
N1=length(x);
M=length(h);
LC=conv(x,h);
x=[x zeros(1,mod(-N1,len))];
N2=length(x);
h=[h zeros(1,len-1)];
H=fft(h,len+M-1);
S=N2/len;
```

```
i=1:len;
X=[zeros(M-1)];
for s=1:S
    xm=[x(i) zeros(1,M-1)];
    X1=fft(xm,len+M-1);
    Y=X1.*H;
    Y=ifft(Y);
    Z=X((length(X)-M+2):length(X))+Y(1:M-1);
    X=[X(1:(s-1)*len) Z Y(M:M+len-1)];
    i=s*len+1:(s+1)*len;
end
i=1:N1+M-1;
X=X(i);
similarity=corrcoef(X,LC);
figure()
subplot(2,1,1)
stem(LC);
title('Convolution[conv() function]')
xlabel('n');
ylabel('y(n)');
subplot(2,1,2)
stem(X);
title('Convolution[Overlap Add Method]')
xlabel('n');
ylabel('y(n)');
```

Graph:



Convolution[conv() function]

Convolution[Overlap Add Method]

Assignment 3

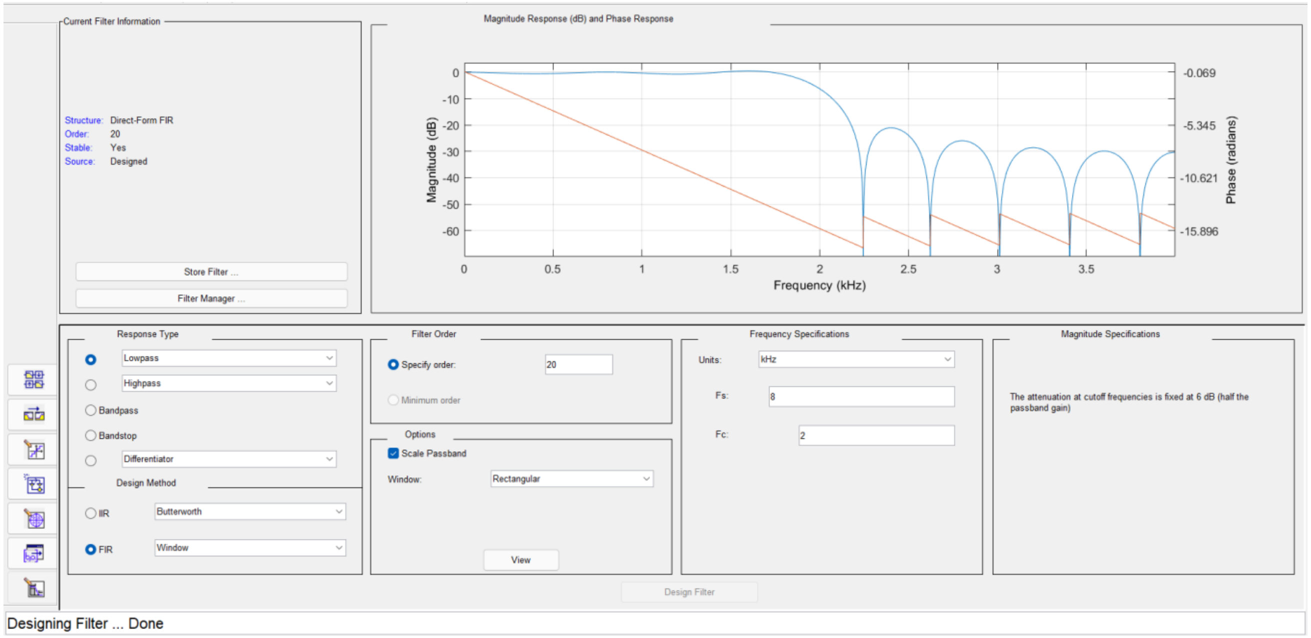
Problem 1:

1) Design a low-pass FIR filter of length 21 and 41 respectively with a cutoff frequency of 2 KHz using the following window functions. Assume the sampling frequency is 8 KHz.

   Window function:

   a. Rectangular window function
   b. Hamming window function
   c. Hanning window function
   d. Blackman window function

Designs:



Current Filter Information

Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

☑ Scale Passband
Window: Rectangular

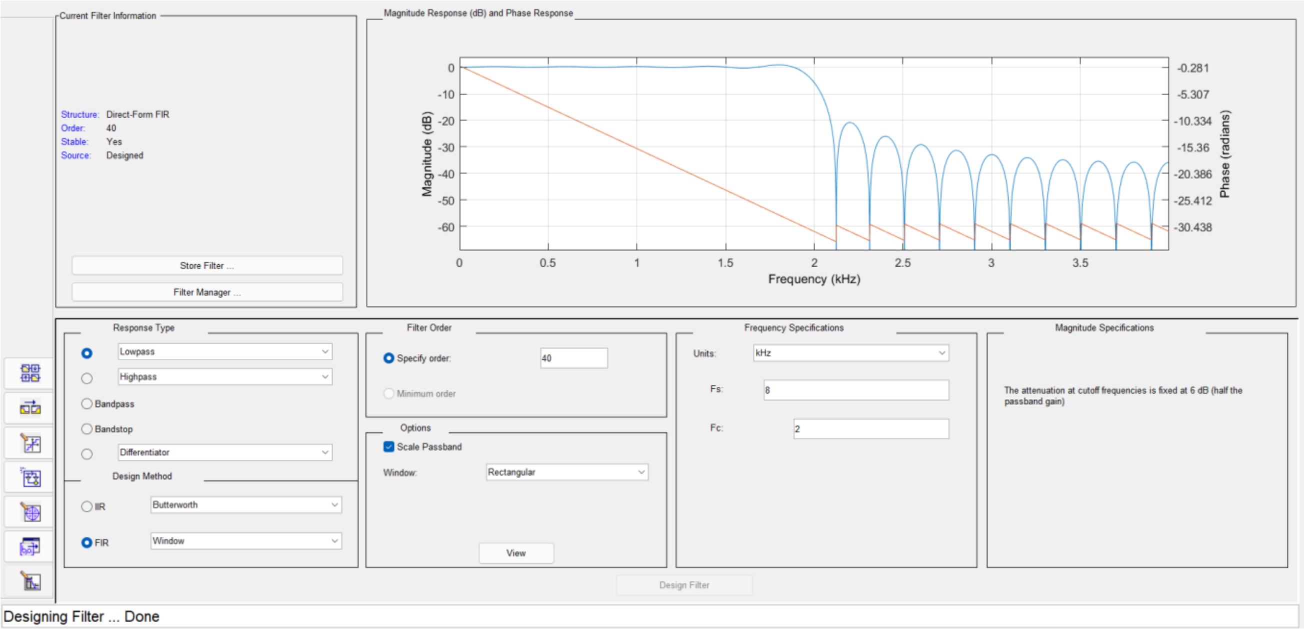
Frequency Specifications

Units: kHz
Fs: 8
Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 40
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Rectangular

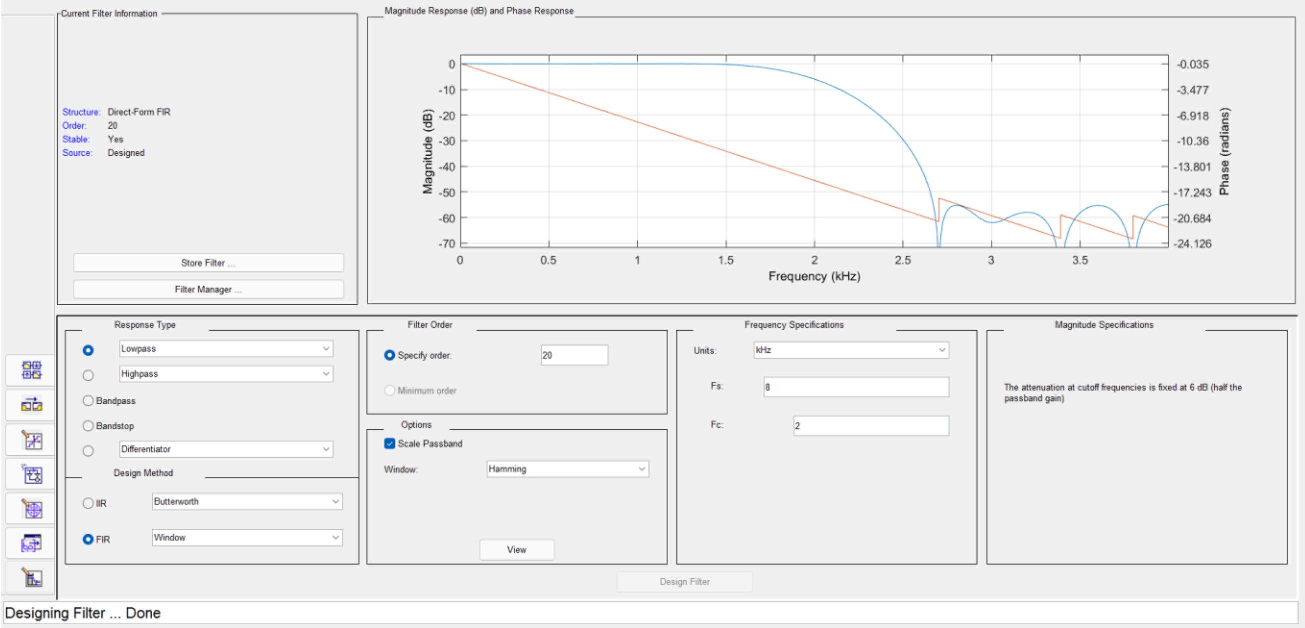
Frequency Specifications

Units: kHz
Fs: 8
Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

☑ Scale Passband
Window: Hamming

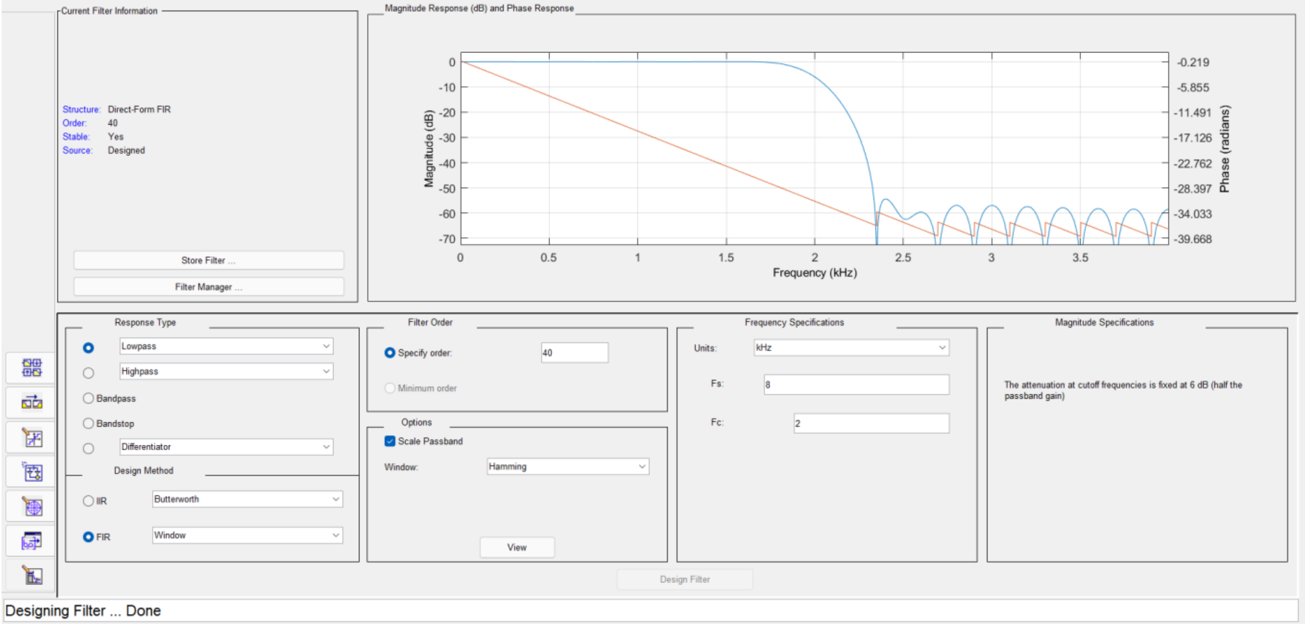
Frequency Specifications

Units: kHz
Fs: 8
Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 40
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Hamming

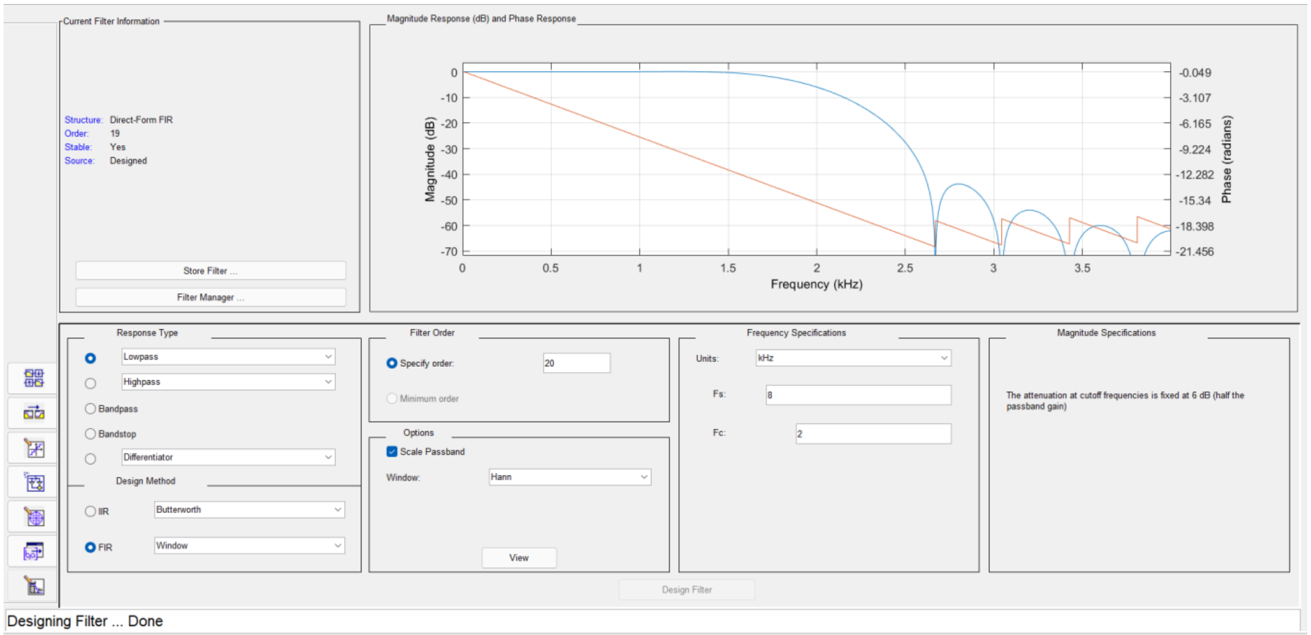
Frequency Specifications

Units: kHz
Fs: 8
Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 19
Stable: Yes
Source: Designed

Response Type

- ● Lowpass
- ○ Highpass
- ○ Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method

- ○ IIR Butterworth
- ● FIR Window

Filter Order

- ● Specify order: 20
- ○ Minimum order

Options

☑ Scale Passband

Window: Hann

Frequency Specifications

Units: kHz

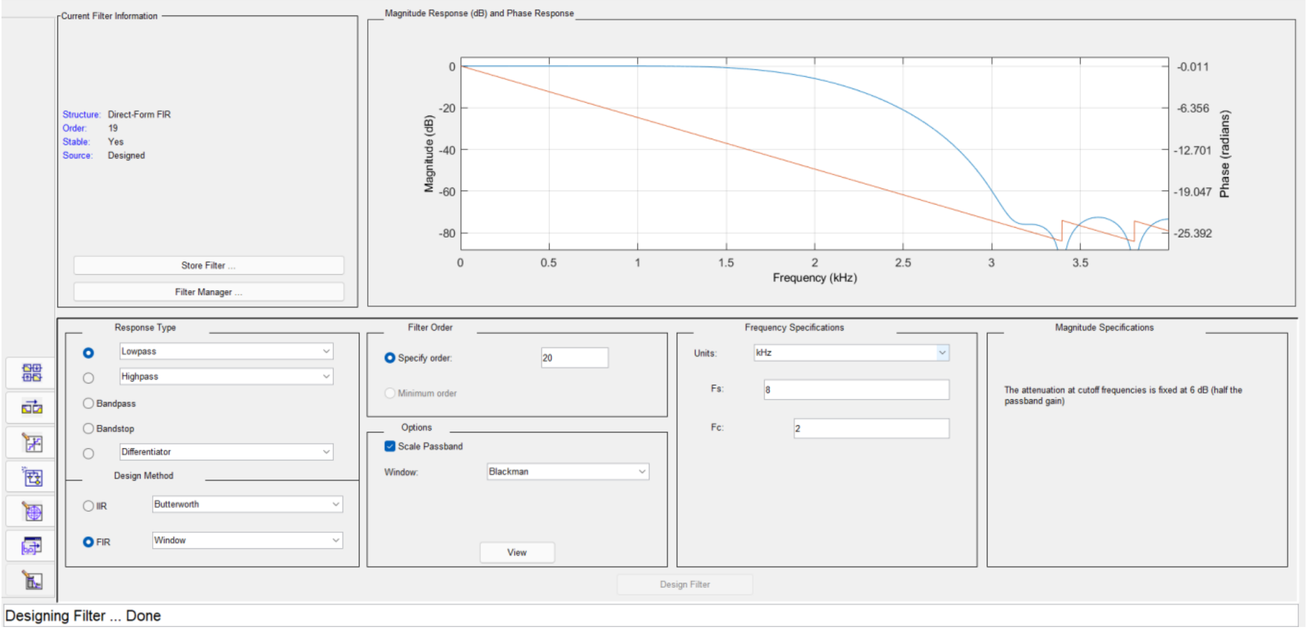
Fs: 8

Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 19
Stable: Yes
Source: Designed

Response Type

- ● Lowpass
- ○ Highpass
- ○ Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method

- ○ IIR Butterworth
- ● FIR Window

Filter Order

- ● Specify order: 20
- ○ Minimum order

Options

☑ Scale Passband

Window: Blackman

Frequency Specifications

Units: kHz

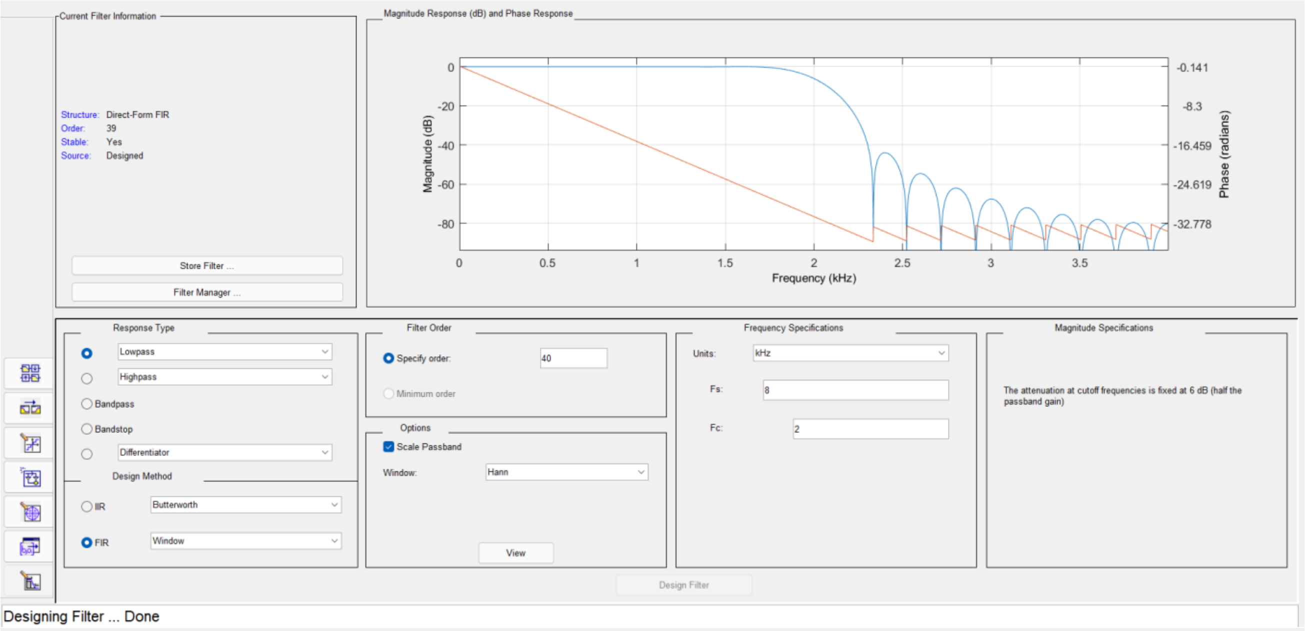
Fs: 8

Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 39
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Hann

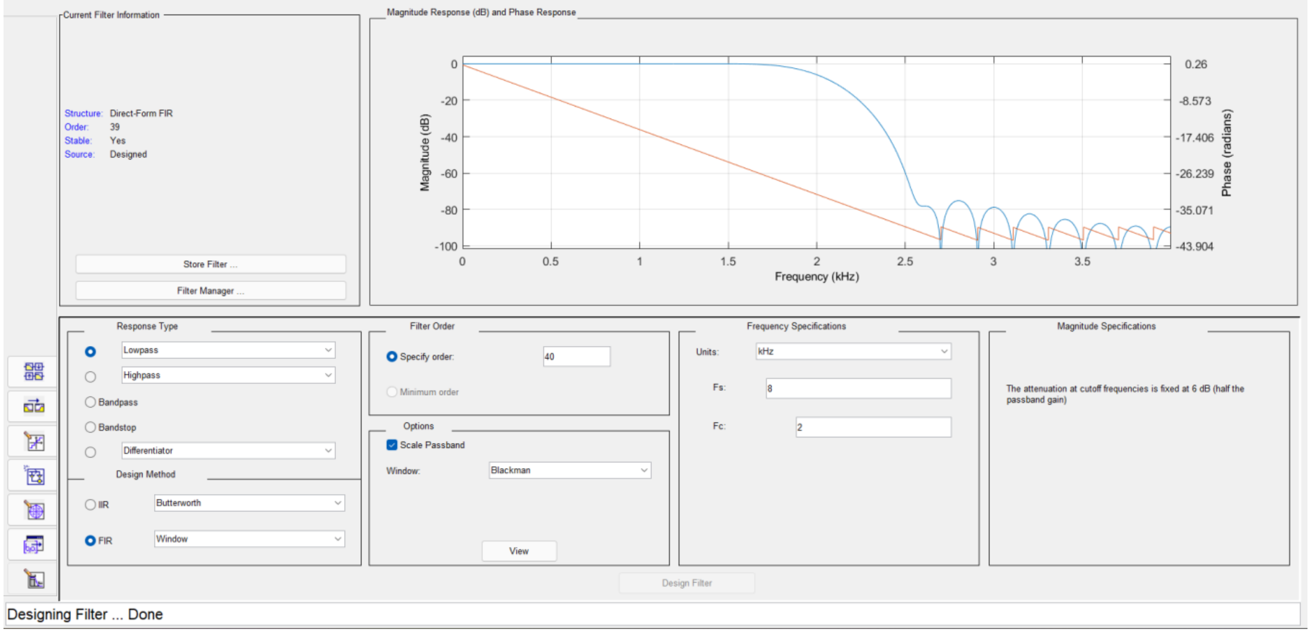
Frequency Specifications

Units: kHz
Fs: 8
Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 39
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Blackman

Frequency Specifications

Units: kHz
Fs: 8
Fc: 2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done

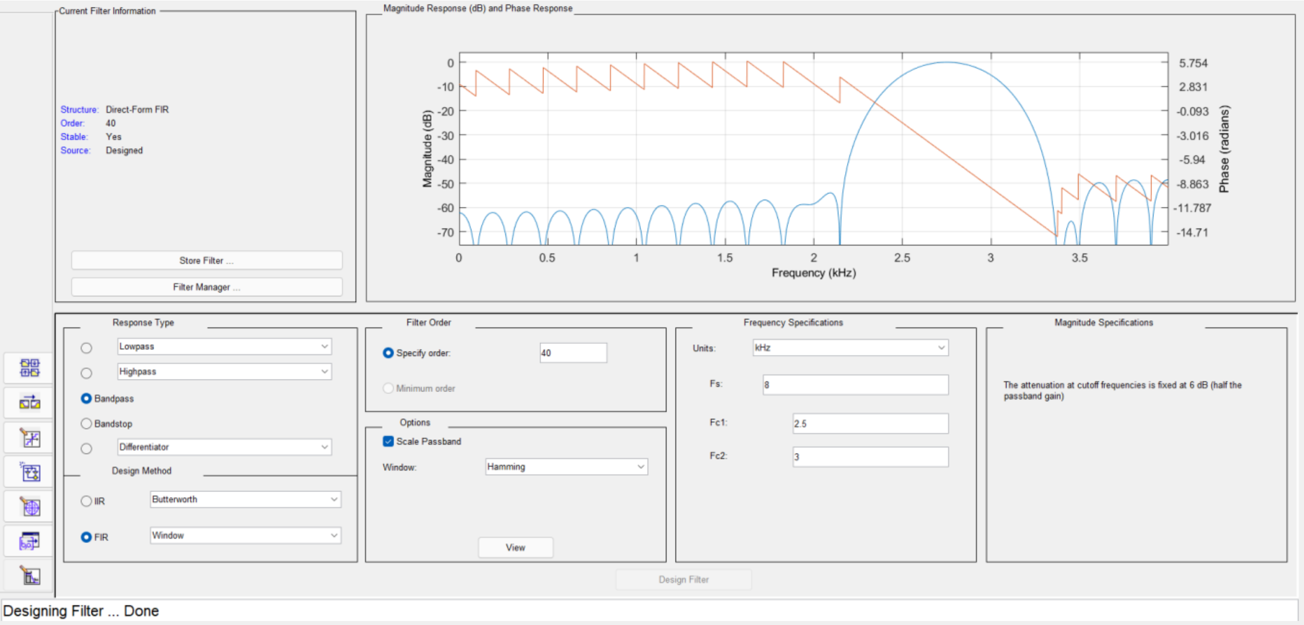
Problem 2:

2) Design 21-length and 41-length band-pass FIR filter with lower and upper cutoff frequency at 2.5 KHz and 3 KHz respectively using the following window functions. Assume the sampling frequency is 8 KHz.

Window function:

a. Rectangular window function
b. Hamming window function
c. Hanning window function
d. Blackman window function

Designs:



Current Filter Information

Structure: Direct-Form FIR
Order: 40
Stable: Yes
Source: Designed

Response Type: ● Bandpass
Design Method: ● FIR Window

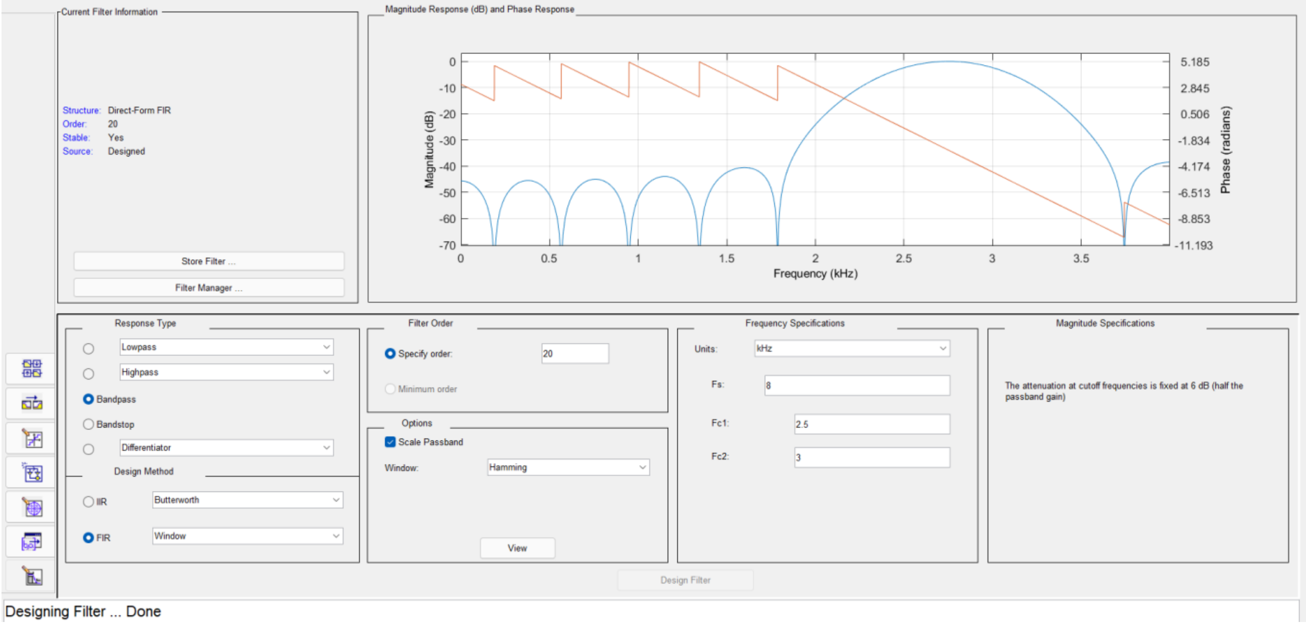
Filter Order: ● Specify order: 40
