Experiment No 09

BECS 32461

Paper A

FILTER DESIGN AND ANALYSIS WITH (FDA) TOOL

Student Name: W. K. G. K. Jayawardana

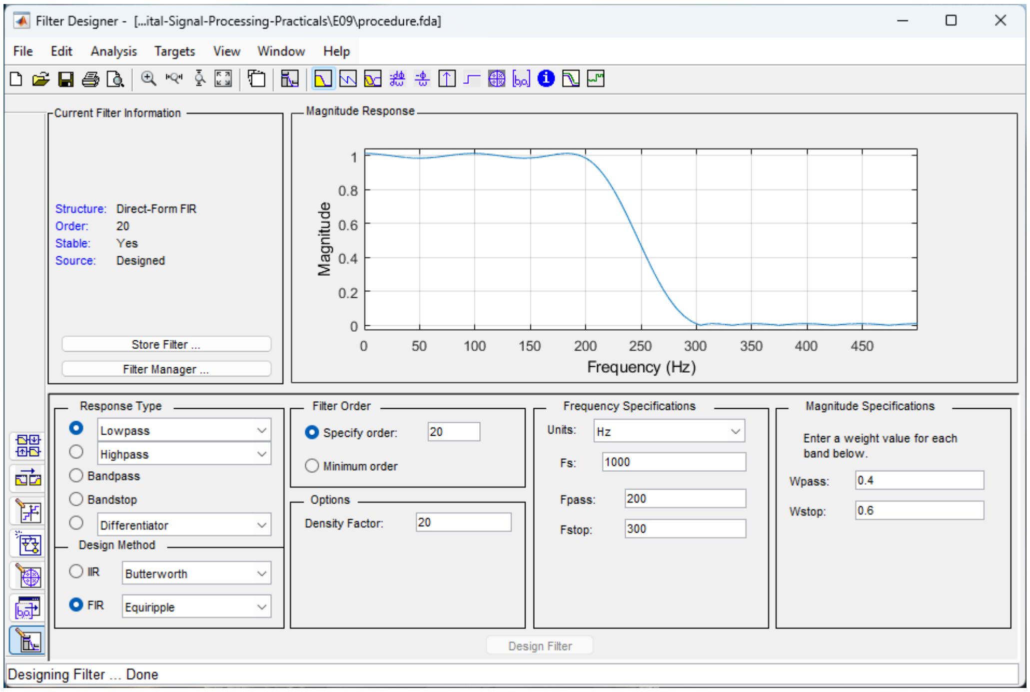
Student No: EC/2021/006

Date Performed: 2025/11/21

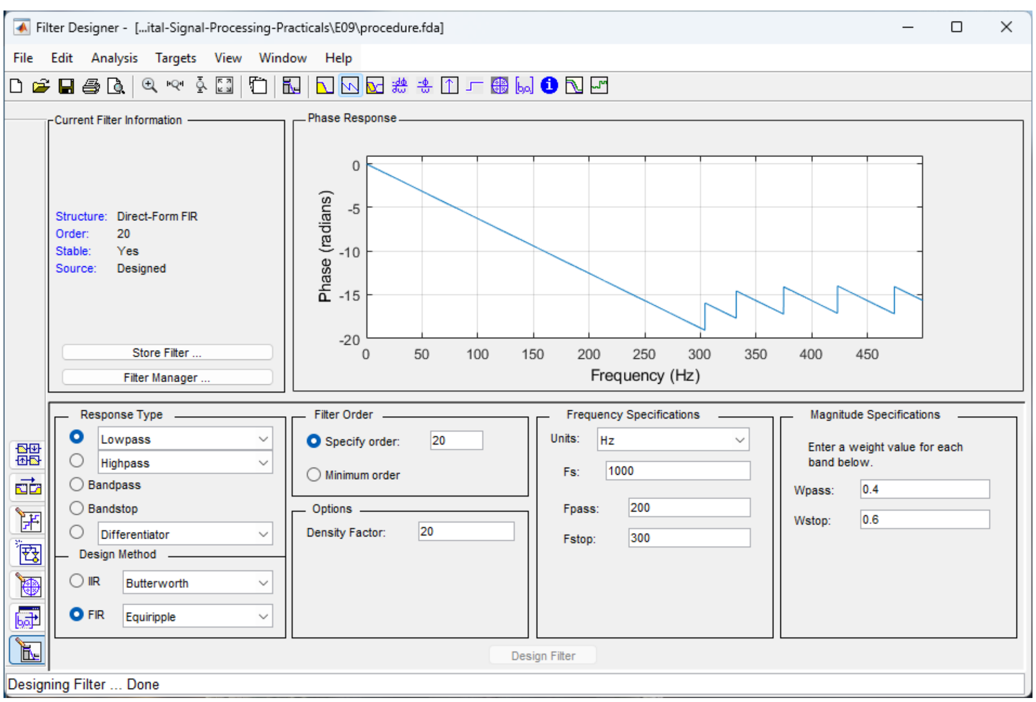
Date Submitted: 2025/11/21

PROCEDURE

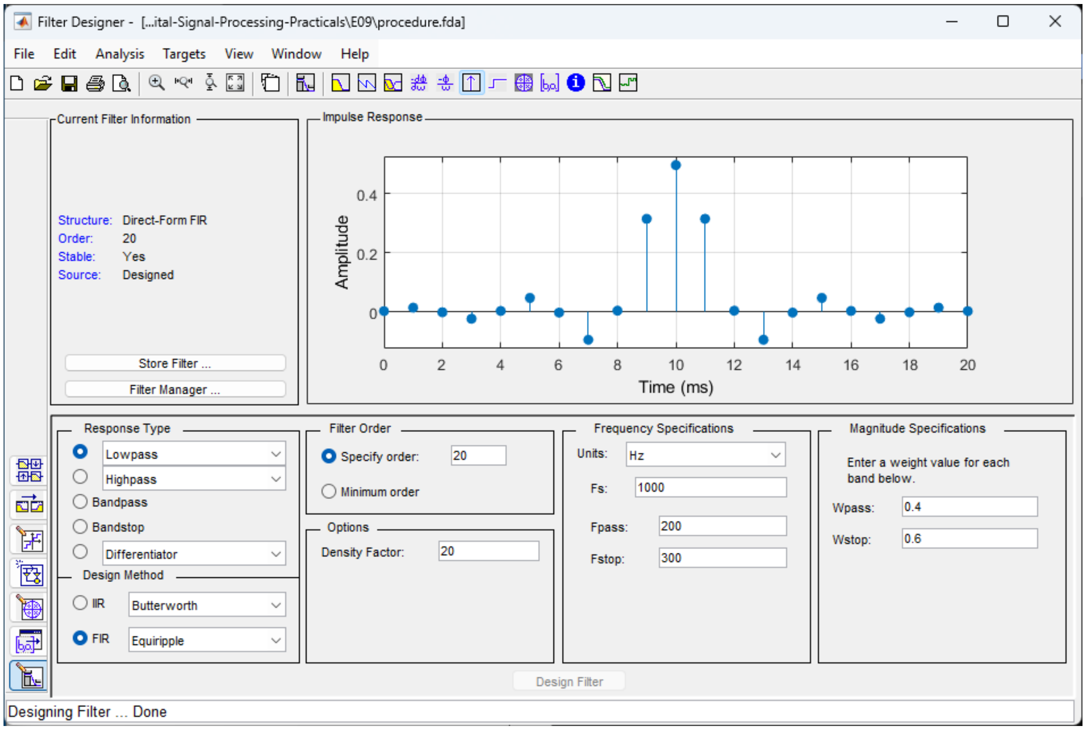
F01.



