UNIVERSITY OF MORATUWA, SRI LANKA

Faculty of Engineering

Department of Electronic and Telecommunication Engineering

Semester 3 (Intake 2020)

EN2063 – SIGNALS AND SYSTEMS

PROJECT - FIR and IIR Filter Design

Name: AHAMED M.B.S.A.

Registration Number: 200014B

19 January 2023

ABSTRACT

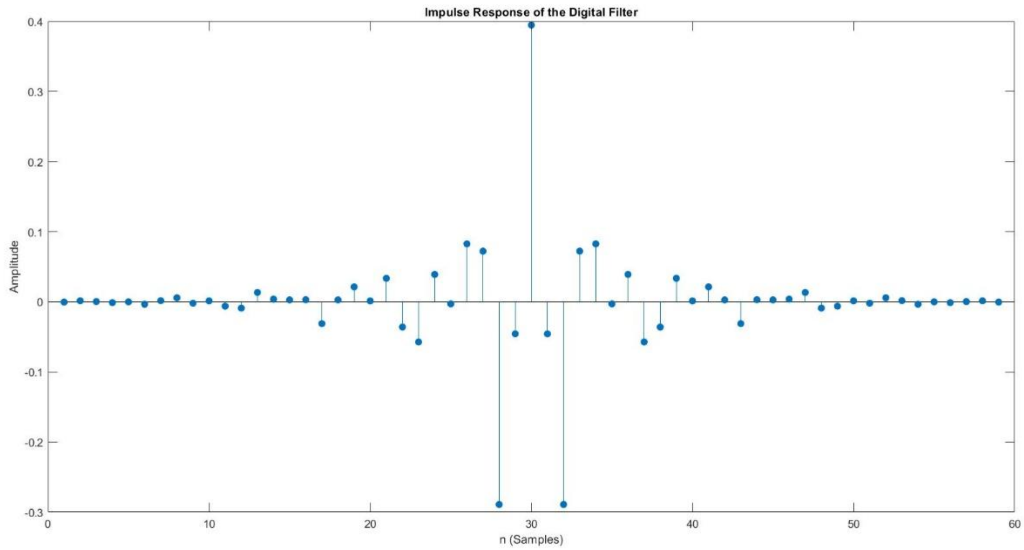
This report includes the design of FIR – Finite Impulse Response and IIR – Infinite Impulse Response bandpass digital filters for prescribed specifications. For FIR filters, the windowing method (in conjunction with the Kaiser window) is used whereas, for IIR filters, the bilinear transformation method is used. The objective of this project is to provide experience in the design of FIR and IIR digital filters for prescribed specifications. MATLAB R2017b is used to program and visualize the design and its results.

Parameter	Value
Maximum baseband ripple, $\tilde{\boldsymbol{A}}_p$	0.1 dB
Minimum stop band attenuation, $\tilde{\boldsymbol{A}}_a$	51 dB
Lower passband edge, Ω_{p1}	800 rad/s
Upper passband edge, Ω_{p2}	1300 rad/s
Lower stopband edge, Ω_{s1}	500 rad/s
Upper stopband edge, Ω_{s2}	1500dB rad/s
Sampling frequency, Ω_{sm}	3800 rad/s

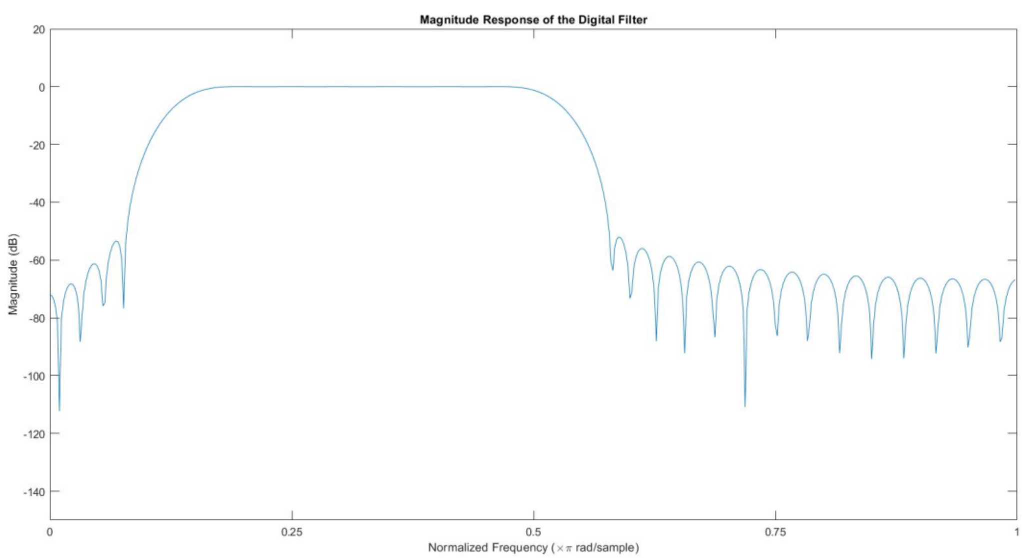
Table 1: Filter specifications.

1) Using the windowing method in conjunction with the Kaiser window, design an FIR bandpass digital filter that will satisfy the specifications given in Table 1.