○ Minimum order

Options: ☑ Scale Passband
Window: Hamming

Frequency Specifications
Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications
The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed

Response Type: ● Bandpass
Design Method: ● FIR Window

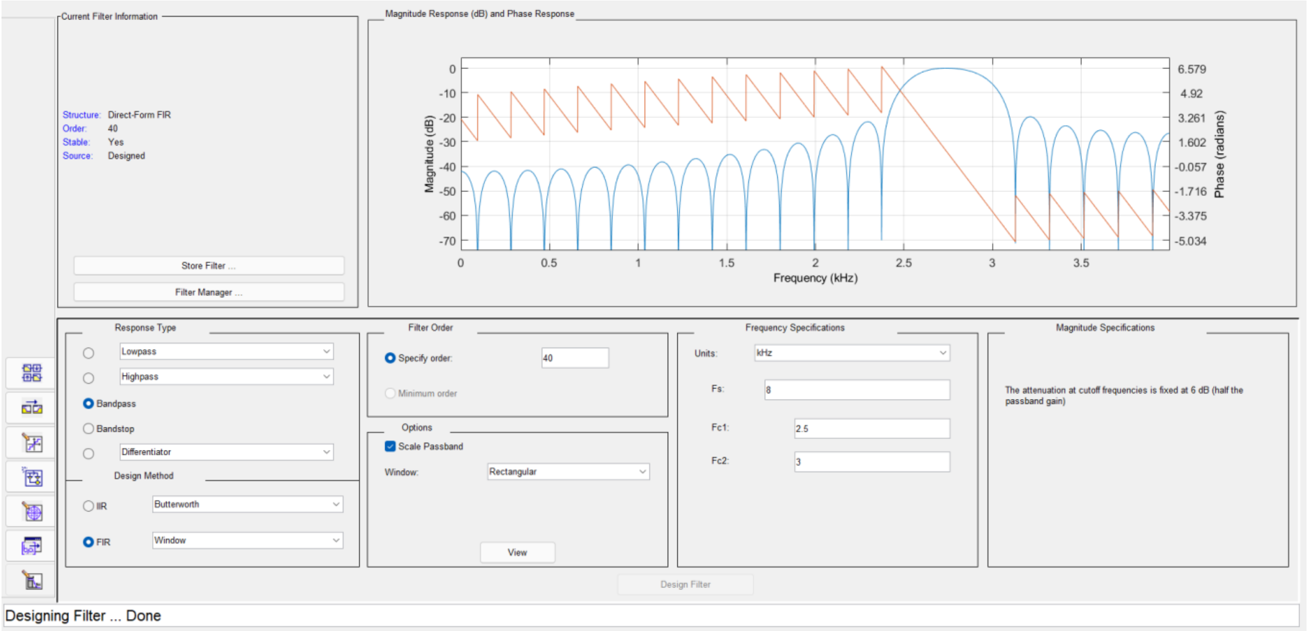
Filter Order: ● Specify order: 20
○ Minimum order

Options: ☑ Scale Passband
Window: Hamming

Frequency Specifications
Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications
The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 40
Stable: Yes
Source: Designed

Response Type

○ Lowpass
○ Highpass
● Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Rectangular

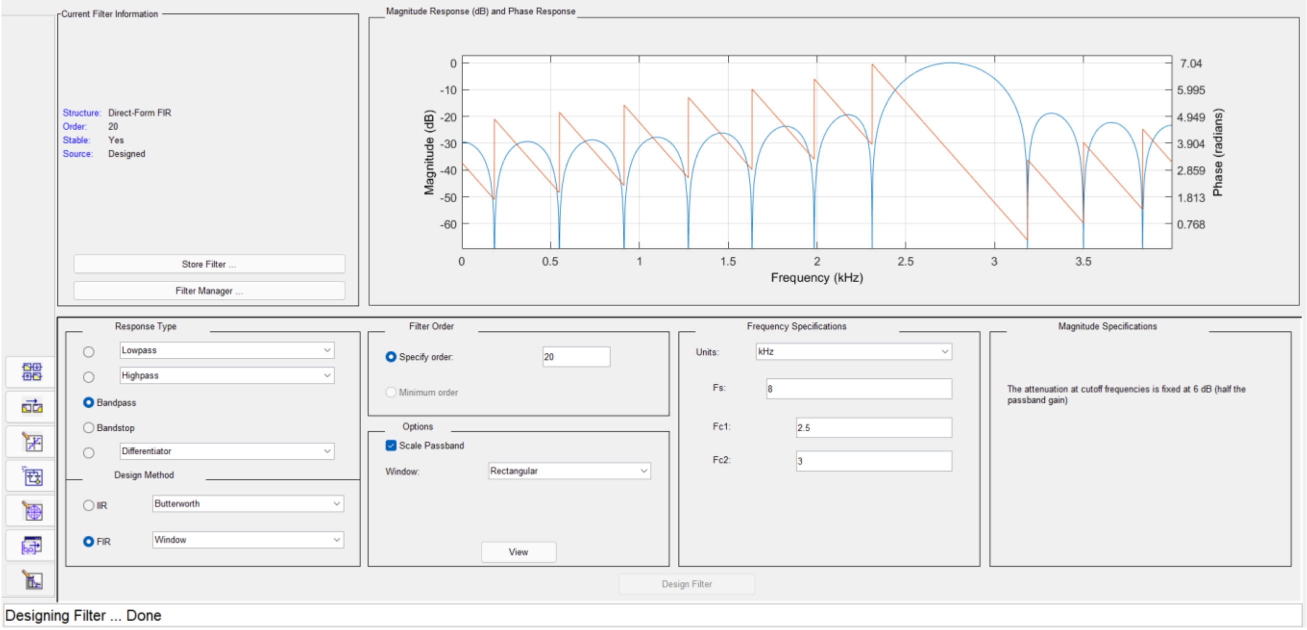
Frequency Specifications

Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed

Response Type

○ Lowpass
○ Highpass
● Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

☑ Scale Passband
Window: Rectangular

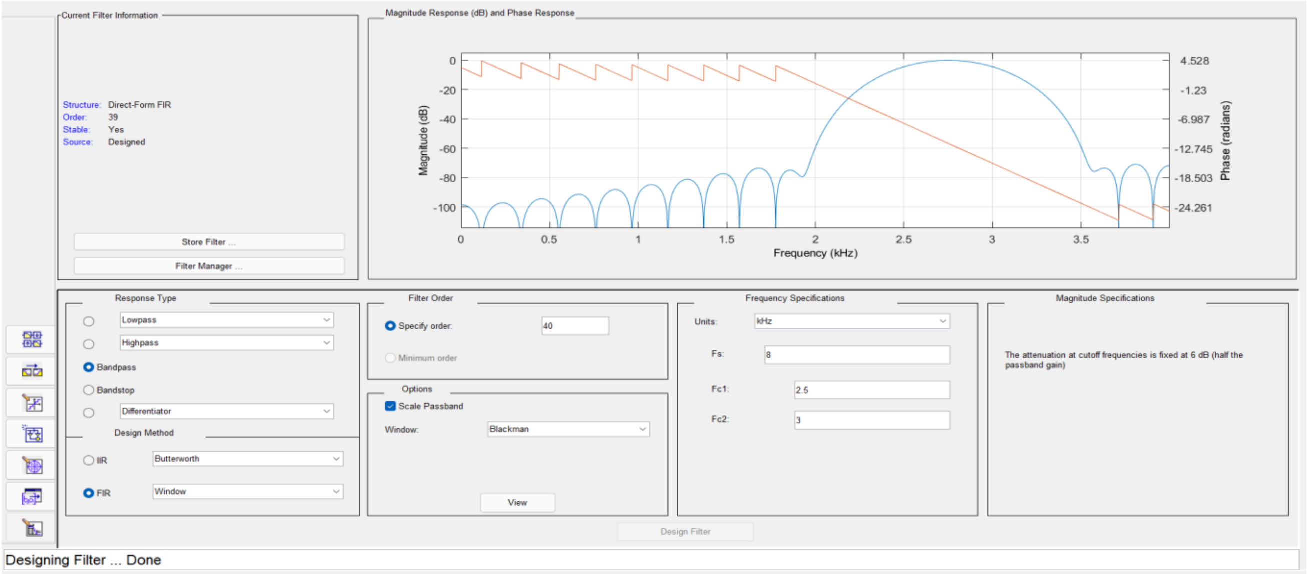
Frequency Specifications

Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 39
Stable: Yes
Source: Designed

Response Type

○ Lowpass
○ Highpass
● Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Blackman

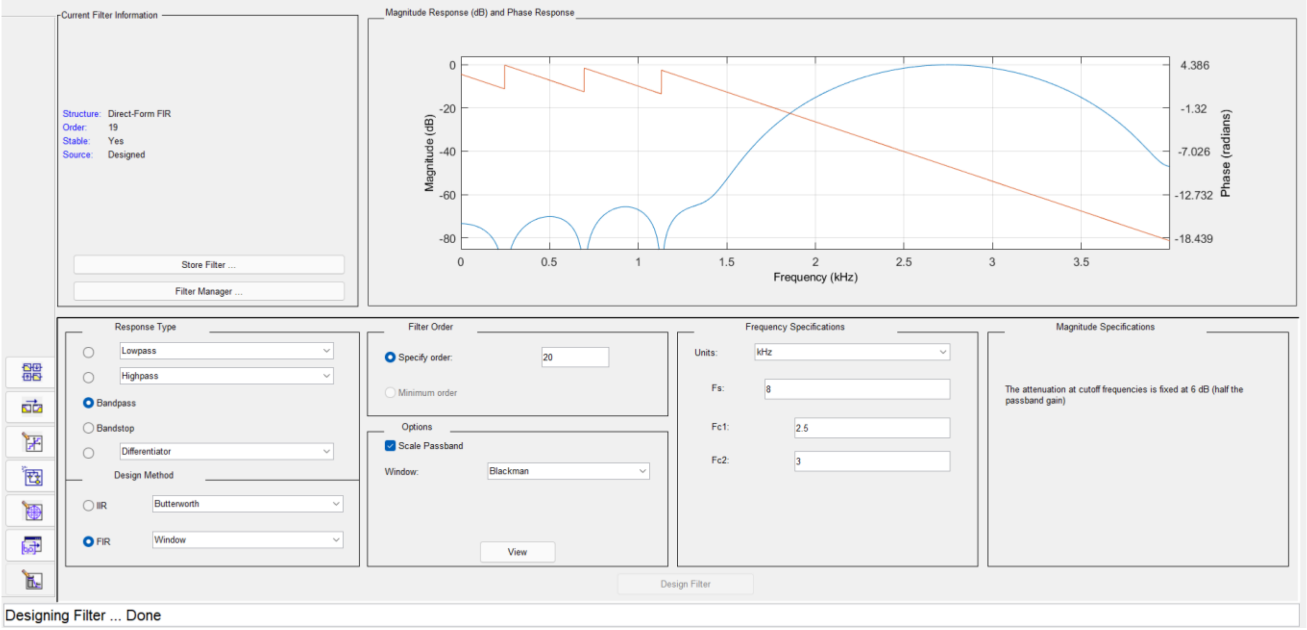
Frequency Specifications

Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 19
Stable: Yes
Source: Designed

Response Type

○ Lowpass
○ Highpass
● Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

☑ Scale Passband
Window: Blackman

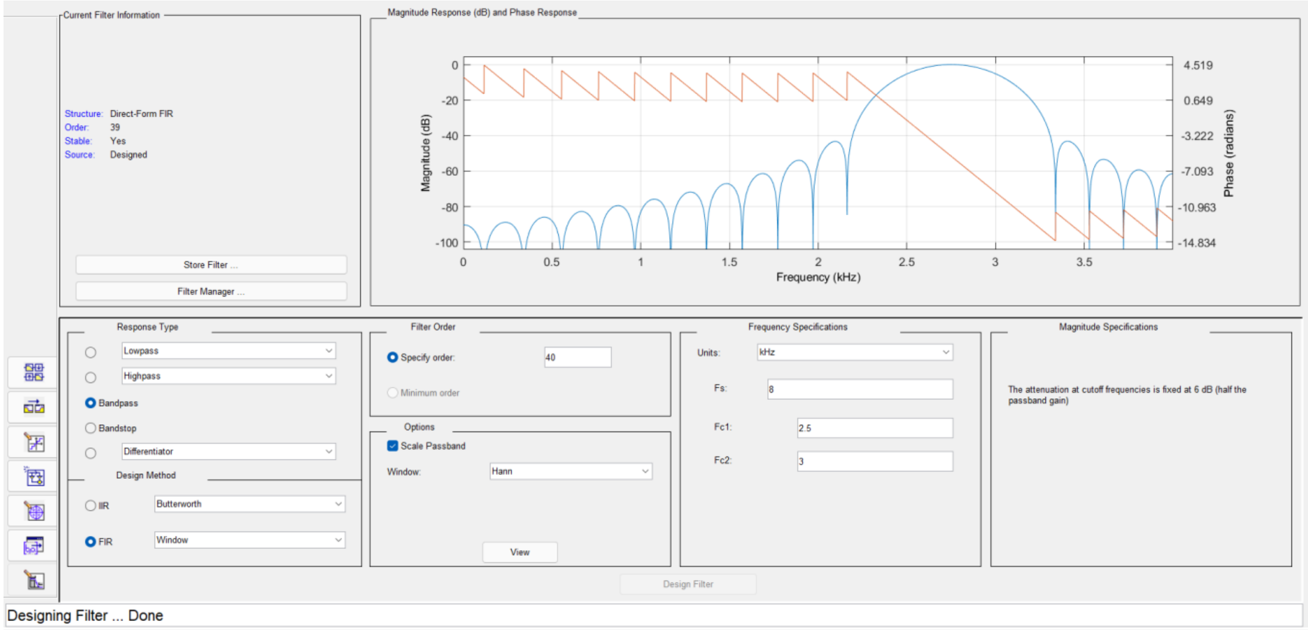
Frequency Specifications

Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 39
Stable: Yes
Source: Designed

Response Type

- ○ Lowpass
- ○ Highpass
- ● Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method

- ○ IIR Butterworth
- ● FIR Window

Filter Order

- ● Specify order: 40
- ○ Minimum order

Options

☑ Scale Passband

Window: Hann

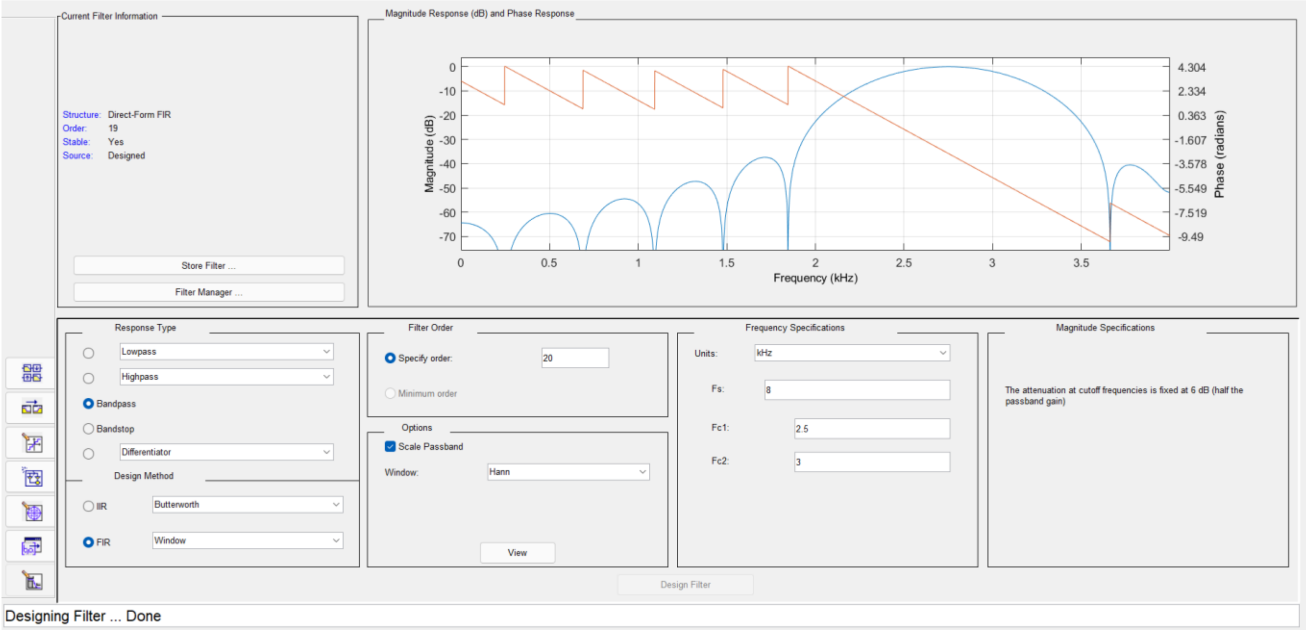
Frequency Specifications

Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 19
Stable: Yes
Source: Designed

Response Type

- ○ Lowpass
- ○ Highpass
- ● Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method

- ○ IIR Butterworth
- ● FIR Window

Filter Order

- ● Specify order: 20
- ○ Minimum order

Options

☑ Scale Passband

Window: Hann

Frequency Specifications

Units: kHz
Fs: 8
Fc1: 2.5
Fc2: 3

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

Designing Filter ... Done

Problem 3:

3) Use the frequency sampling method to design a linear phase low-pass FIR filter of length 21 and 41 respectively. Let the cutoff frequency be 2 KHz and assume a sampling frequency of 8KHz. List FIR filter coefficients and plot the frequency responses.

Code:

LENGTH=21

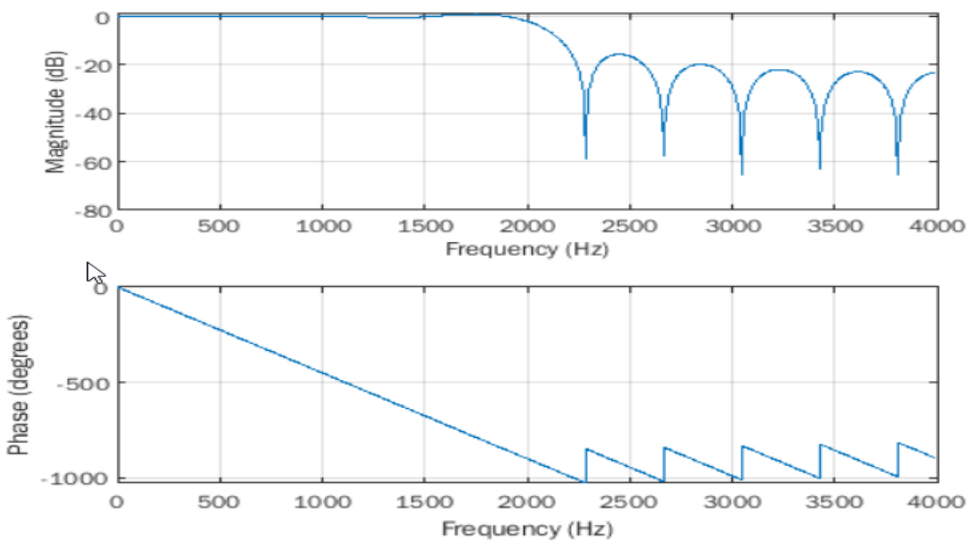
```
% Lowpass filter
N=21;
% Sampling frequency
Fs=8000;
% lower cutoff frequency
Fc=2000;
% normalized frequency
w = 2*pi*Fc/Fs;
M=(N-1)/2;
wk=(1:M)*2*pi/N;
Hk = wk<=w;
n=-M:M;
for index=0:M
h(index+1) =
1/N*(1+2*sum(Hk.*cos(wk*index)));
end
g = h(2:M+1);
h= [flip(g) h];
figure()
freqz(h,1,1024,Fs)
```

LENGTH=41

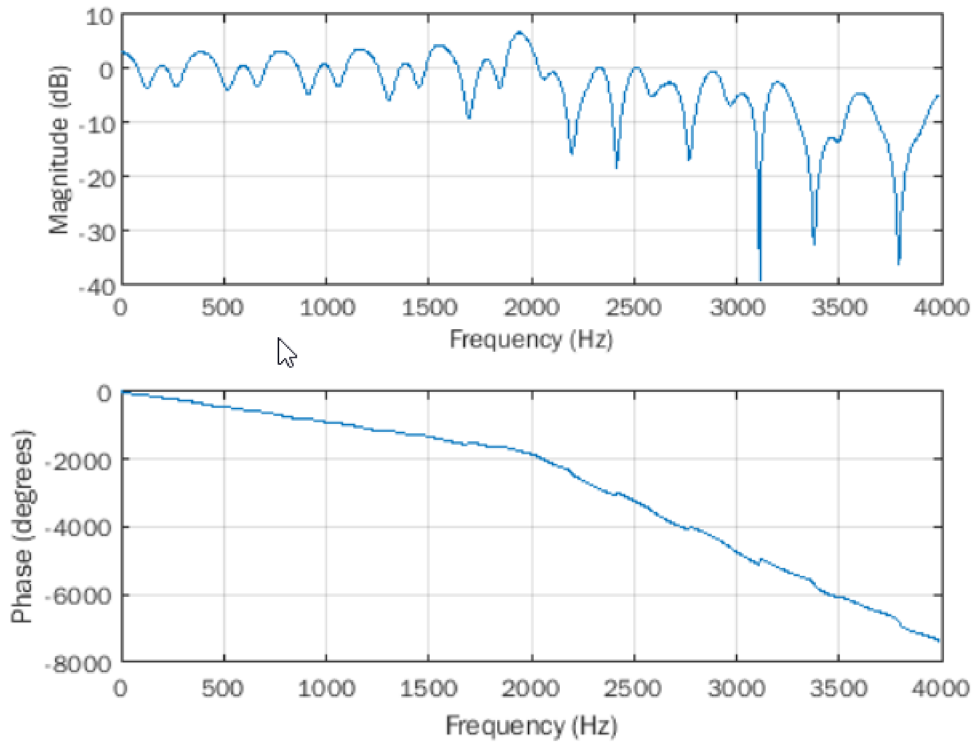
```
% Lowpass filter
N=41;
% Sampling frequency
Fs=8000;
% lower cutoff frequency
Fc=2000;
% normalized frequency
w = 2*pi*Fc/Fs;
M=(N-1)/2;
wk=(1:M)*2*pi/N;
Hk = wk<=w;
n=-M:M;
for index=0:M
h(index+1) =
1/N*(1+2*sum(Hk.*cos(wk*index)));
end
g = h(2:M+1);
h= [flip(g) h];
figure()
freqz(h,1,1024,Fs)
```

Graph:

LENGTH=21



LENGTH=41



Problem 4:

4) Use the frequency sampling method to design a linear phase band-pass FIR filter of length 21 and 41 respectively. Let the upper and lower cutoff frequency be 2.5 KHz and 3 KHz respectively and assume a sampling frequency of 8KHz. List FIR filter coefficients and plot the frequency responses.

Code:

LENGTH=21