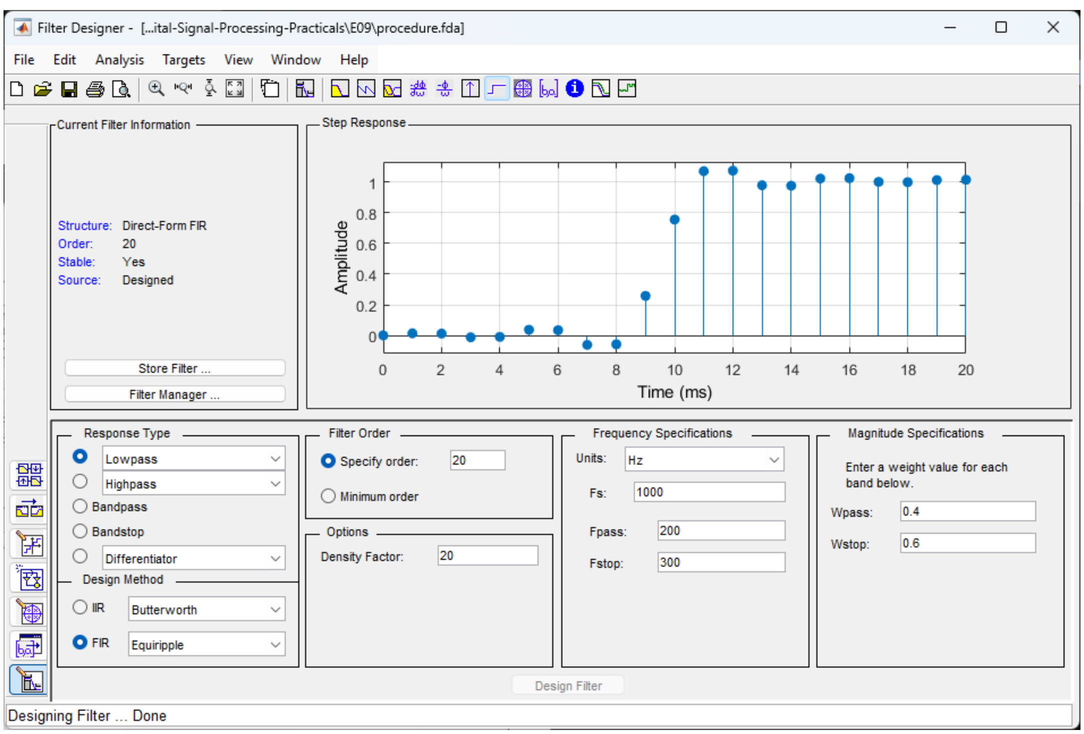
F02.



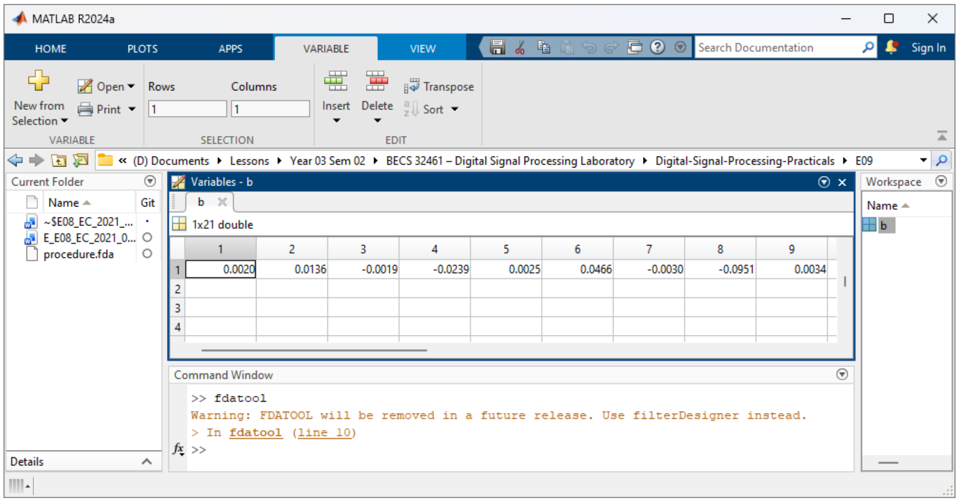
F03.


Filter Designer - [...ital-Signal-Processing-Practicals\E09\procedure.fda]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Impulse Response
Amplitude
Time (ms)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Equiripple
Filter Order
Specify order: 20
Minimum order
Options
Density Factor: 20
Frequency Specifications
Units: Hz
Fs: 1000
Fpass: 200
Fstop: 300
Magnitude Specifications
Enter a weight value for each band below.
Wpass: 0.4
Wstop: 0.6
Design Filter
Designing Filter ... Done

F04.

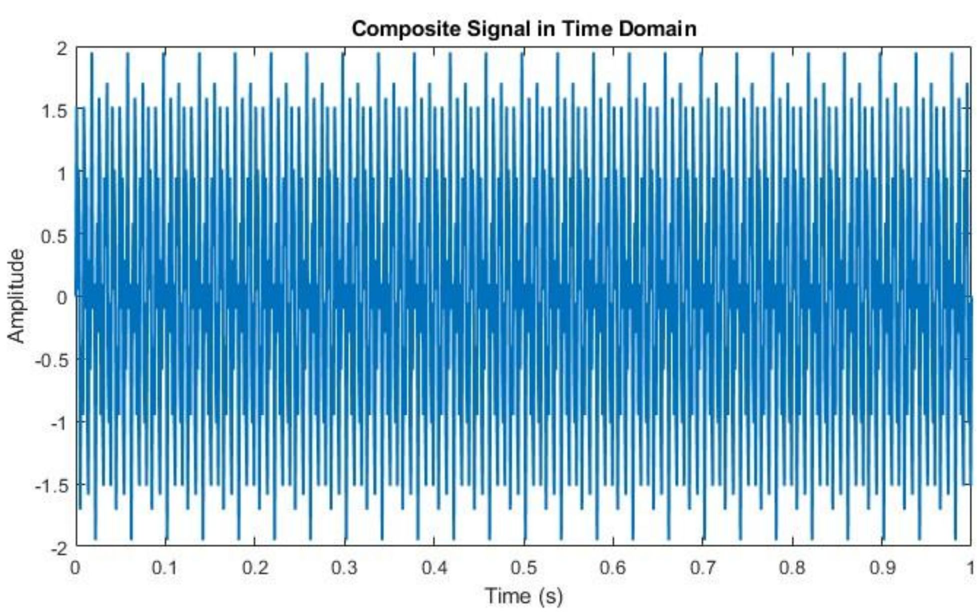

Filter Designer - [...ital-Signal-Processing-Practicals\E09\procedure.fda]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 20
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Step Response
Amplitude
Time (ms)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Equiripple
Filter Order
Specify order: 20
Minimum order
Options
Density Factor: 20
Frequency Specifications
Units: Hz
Fs: 1000
Fpass: 200
Fstop: 300
Magnitude Specifications
Enter a weight value for each band below.
Wpass: 0.4
Wstop: 0.6
Design Filter
Designing Filter ... Done

P09.



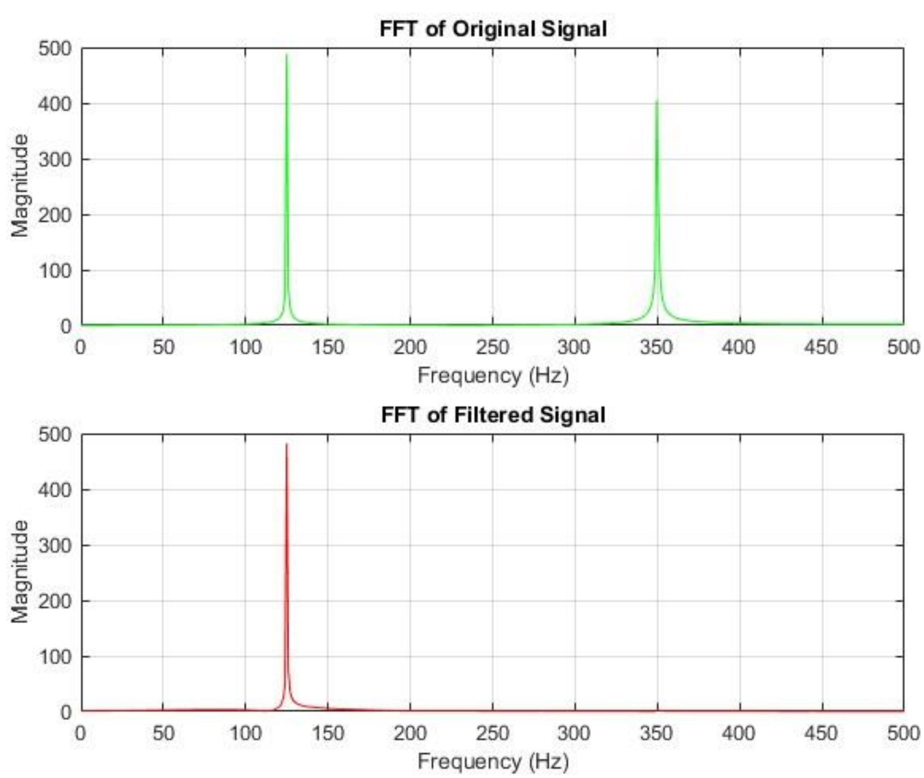
F05.

```matlab
Fs = 1000;
t = 0:1/Fs:1;

x1 = sin(2*pi*125*t);
x2 = sin(2*pi*350*t);

x = x1 + x2;

figure;
plot(t, x,'LineWidth',1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('Composite Signal in Time Domain');
```



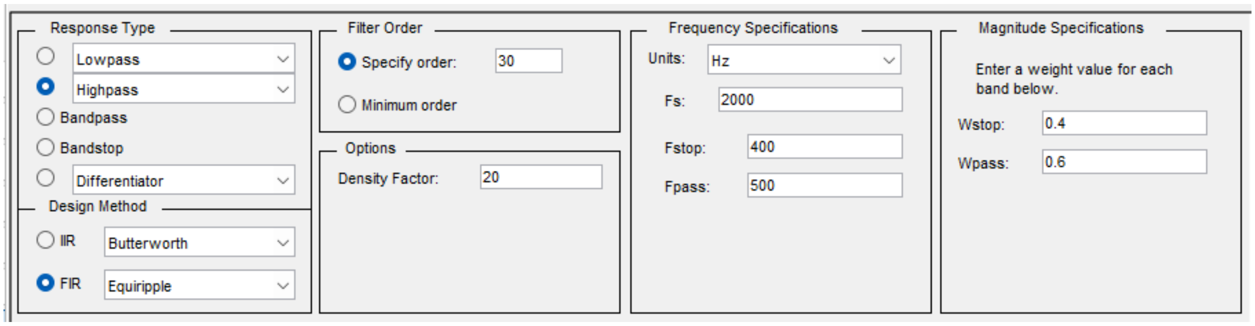
F06.

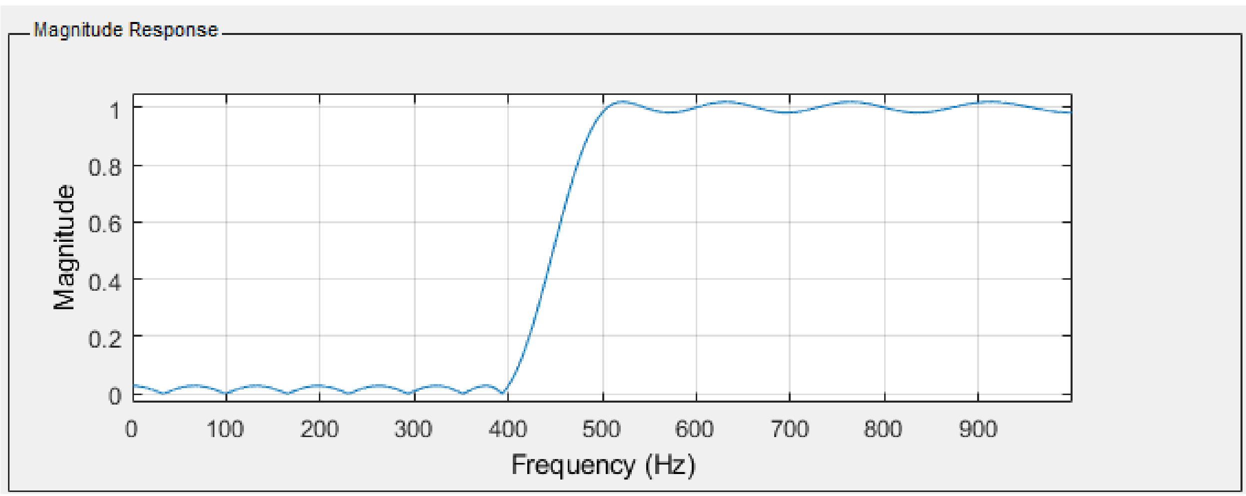
```
Fs = 1000;
t = 0:1/Fs:1;
x1 = sin(2*pi*125*t);
x2 = sin(2*pi*350*t);
x = x1 + x2;

y = filter(b, 1, x);

N = length(x);
X = abs(fft(x));
Y = abs(fft(y));

f = (0:N-1)*(Fs/N);

figure;
subplot(2,1,1);
plot(f, X,'g-');
title('FFT of Original Signal');
xlabel('Frequency (Hz)');
ylabel('Magnitude');
xlim([0 500]);
grid on;

subplot(2,1,2);
plot(f, Y,'r-');
title('FFT of Filtered Signal');
xlabel('Frequency (Hz)');
ylabel('Magnitude');
xlim([0 500]);
grid on;
```