```
% Lowpass filter Order
N=21;
% Sampling frequency]
Fs=8000;
Fl=2500;
Fh=3000;
f1 = 2*pi*Fl/Fs;
f2 = 2*pi*Fh/Fs;
M=(N-1)/2;
wk=(1:M)*2*pi/N;
Hk = f1<=wk & wk<=f2;
n=-M:M;
h=[];
for index=0:N-1
h(index+1) =
1/N*(2*sum(Hk.*cos(wk*(M-
index))));
end
figure()
freqz(h,1,1024,Fs)
```

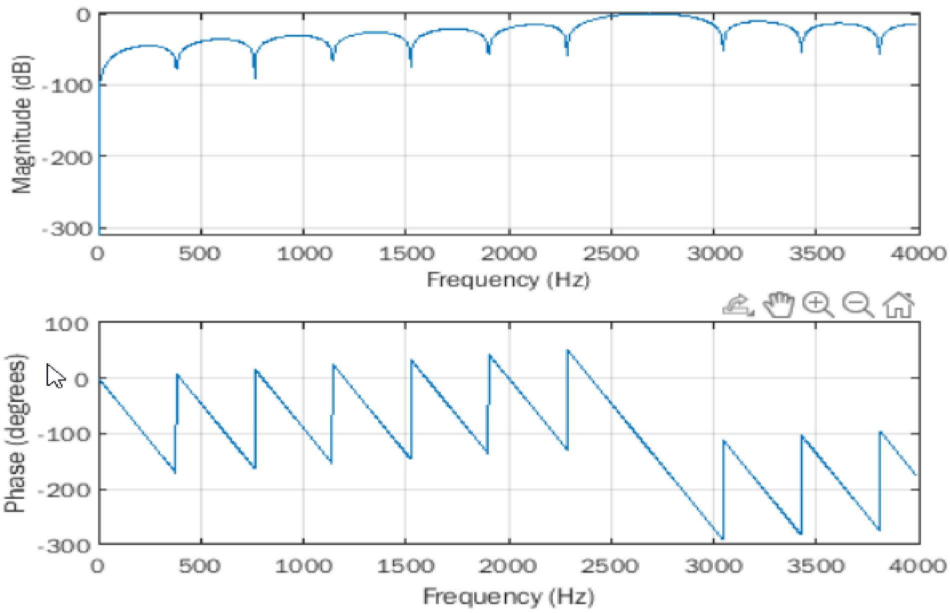
LENGTH=41