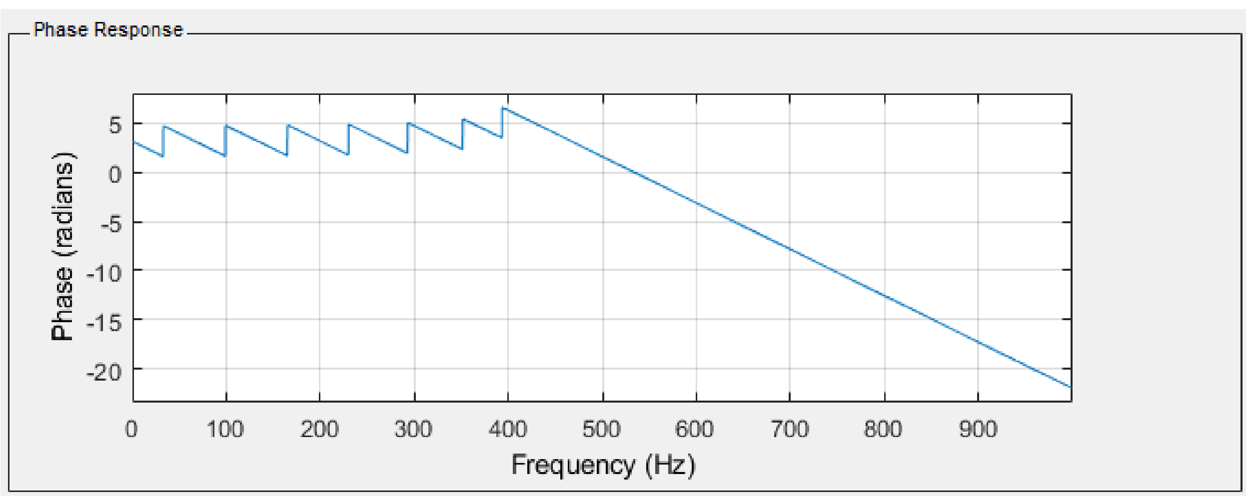


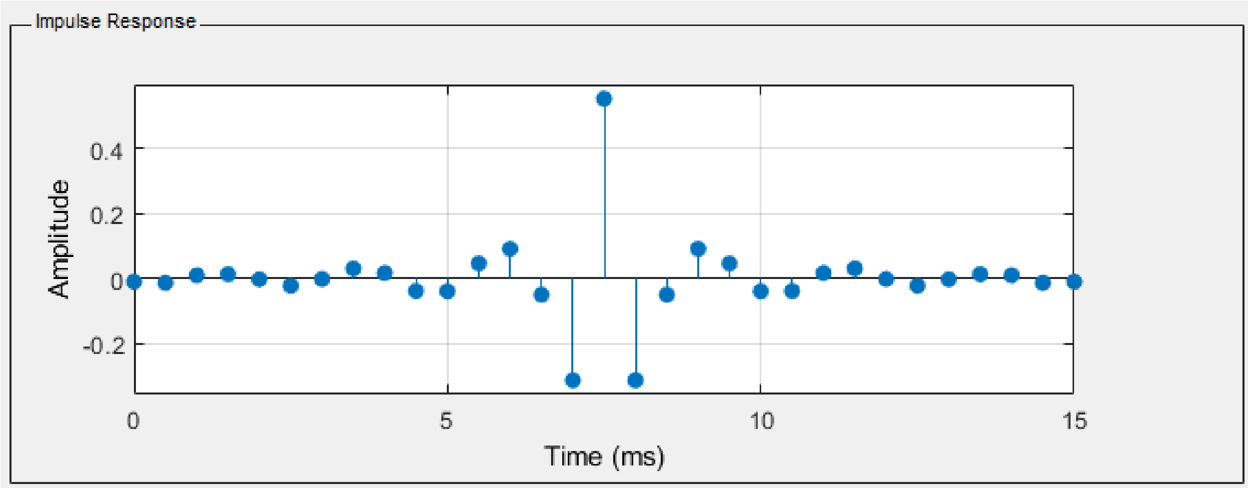
EXERCISE

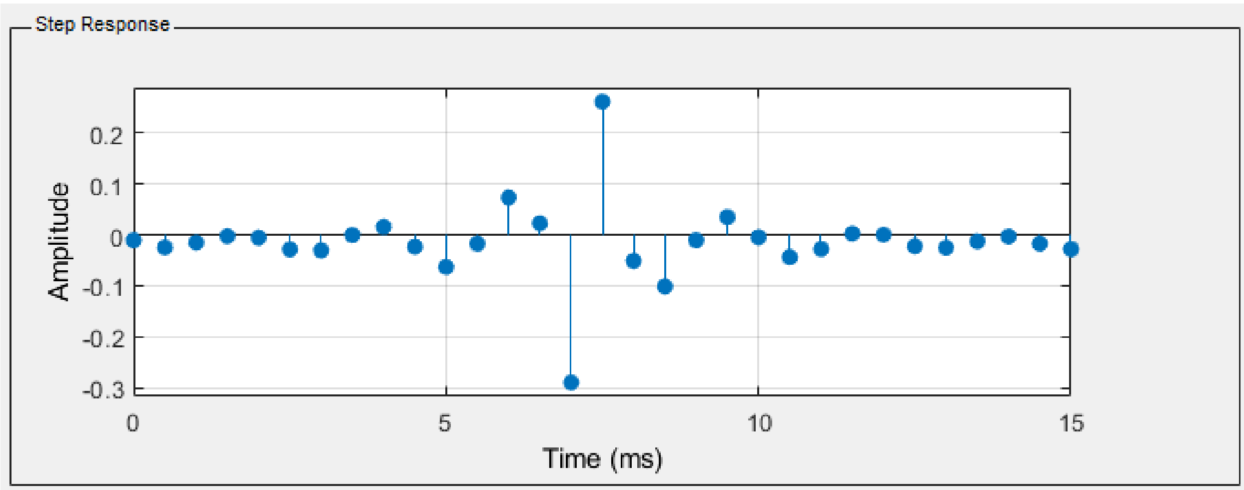
E01.

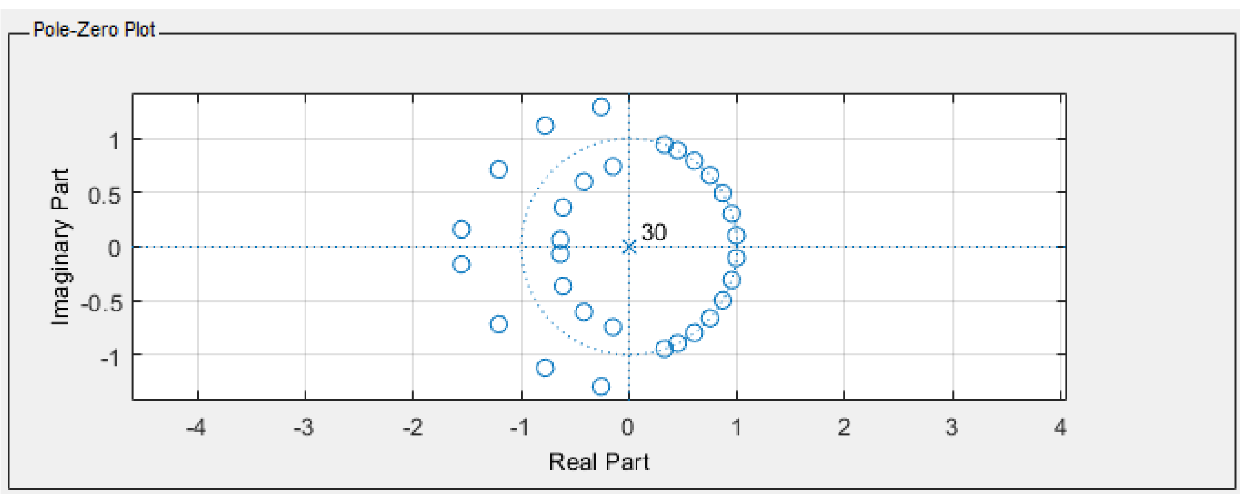






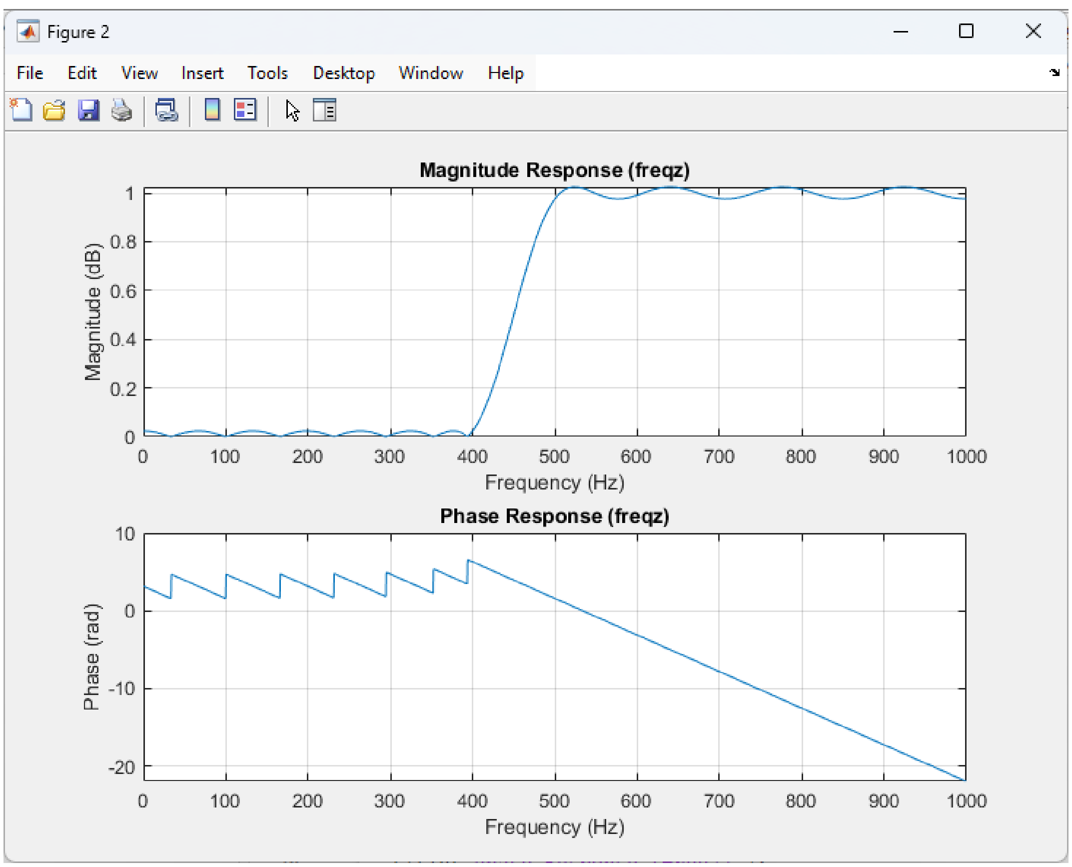

Impulse Response
Amplitude
0.4
0.2
0
-0.2
0
5
10
15
Time (ms)