```
% Lowpass filter Order
N=41;
% Sampling frequency]
Fs=8000;
Fl=2500;
Fh=3000;
f1 = 2*pi*Fl/Fs;
f2 = 2*pi*Fh/Fs;
M=(N-1)/2;
wk=(1:M)*2*pi/N;
Hk = f1<=wk & wk<=f2;
n=-M:M;
h=[];
for index=0:N-1
h(index+1) =
1/N*(2*sum(Hk.*cos(wk*(M-
index))));
end
```

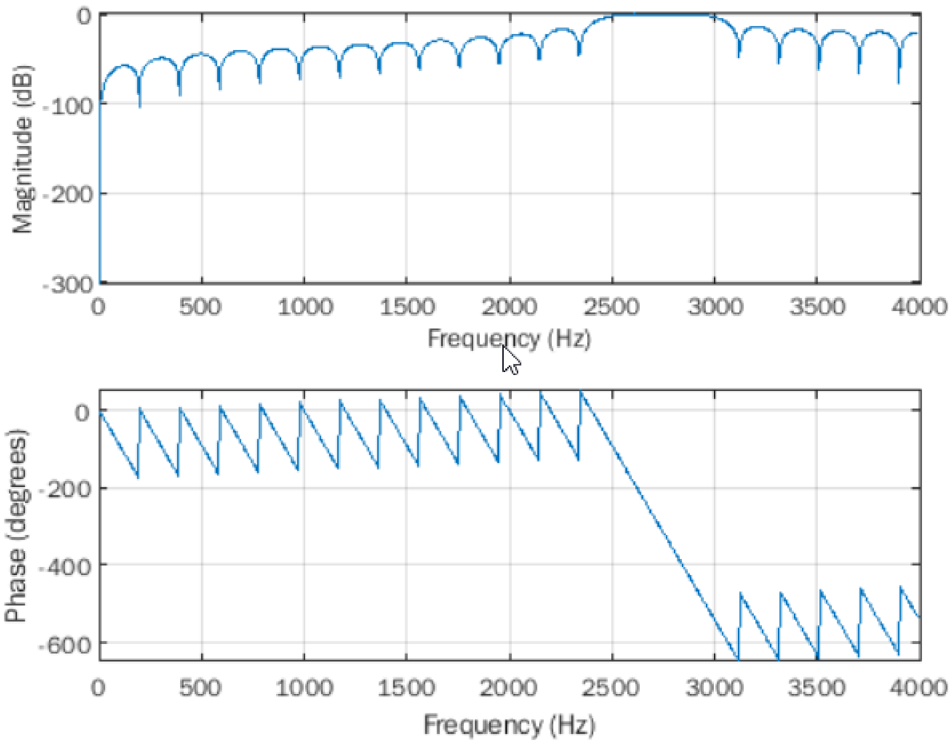
```
figure()                                       freqz(h,1,1024,Fs)
```

Graph:

LENGTH=21

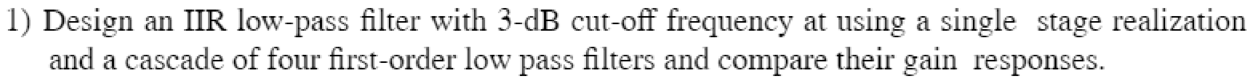


LENGTH=41



Assignment 4

Problem 1:

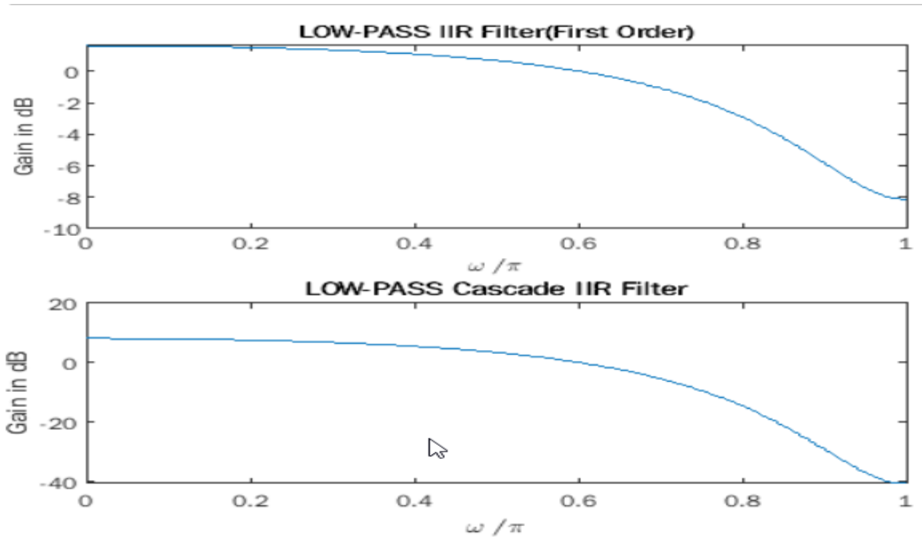


1) Design an IIR low-pass filter with 3-dB cut-off frequency at using a single stage realization and a cascade of four first-order low pass filters and compare their gain responses.

Code:

```
clc;
clear all;
w = 0:pi/255:pi;

n1 = [1 0.45*pi]/1.5267;
d1 = [1 0.31 ];
num = n1;
den = d1;
h1 = freqz(num, den, w);
g1 = 20*log10(abs(h1));

subplot(2,1,1);
plot(w/pi,g1);
xlabel("\omega /\pi");
ylabel("Gain in dB");
title("LOW-PASS IIR Filter(First Order)");
for index=1:4
num = conv (num,n1);
den = conv (den,d1);
end
h1 = freqz(num, den, w);
g1 = 20*log10(abs(h1));

subplot(2,1,2);
plot(w/pi,g1);
xlabel("\omega /\pi");ylabel("Gain in dB");
title("LOW-PASS Cascade IIR Filter");
```

Graph:



Problem 2:

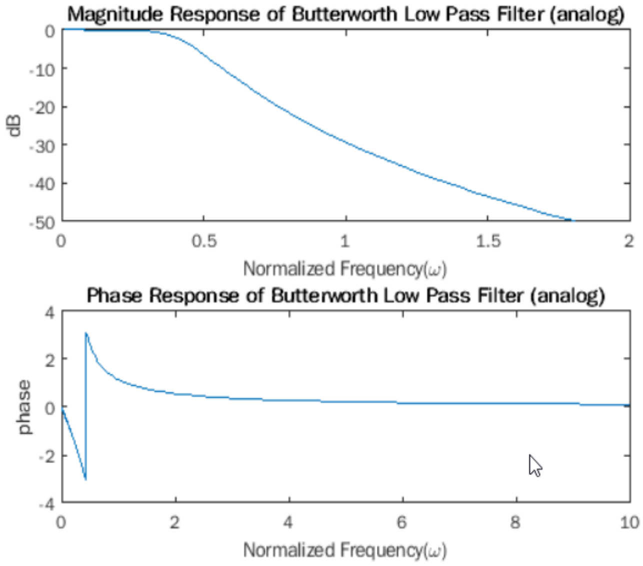
2) Write a MATLAB program to design a digital Butterworth low-pass filter using impulse invariance method. Determine the order of the analog prototype filter for this purpose. The input data required for your program are sampling frequency (F_S), pass-band edge frequency (F_p), stop-band edge frequency (F_s), maximum pass-band ripple (δ_p) and minimum stop band attenuation (δ_s). Plot the gain response of the designed filter for the flowing inputs: , , , , . You may use the M-file of MATLAB.

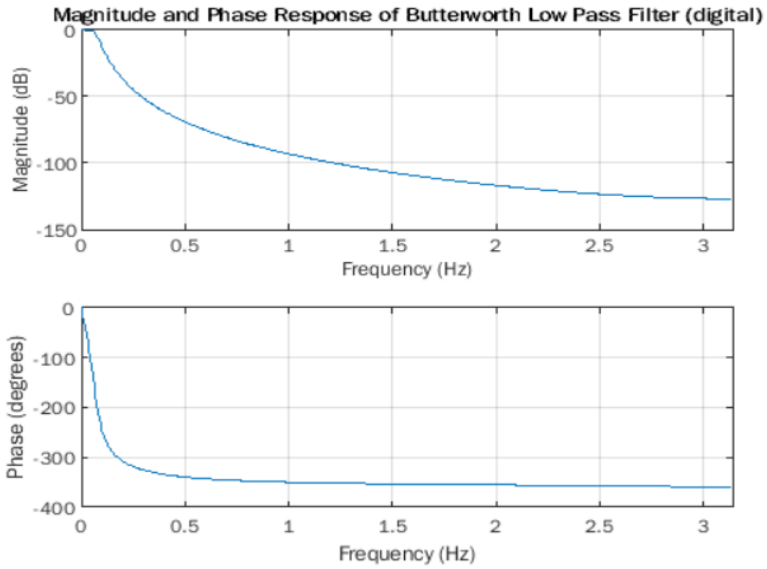
Code:

```
clear all;
close all;

fs=80000;
F_P=4000;
F_S=20000;

Rp=0.5;
Rs=45;

Wp=2*pi*F_P/fs;
Ws=2*pi*F_S/fs;

[N,Wc]=buttord(Wp,Ws,Rp,Rs,'s');
[B,A]=butter(N,Wc,"low",'s');
[h,w] = freqs(B,A);
dB = mag2db(abs(h));

subplot(2,1,1)
plot(w,dB)
title('Magnitude Response of Butterworth Low Pass Filter (analog)')
```

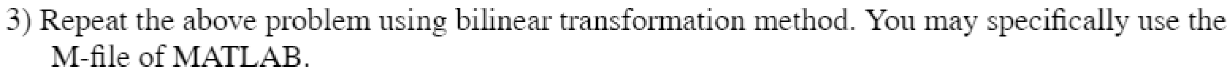
```
xlabel('Normalized Frequency(\omega)')
ylabel('dB')
ylim([-50 0])

subplot(2,1,2)
plot(w,angle(h))
title('Phase Response of Butterworth Low Pass Filter (analog)')
xlabel('Normalized Frequency(\omega)')
ylabel('phase')


[num,den]=impinvar(B,A,2*pi);
figure
freqz(num,den,[],2*pi)
title('Magnitude and Phase Response of Butterworth Low Pass Filter (digital)')
```

Graph:





Problem 3:

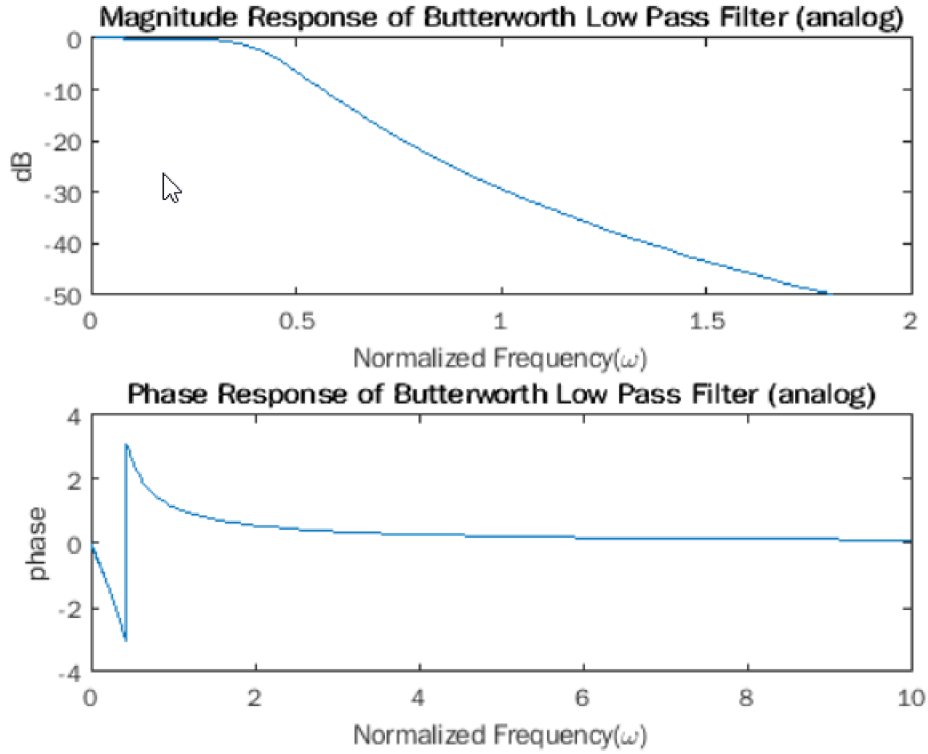


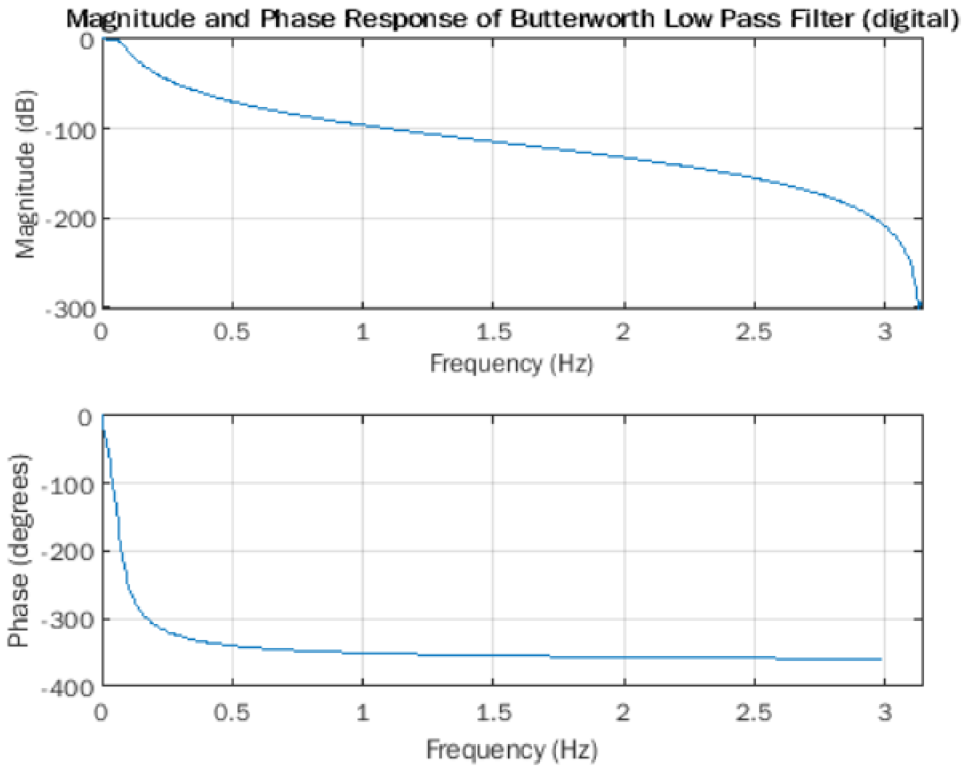
3) Repeat the above problem using bilinear transformation method. You may specifically use the M-file of MATLAB.

Code:

```
clear all;
close all;

fs=80000;
F_P=4000;
F_S=20000;

Rp=0.5;
Rs=45;

Wp=2*pi*F_P/fs;
Ws=2*pi*F_S/fs;

[N,Wc]=buttord(Wp,Ws,Rp,Rs,'s');
[B,A]=butter(N,Wc,"low",'s');
[h,w] = freqs(B,A);
dB = mag2db(abs(h));

subplot(2,1,1)
plot(w,dB)
title('Magnitude Response of Butterworth Low Pass Filter (analog)')
xlabel('Normalized Frequency(\omega)')
ylabel('dB')
ylim([-50 0])
subplot(2,1,2)
plot(w,angle(h))
title('Phase Response of Butterworth Low Pass Filter (analog)')
xlabel('Normalized Frequency(\omega)')
ylabel('phase')


[num,den]=bilinear(B,A,2*pi);
figure
freqz(num,den,[],2*pi)
title('Magnitude and Phase Response of Butterworth Low Pass Filter (digital)')
```

Graph:


Magnitude Response of Butterworth Low Pass Filter (analog)
dB
0
-10
-20
-30
-40
-50
0
0.5
1
1.5
2
Normalized Frequency(ω)
Phase Response of Butterworth Low Pass Filter (analog)
phase
4
2
0
-2
-4
0
2
4
6
8
10
Normalized Frequency(ω)


Magnitude and Phase Response of Butterworth Low Pass Filter (digital)
Magnitude (dB)
0
-100
-200
-300
0
0.5
1
1.5
2
2.5
3
Frequency (Hz)
Phase (degrees)
0
-100
-200
-300
-400
0
0.5
1
1.5
2
2.5
3
Frequency (Hz)

Assignment 5

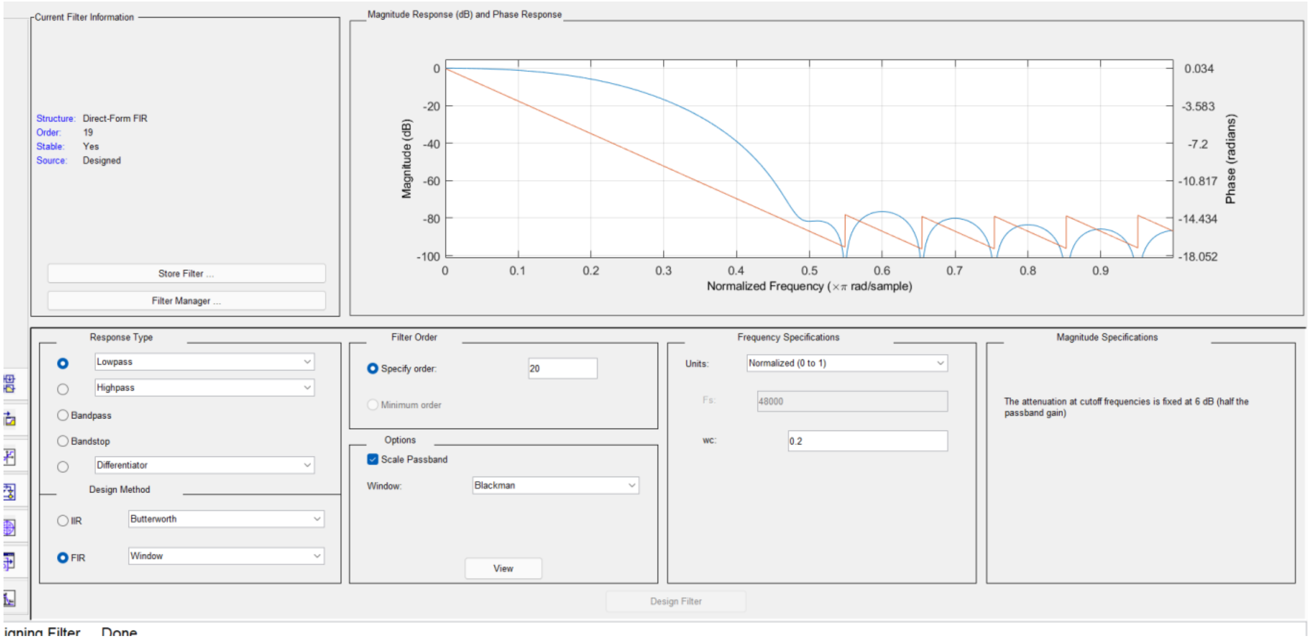
Problem 1:

1) Design a low-pass FIR filter of length 21 and 41 respectively using MATLAB command ‘fdatool’ with a normalized cutoff frequency of 0.2 rad/pi for the following window functions.

   Window function:
   (i) Rectangular window function
   (ii) Hamming window function
   (iii) Hanning window function
   (iv) Blackman window function

   Export the filter coefficients in MATLAB workspace as variables for Problem 3.

Designs:



Current Filter Information

Structure: Direct-Form FIR
Order: 19
Stable: Yes
Source: Designed

Response Type: ● Lowpass

Design Method: ● FIR Window

Filter Order: ● Specify order: 20

Options: ☑ Scale Passband

Window: Blackman

Frequency Specifications

Units: Normalized (0 to 1)

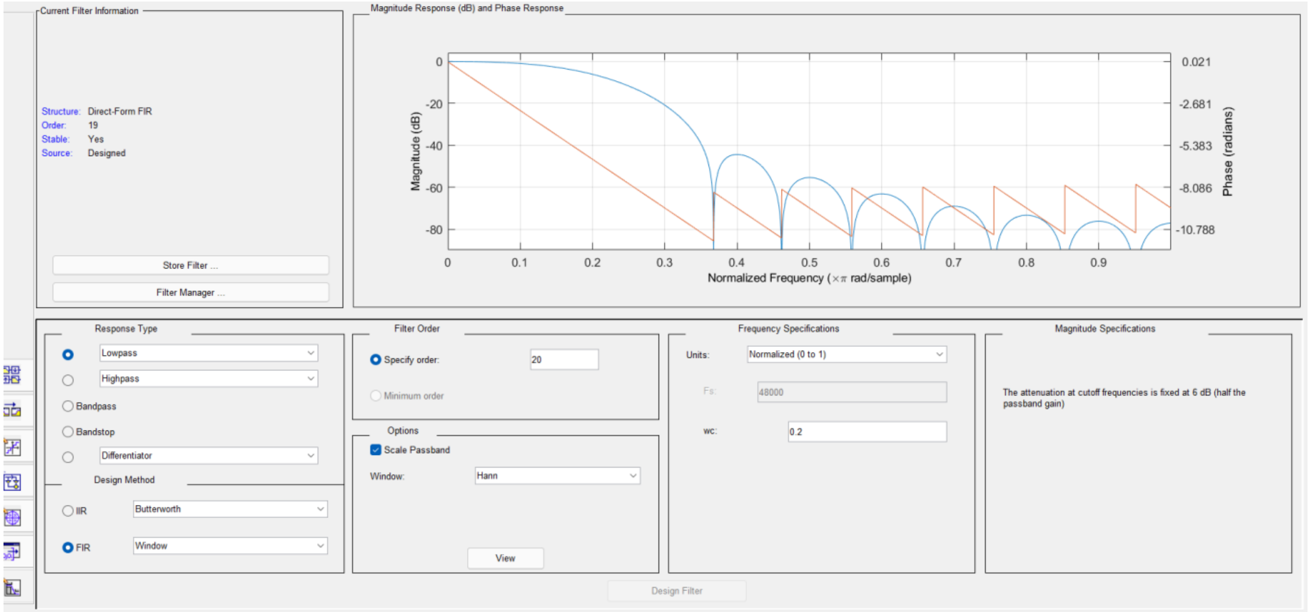
Fs: 48000

wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

ianing Filter Done



Current Filter Information

Structure: Direct-Form FIR
Order: 19
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

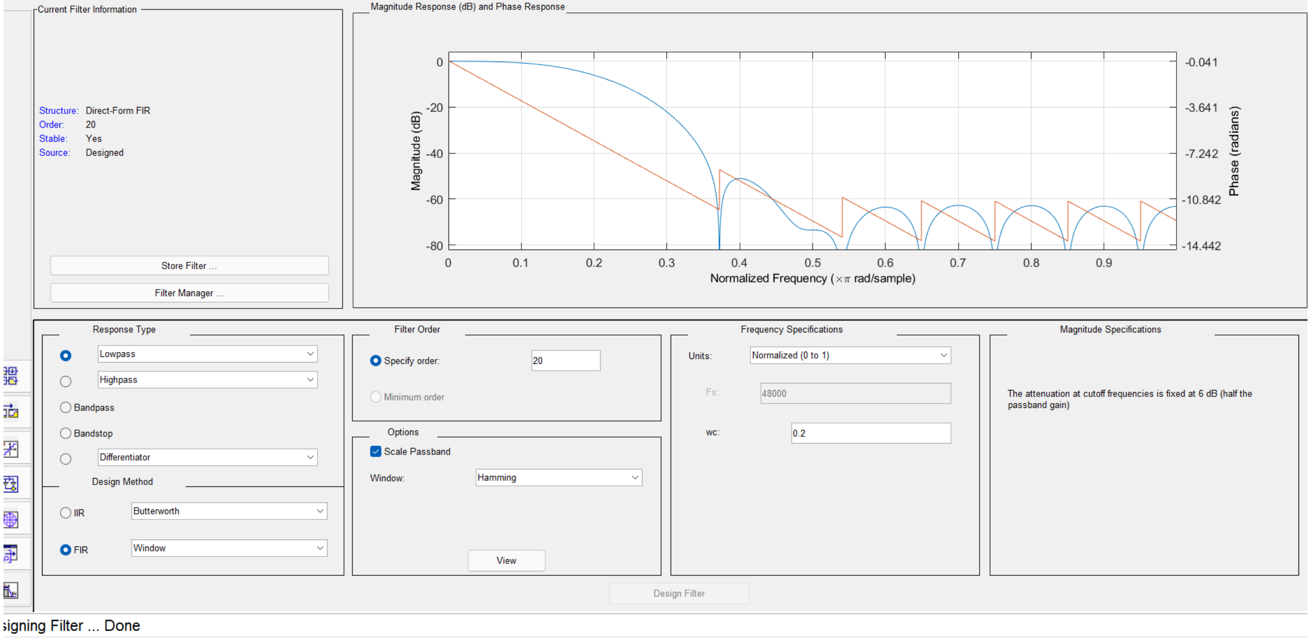
☑ Scale Passband
Window: Hann

Frequency Specifications

Units: Normalized (0 to 1)
Fs: 48000
wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)



Current Filter Information

Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

☑ Scale Passband
Window: Hamming

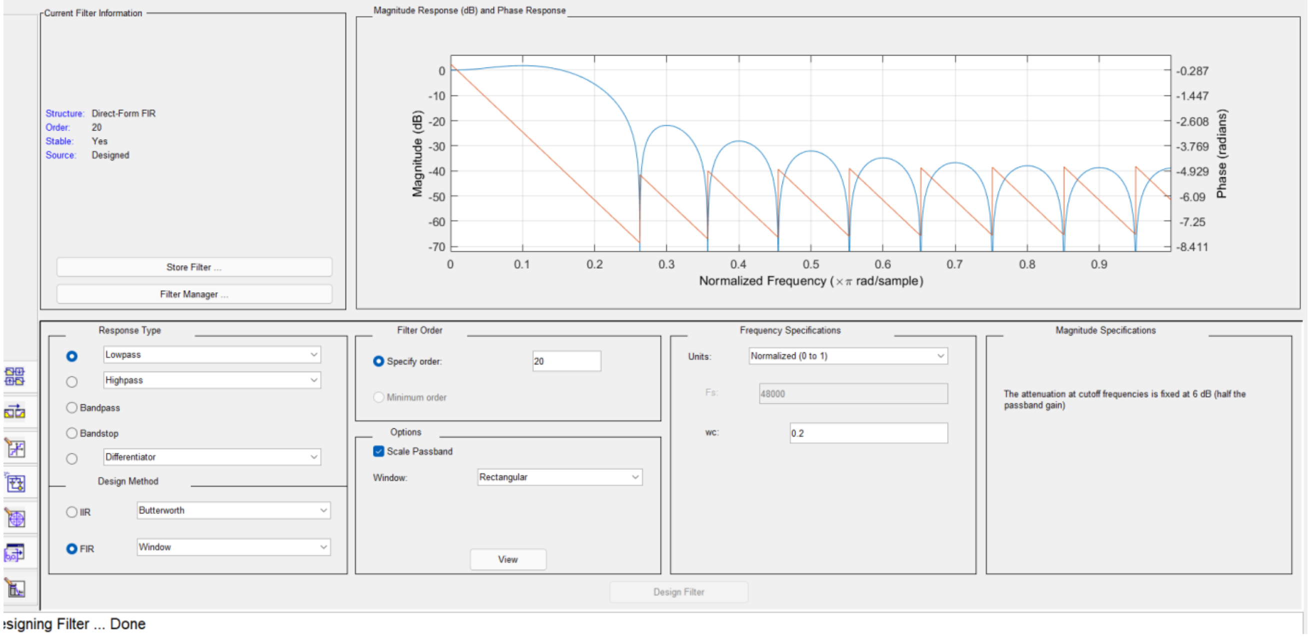
Frequency Specifications

Units: Normalized (0 to 1)
Fs: 48000
wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

signing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Butterworth
● FIR Window

Filter Order

● Specify order: 20
○ Minimum order

Options

☑ Scale Passband
Window: Rectangular

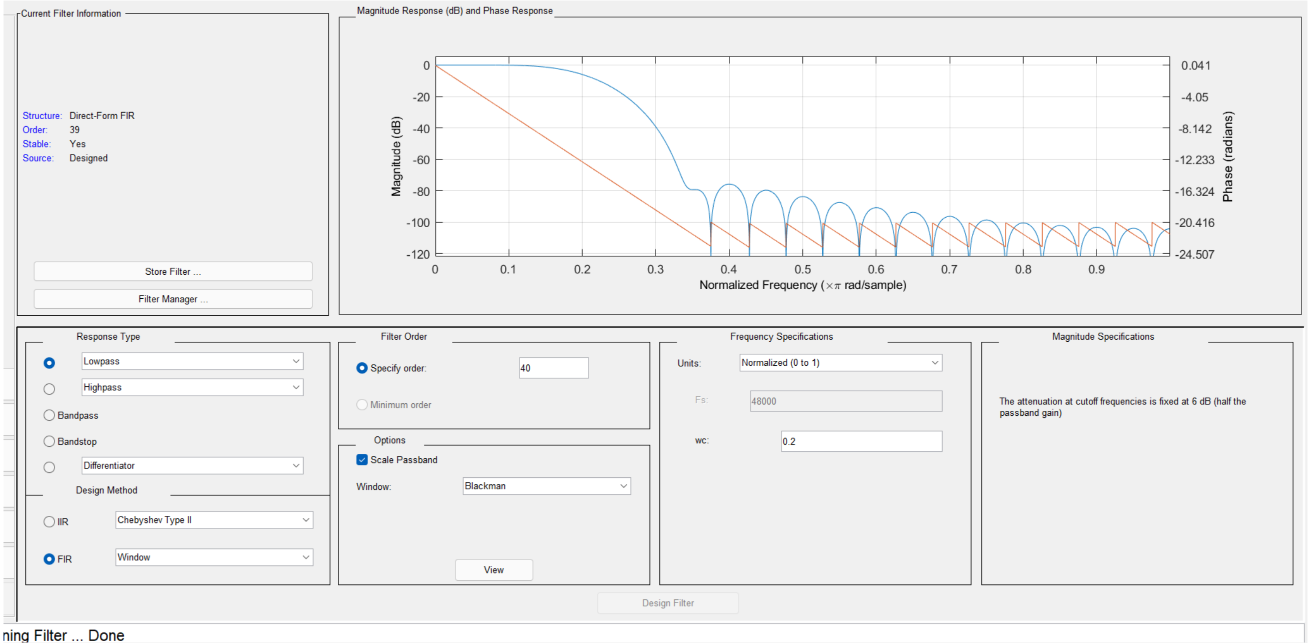
Frequency Specifications

Units: Normalized (0 to 1)
Fs: 48000
wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

signing Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 39
Stable: Yes
Source: Designed

Response Type

- ● Lowpass
- ○ Highpass
- ○ Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method

- ○ IIR Chebyshev Type II
- ● FIR Window

Filter Order

- ● Specify order: 40
- ○ Minimum order

Options

☑ Scale Passband

Window: Blackman

Frequency Specifications

Units: Normalized (0 to 1)

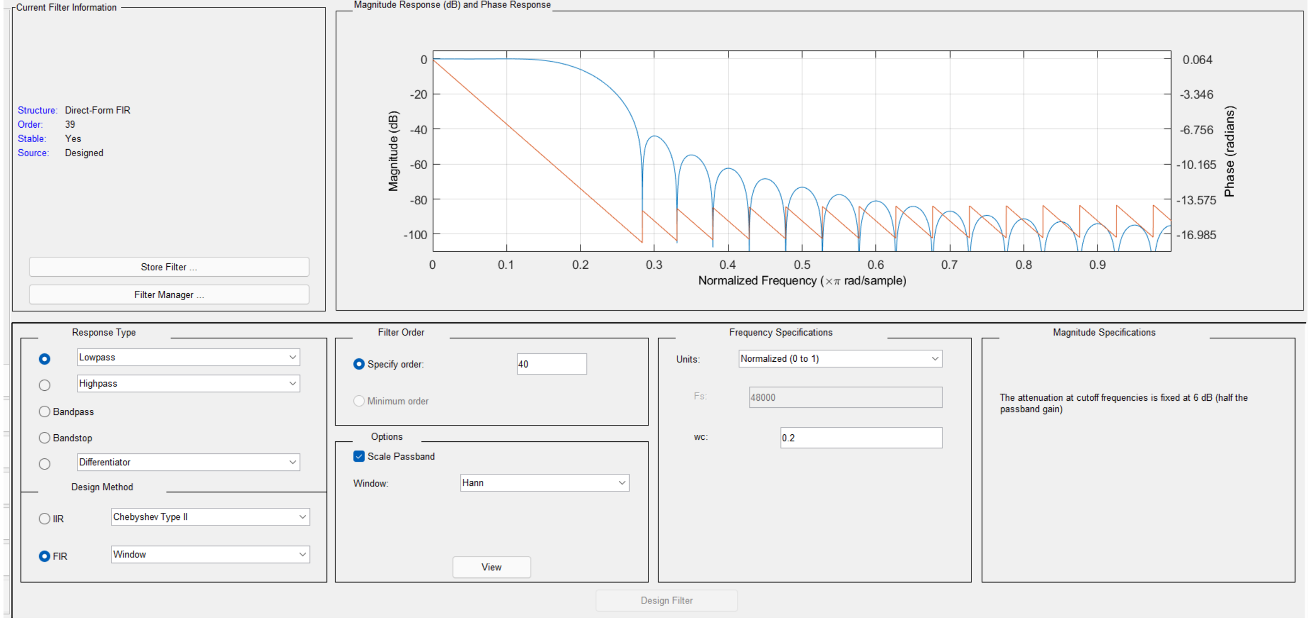
Fs: 48000

wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

ning Filter ... Done



Current Filter Information

Structure: Direct-Form FIR
Order: 39
Stable: Yes
Source: Designed

Response Type

- ● Lowpass
- ○ Highpass
- ○ Bandpass
- ○ Bandstop
- ○ Differentiator

Design Method

- ○ IIR Chebyshev Type II
- ● FIR Window

Filter Order

- ● Specify order: 40
- ○ Minimum order

Options

☑ Scale Passband

Window: Hann

Frequency Specifications

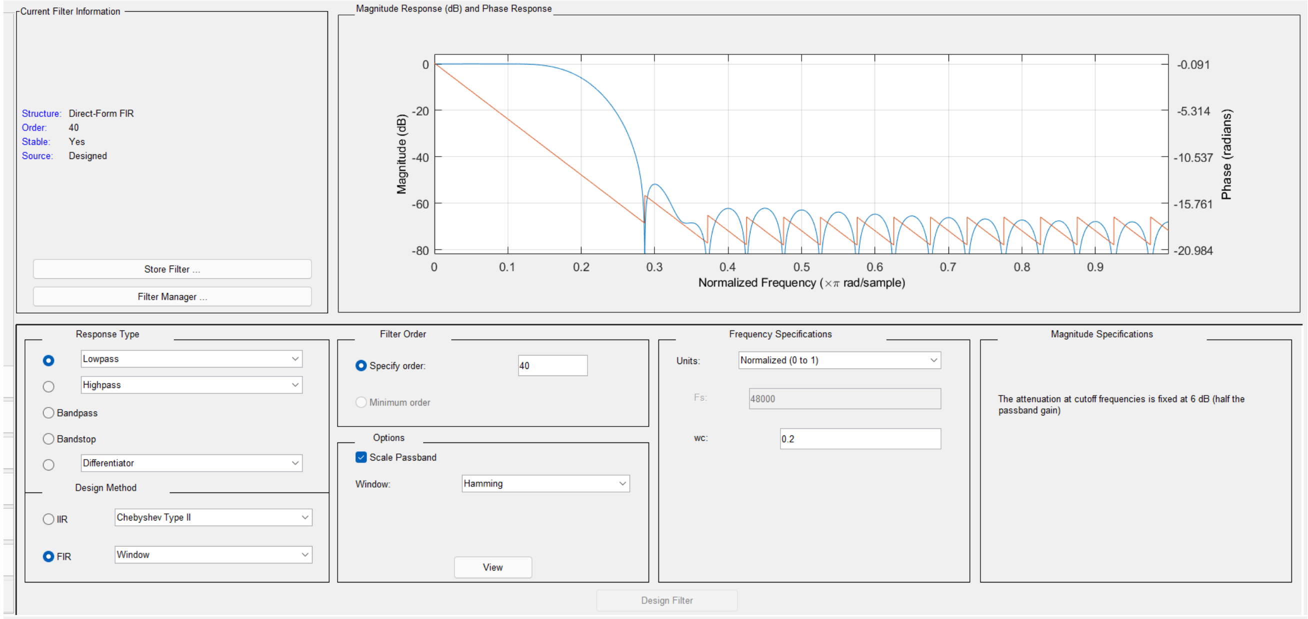
Units: Normalized (0 to 1)

Fs: 48000

wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)



Current Filter Information

Structure: Direct-Form FIR
Order: 40
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Chebyshev Type II
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