Step Response
Amplitude
0.2
0.1
0
-0.1
-0.2
-0.3
0
5
10
15
Time (ms)


Pole-Zero Plot
Imaginary Part
1
0.5
0
-0.5
-1
30
-4
-3
-2
-1
0
1
2
3
4
Real Part

c.

```
Fs = 2000;
Fstop = 400;
Fpass = 500;
N = 30;

nyq = Fs/2;
f = [0 Fstop Fpass nyq] / nyq;
a = [0 0 1 1];

nfft = 2048;
[H, w] = freqz(b_hp, 1, nfft, Fs);

figure;
subplot(2,1,1);
plot(w, abs(H));
grid on;
xlabel('Frequency (Hz)');
ylabel('Magnitude (dB)');
title('Magnitude Response (freqz)');

subplot(2,1,2);
plot(w, unwrap(angle(H)));
grid on;
xlabel('Frequency (Hz)');
ylabel('Phase (rad)');
title('Phase Response (freqz)');
```

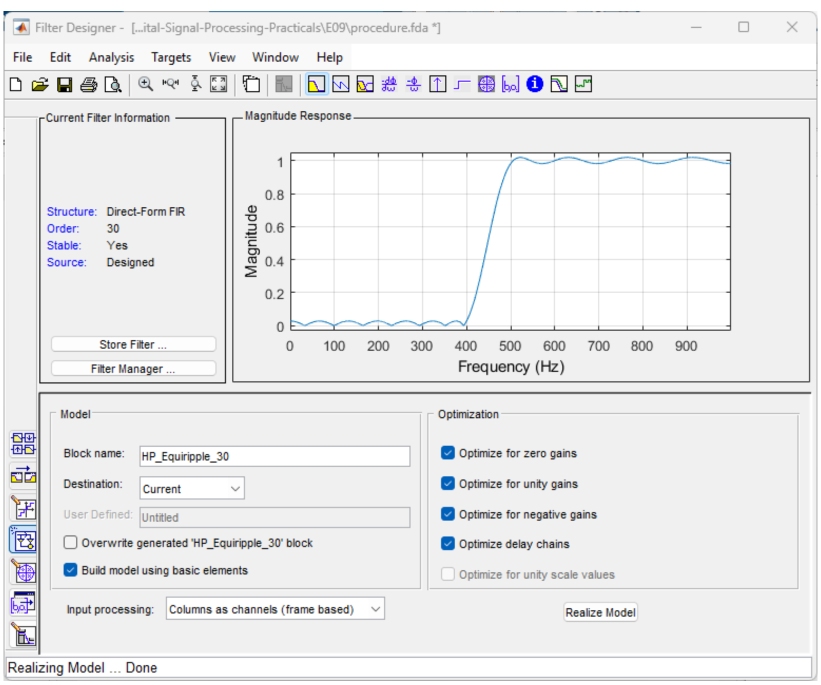


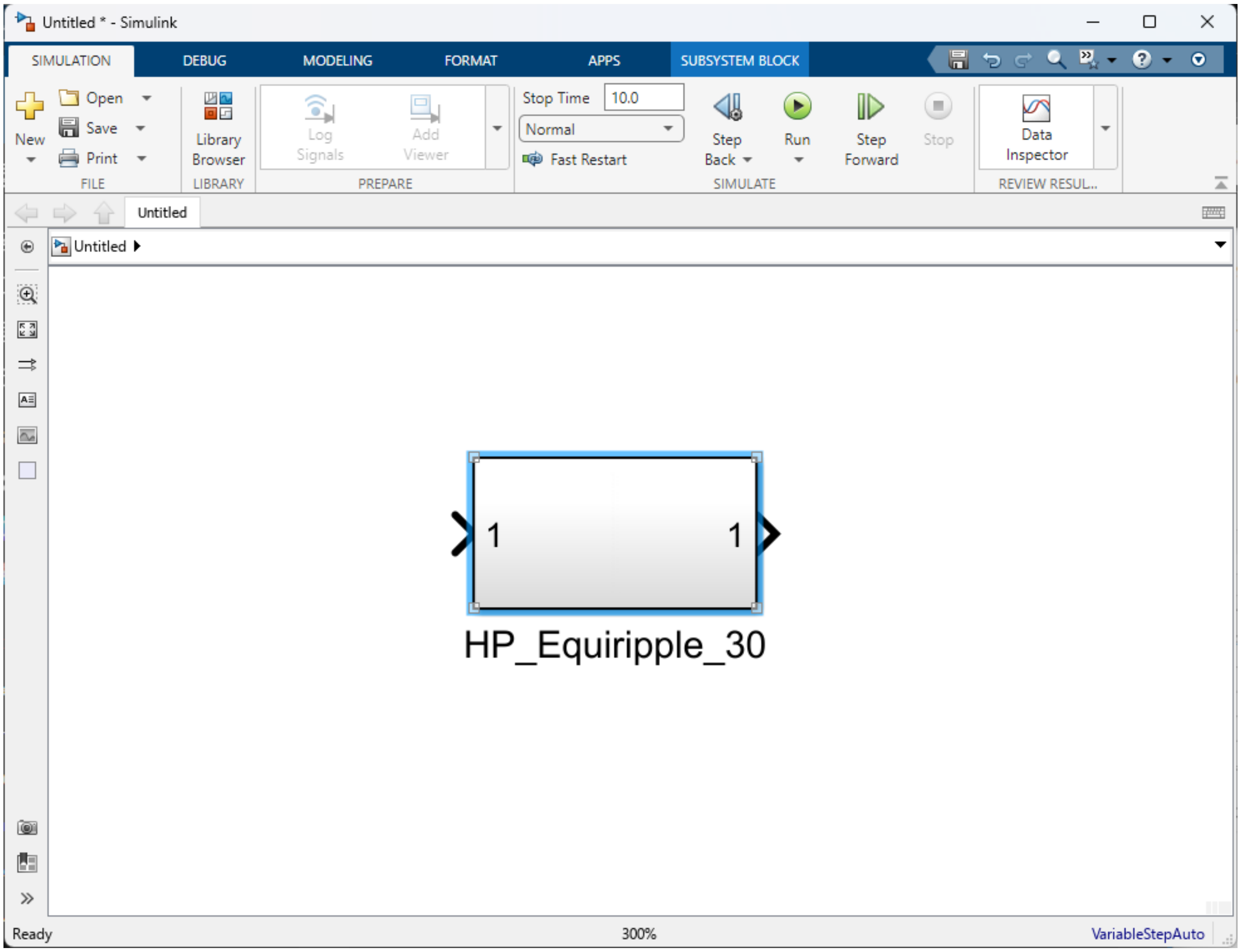
d.

```
Ts = 1/Fs;
Hd_tf = tf(b_hp, 1, Ts);
disp('Transfer function (discrete-time):');
Hd_tf
```

```
Hd_tf =


  -0.006609 z^30 - 0.01643 z^29 + 0.008775 z^28 + 0.01354 z^27

          - 0.00256 z^26 - 0.02217 z^25 - 0.003988 z^24 + 0.03005 z^23

          + 0.01725 z^22 - 0.03782 z^21 - 0.04116 z^20 + 0.04429 z^19

          + 0.09157 z^18 - 0.04858 z^17 - 0.3133 z^16 + 0.5501 z^15

          - 0.3133 z^14 - 0.04858 z^13 + 0.09157 z^12 + 0.04429 z^11

          - 0.04116 z^10 - 0.03782 z^9 + 0.01725 z^8 + 0.03005 z^7

          - 0.003988 z^6 - 0.02217 z^5 - 0.00256 z^4 + 0.01354 z^3

                                  + 0.008775 z^2 - 0.01643 z - 0.006609
```