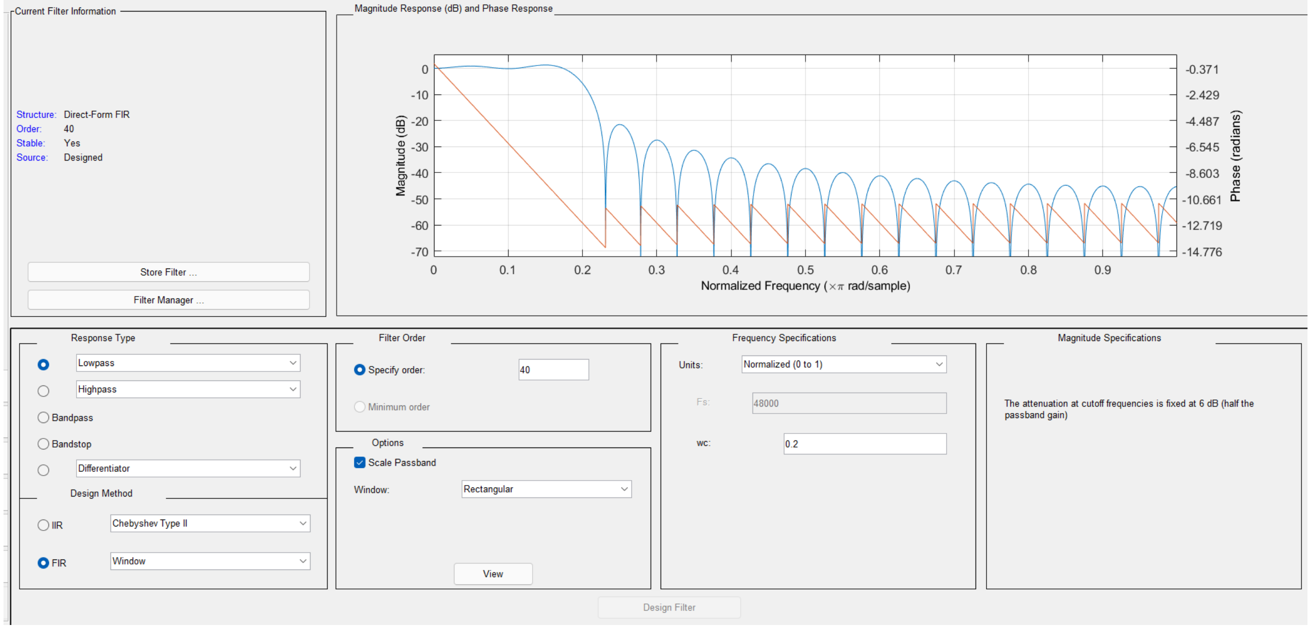
☑ Scale Passband
Window: Hamming

Frequency Specifications

Units: Normalized (0 to 1)
Fs: 48000
wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)



Current Filter Information

Structure: Direct-Form FIR
Order: 40
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

○ IIR Chebyshev Type II
● FIR Window

Filter Order

● Specify order: 40
○ Minimum order

Options

☑ Scale Passband
Window: Rectangular

Frequency Specifications

Units: Normalized (0 to 1)
Fs: 48000
wc: 0.2

Magnitude Specifications

The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)

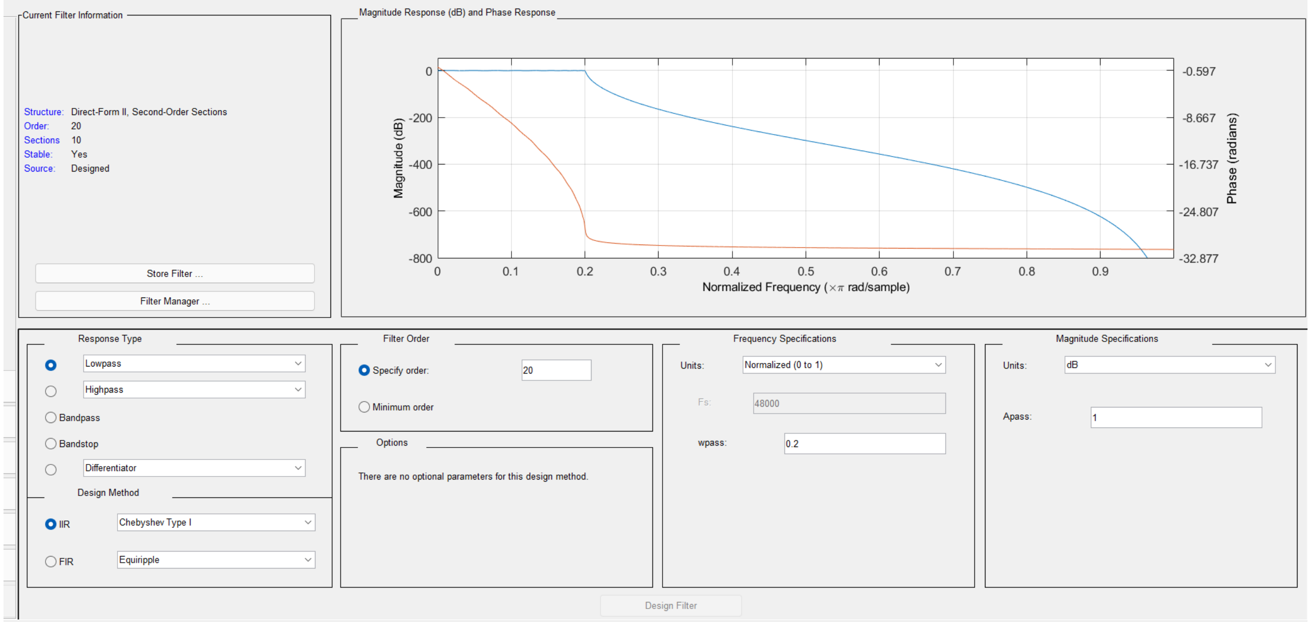
Problem 2:

2) Design a low-pass IIR filter using MATLAB command ‘fdatool’ with normalized cutoff frequency of 0.2 rad/pi for the following specifications.

Analog prototype filter:
(i) Butterworth filter
(ii) Chebyshev filter

Export the filter coefficients in MATLAB workspace as variables for Problem 3.

Designs:



Current Filter Information

Structure: Direct-Form II, Second-Order Sections
Order: 20
Sections: 10
Stable: Yes
Source: Designed

Response Type: ● Lowpass

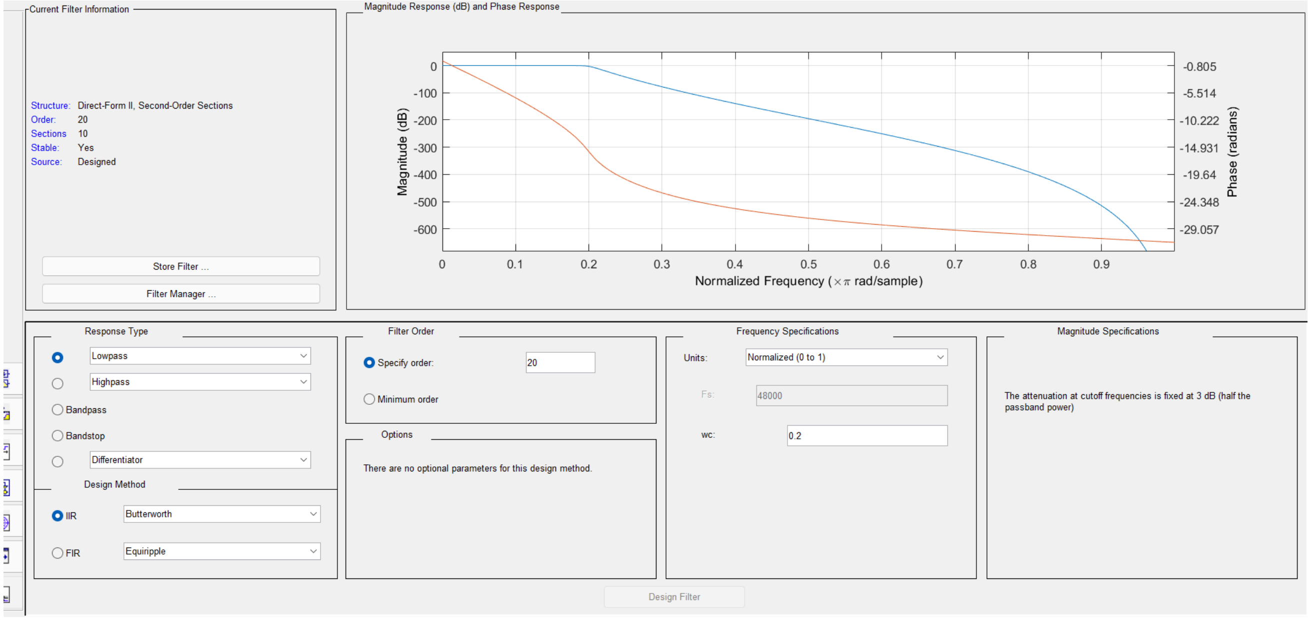
Design Method: ● IIR Chebyshev Type I

Filter Order: ● Specify order: 20

Options: There are no optional parameters for this design method.

Frequency Specifications: Units: Normalized (0 to 1); Fs: 48000; wpass: 0.2

Magnitude Specifications: Units: dB; Apass: 1



Current Filter Information

Structure: Direct-Form II, Second-Order Sections
Order: 20
Sections: 10
Stable: Yes
Source: Designed

Response Type: ● Lowpass

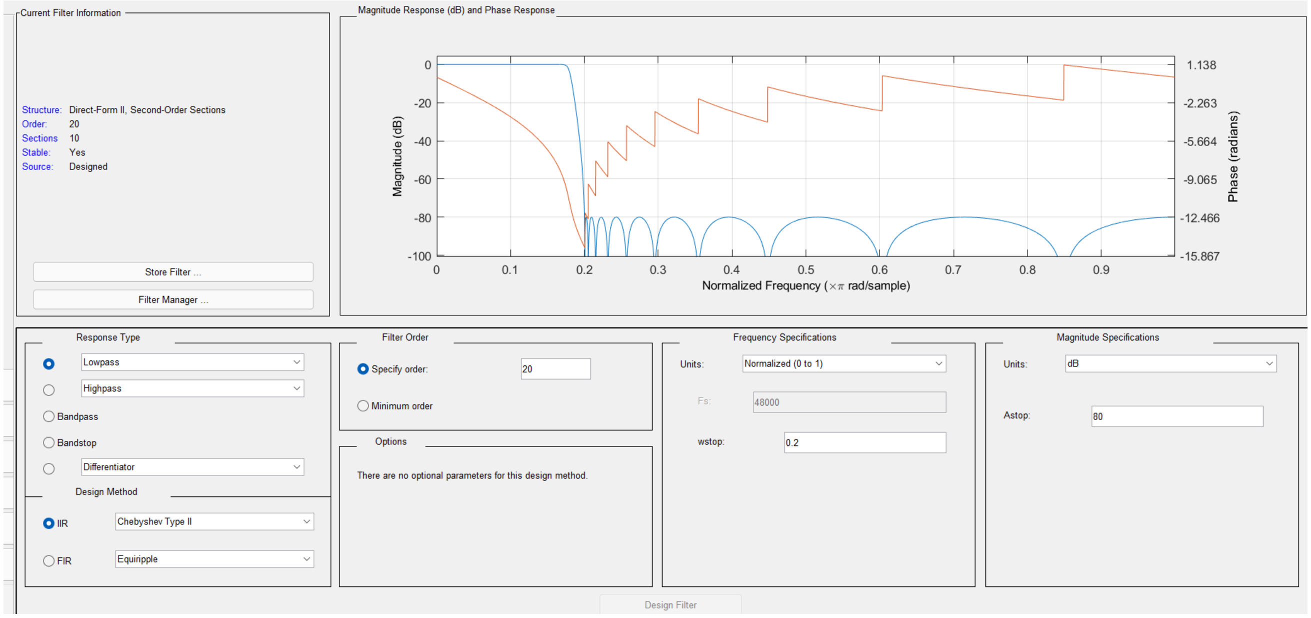
Design Method: ● IIR Butterworth

Filter Order: ● Specify order: 20

Options: There are no optional parameters for this design method.

Frequency Specifications: Units: Normalized (0 to 1); Fs: 48000; wc: 0.2

Magnitude Specifications: The attenuation at cutoff frequencies is fixed at 3 dB (half the passband power)



Current Filter Information

Structure: Direct-Form II, Second-Order Sections
Order: 20
Sections 10
Stable: Yes
Source: Designed

Response Type

● Lowpass
○ Highpass
○ Bandpass
○ Bandstop
○ Differentiator

Design Method

● IIR Chebyshev Type II
○ FIR Equiripple

Filter Order

● Specify order: 20
○ Minimum order

Options

There are no optional parameters for this design method.

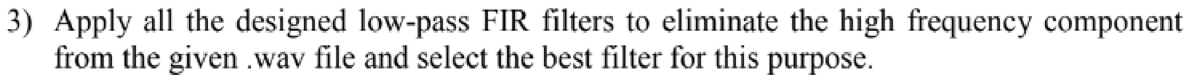
Frequency Specifications

Units: Normalized (0 to 1)
Fs: 48000
wstop: 0.2

Magnitude Specifications

Units: dB
Astop: 80

Problem 3:



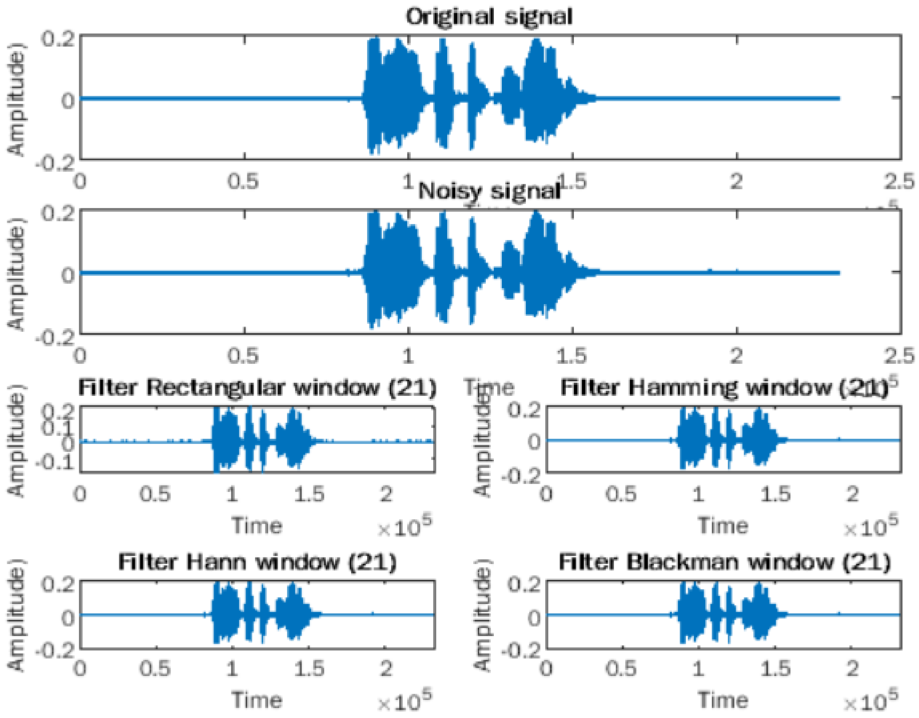
3) Apply all the designed low-pass FIR filters to eliminate the high frequency component from the given .wav file and select the best filter for this purpose.

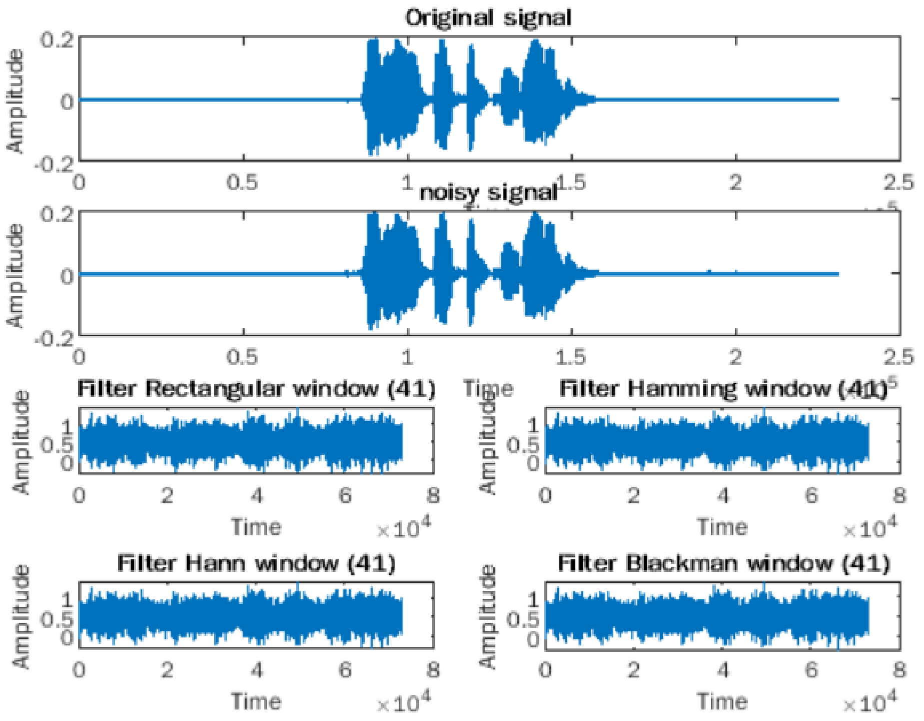
Code:

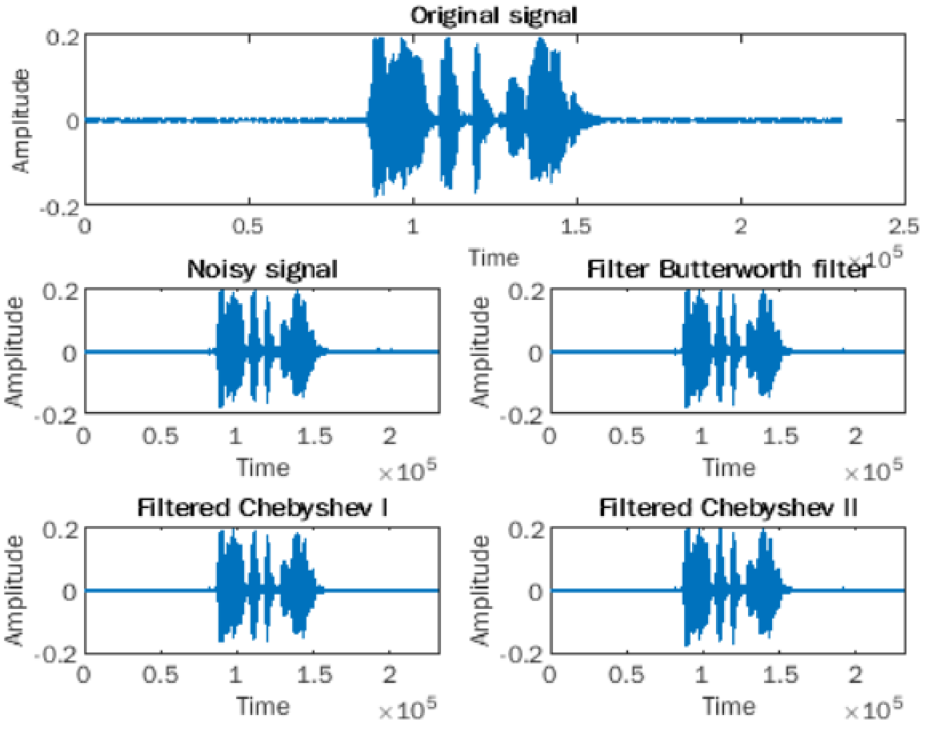
```
close all;
load matlab.mat
[y, Fs] = audioread("Recording (2).m4a");
sound(y,Fs)
y=y(:,1);
YNoisy = y + 0.002*rand(size(y));
YRect21= filter(rect21,1, YNoisy);
YHamming21= filter(hamming21,1, YNoisy);
YHann21= filter(hann21,1, YNoisy);
YBlackman21= filter(blackman21,1, YNoisy);
subplot(4,2,[1 2]);
plot(y);title('Original signal');
xlabel("Time");ylabel("Amplitude)");
subplot(4,2,[3 4]);
plot(YNoisy);title('Noisy signal');
xlabel("Time");ylabel("Amplitude)");
subplot(425);
plot(YRect21);title('Filter Rectangular window (21)');
xlabel("Time");ylabel("Amplitude)");
subplot(426);
plot(YHamming21);title('Filter Hamming window (21)');
xlabel("Time");ylabel("Amplitude)");
subplot(427);
plot(YHann21);title('Filter Hann window (21)');
xlabel("Time");ylabel("Amplitude)");
subplot(428);
plot(YBlackman21);title('Filter Blackman window (21)');
xlabel("Time");ylabel("Amplitude)");