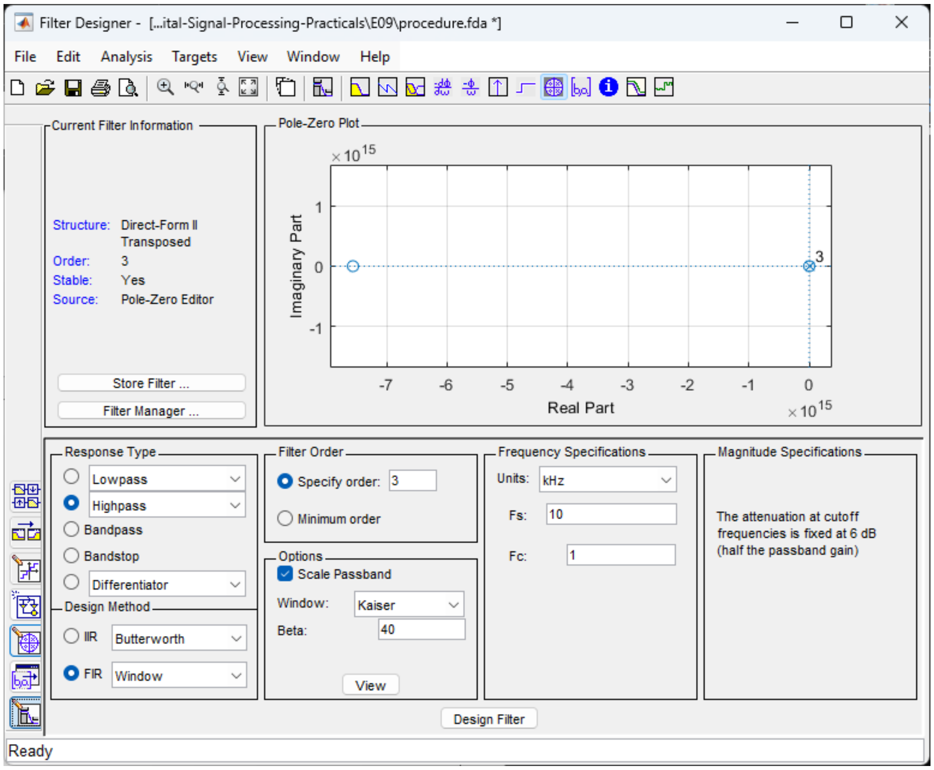
e.



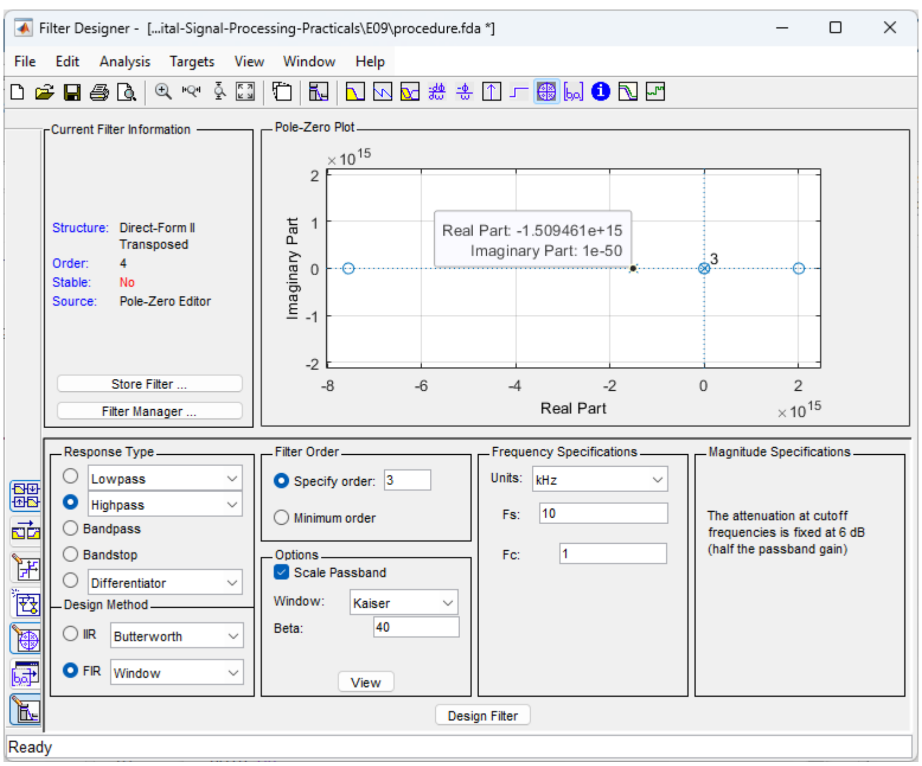

Untitled * - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
SUBSYSTEM BLOCK
New
Open
Save
Print
FILE
Library Browser
LIBRARY
Log Signals
Add Viewer
PREPARE
Stop Time
10.0
Normal
Fast Restart
Step Back
Run
Step Forward
Stop
SIMULATE
Data Inspector
REVIEW RESUL...
Untitled
1
1
HP_Equiripple_30
Ready
300%
VariableStepAuto

E02.

c.



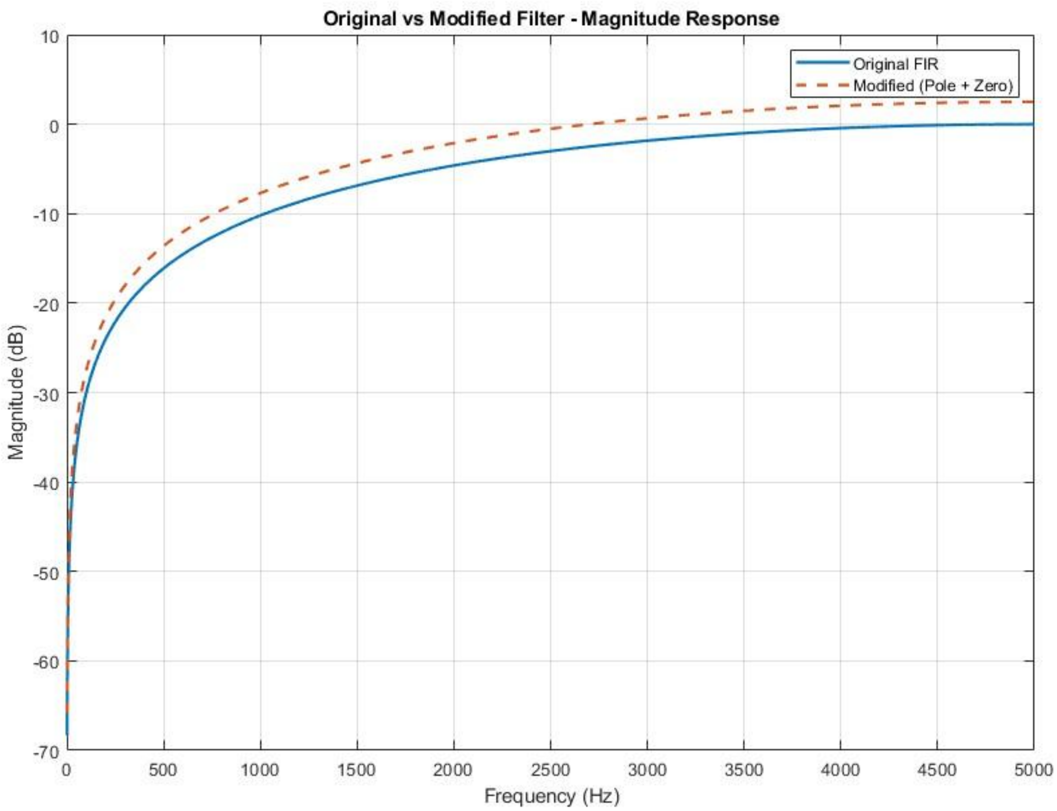
Filter Designer - [...ital-Signal-Processing-Practicals\E09\procedure.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II Transposed
Order: 3
Stable: Yes
Source: Pole-Zero Editor
Store Filter ...
Filter Manager ...
Pole-Zero Plot
Imaginary Part
Real Part
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Window
Filter Order
Specify order: 3
Minimum order
Options
Scale Passband
Window: Kaiser
Beta: 40
View
Frequency Specifications
Units: kHz
Fs: 10
Fc: 1
Magnitude Specifications
The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)
Design Filter
Ready