figure
YRect41= filter(rect41,1, YNoisy);
YHamming41= filter(hamming41,1, YNoisy);
YHann41= filter(hann41,1, YNoisy);
YBlackman41= filter(blackman41,1, YNoisy);
subplot(4,2,[1 2]);
plot(y);title('Original signal');
xlabel("Time");ylabel("Amplitude");
subplot(4,2,[3 4]);
plot(YNoisy);title('noisy signal');
xlabel("Time");ylabel("Amplitude");
subplot(425);
plot(y_rect41);title('Filter Rectangular window (41)');
xlabel("Time");ylabel("Amplitude");
subplot(426);
plot(y_hamming41);title('Filter Hamming window (41)');
xlabel("Time");ylabel("Amplitude");
subplot(427);
plot(y_hann41);title('Filter Hann window (41)');
xlabel("Time");ylabel("Amplitude");
subplot(428);
plot(y_blackman41);title('Filter Blackman window (41)');
xlabel("Time");ylabel("Amplitude");
```

```
figure
[b1,a1] = sos2tf(butter,G);
y_butter= filter(b1,a1,YNoisy);
[b2,a2] = sos2tf(chebyshev1,G1);
y_cheb1= filter(b2,a2, YNoisy);
[b3,a3] = sos2tf(chebyshev2,G2);
y_cheb2= filter(b3,a3, YNoisy);

subplot(3,2,[1 2]);
plot(y);title('Original signal');
xlabel("Time");ylabel("Amplitude");
subplot(3,2,3);
plot(YNoisy);title('Noisy signal');
xlabel("Time");ylabel("Amplitude");
subplot(324);
plot(y_butter);title('Filter Butterworth filter');
xlabel("Time");ylabel("Amplitude");
subplot(325);
plot(y_cheb1);title('Filtered Chebyshev I');
xlabel("Time");ylabel("Amplitude");
subplot(326);
plot(y_cheb2);title('Filtered Chebyshev II');
xlabel("Time");ylabel("Amplitude");
```

Graph:




Original signal
Amplitude
0.2
0
-0.2
0
0.5
1
1.5
2
2.5
noisy signal
Amplitude
Time
×10^5
Filter Rectangular window (41)
Filter Hamming window (41)
Filter Hann window (41)
Filter Blackman window (41)
Amplitude
1
0.5
0
0
2
4
6
8
Time
×10^4


Original signal
Amplitude
0.2
0
-0.2
0
0.5
1
1.5
2
2.5
Time
×10^5
Noisy signal
Filter Butterworth filter
Filtered Chebyshev I
Filtered Chebyshev II
Amplitude
Time
×10^5