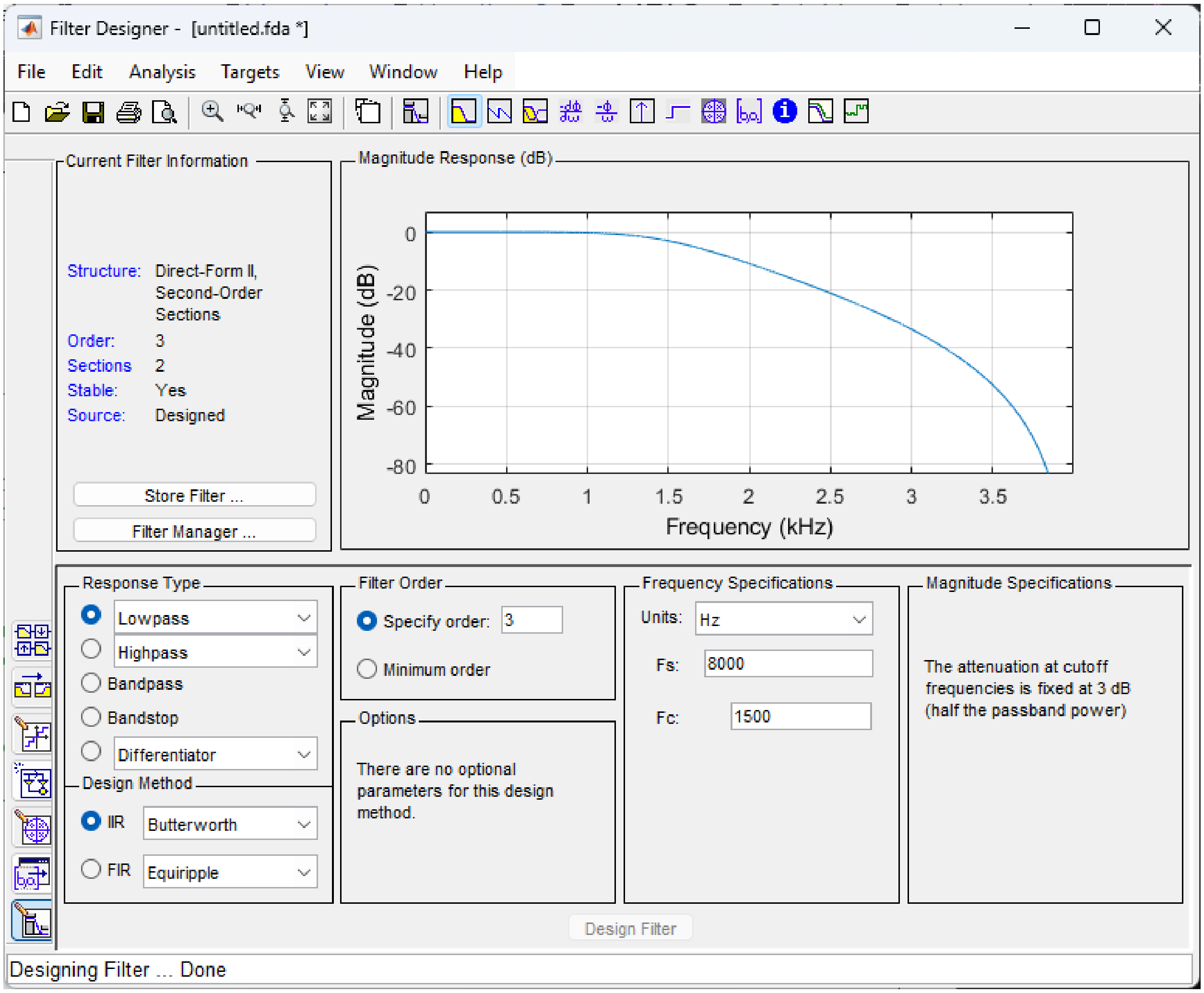
Filter Designer - [...ital-Signal-Processing-Practicals\E09\procedure.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II Transposed
Order: 4
Stable: No
Source: Pole-Zero Editor
Store Filter ...
Filter Manager ...
Pole-Zero Plot
Real Part: -1.509461e+15
Imaginary Part: 1e-50
Imaginary Part
Real Part
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Window
Filter Order
Specify order: 3
Minimum order
Options
Scale Passband
Window: Kaiser
Beta: 40
View
Frequency Specifications
Units: kHz
Fs: 10
Fc: 1
Magnitude Specifications
The attenuation at cutoff frequencies is fixed at 6 dB (half the passband gain)
Design Filter
Ready

After adding a **zero at z = 2** and a **pole at z = –1.5**, the modified system becomes an IIR filter. The impulse response plot shows that the output does **not decay** with time; instead, it grows in magnitude, indicating an unstable behavior. This occurs because the added pole is located at **|–1.5| = 1.5**, which is **outside the unit circle**. For a discrete-time system to be stable, all poles must lie inside the unit circle (|p| < 1). Since this condition is violated, the modified filter is **unstable**, and its impulse response clearly reflects this instability through unbounded oscillations.

d.

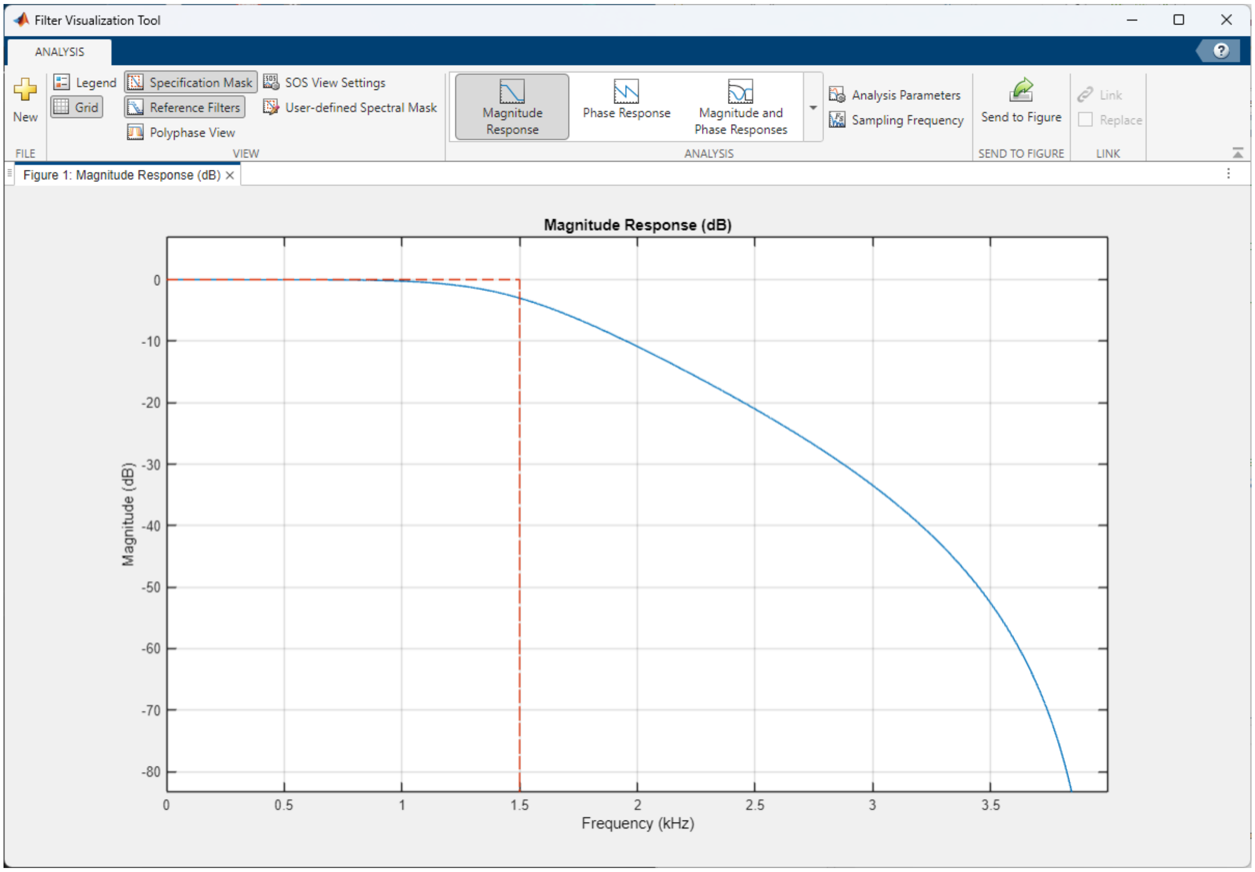


E03.

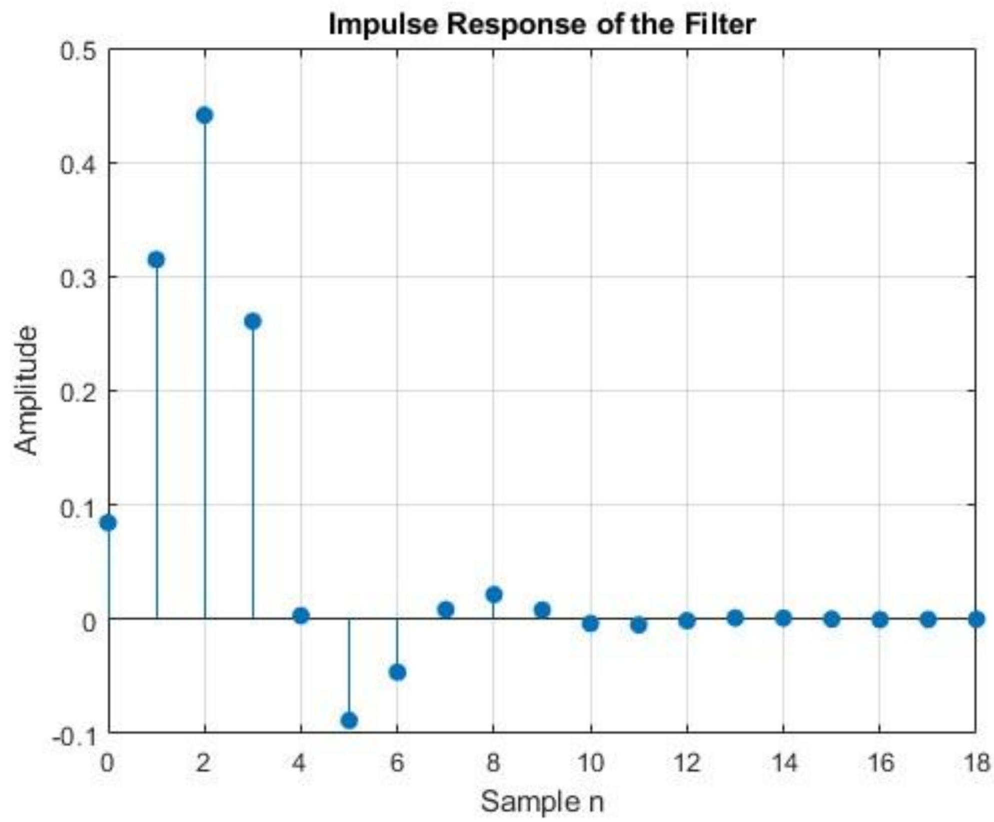



```
function Hd = butter_lp3
%BUTTER_LP3 Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 24.1 and DSP System Toolbox 24.1.
% Generated on: 21-Nov-2025 15:27:40

% Butterworth Lowpass filter designed using FDESIGN.LOWPASS.

% All frequency values are in Hz.
Fs = 8000;  % Sampling Frequency

N  = 3;     % Order
Fc = 1500;  % Cutoff Frequency

% Construct an FDESIGN object and call its BUTTER method.
h  = fdesign.lowpass('N,F3dB', N, Fc, Fs);
Hd = design(h, 'butter');
```

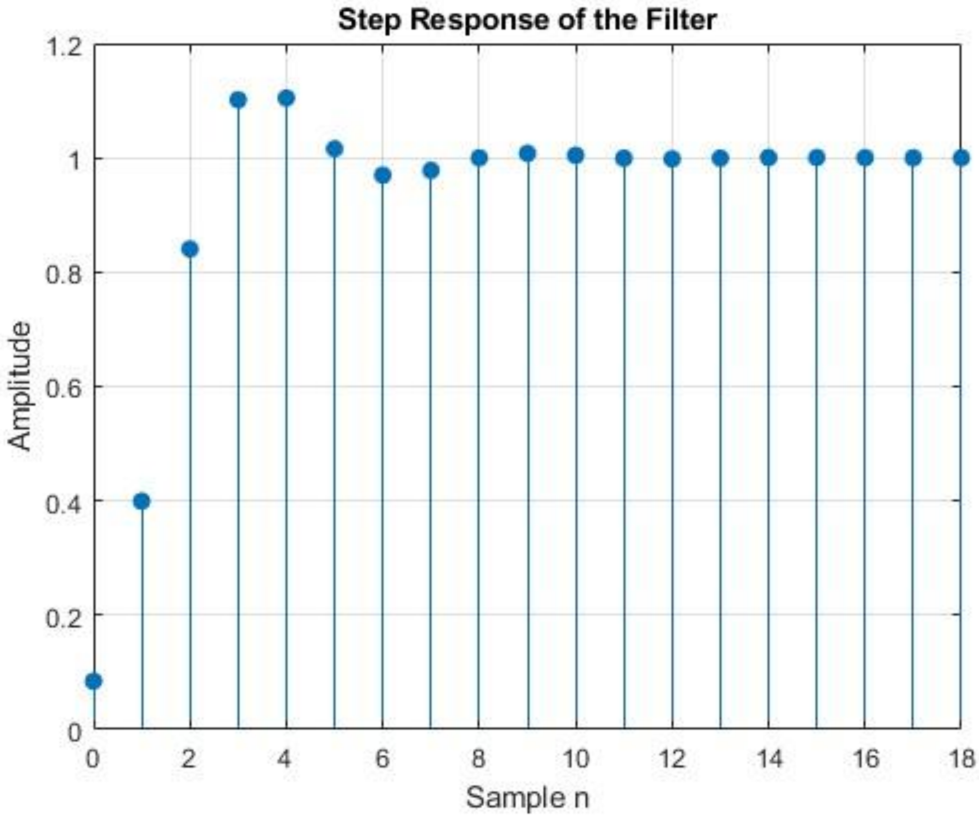
```
% [EOF]

fvtool(Hd);
```



C.


Impulse Response of the Filter
Amplitude
Sample n
0.5
0.4
0.3
0.2
0.1
0
-0.1
0
2
4
6
8
10
12
14
16
18

d.




```
function Hd = butter_lp3
%BUTTER_LP3 Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 24.1 and DSP System Toolbox 24.1.
% Generated on: 21-Nov-2025 15:27:40

% Butterworth Lowpass filter designed using FDESIGN.LOWPASS.

% All frequency values are in Hz.
Fs = 8000;  % Sampling Frequency

N  = 3;     % Order
Fc = 1500;  % Cutoff Frequency

% Construct an FDESIGN object and call its BUTTER method.
h  = fdesign.lowpass('N,F3dB', N, Fc, Fs);
Hd = design(h, 'butter');

% [EOF]

fvtool(Hd);

figure;
[impResp, n] = impz(Hd);  % Compute impulse response
stem(n, impResp, 'filled');  % Stem plot
title('Impulse Response of the Filter');
```

```
xlabel('Sample n');
ylabel('Amplitude');
grid on;

figure;
[stepResp, n] = stepz(Hd);  % Compute step response
stem(n, stepResp, 'filled');  % Stem plot
title('Step Response of the Filter');
xlabel('Sample n');
ylabel('Amplitude');
grid on;
```