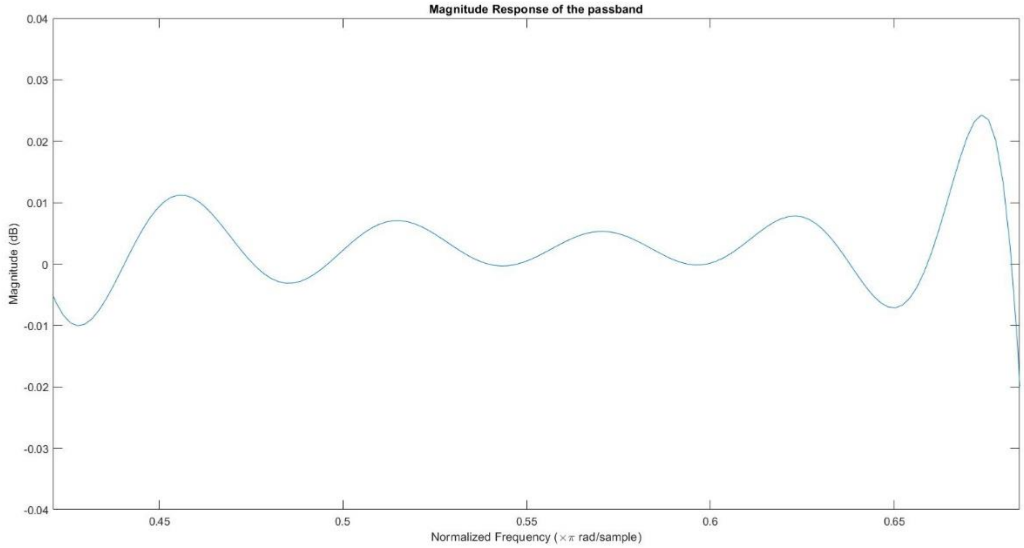
   a) Plot the impulse response.
   b) Plot the magnitude response of the digital filter for $\pi \leq \omega < \pi$ rad/sample.
   c) Plot the magnitude response for $\omega p1 \leq \omega \leq \omega p2$ (in the passband), where $\omega p1$ and $\omega p2$ are the passband edges in the discrete-time angular frequency domain.



Impulse response of the Digital Filter



Magnitude response of the digital filter for π ≤ ω < π rad/sample.



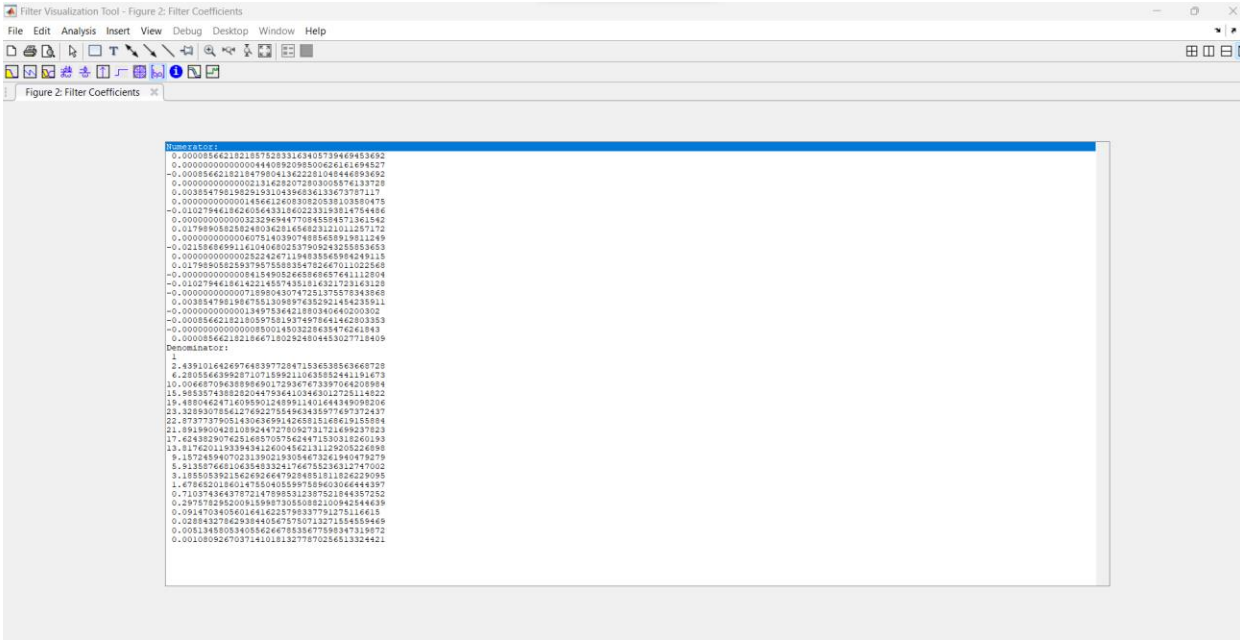
magnitude response for ωp1 ≤ ω ≤ ωp2 (in the passband)

2) Using the bilinear transformation method, design an IIR bandpass digital filter that will satisfy the specifications given in Table 1. Here, you need to first design an appropriate analog filter, and the required digital filter should be obtained by applying the bilinear transform to the transfer function of the analog filter. Note that, prewarping of frequencies is essential in order to obtain the required digital filter. The approximation method (or the type) of the IIR filter is determined according to your index number as follows. Let D be the remainder after dividing the digit C of your index number by 4. Then the approximation method should be selected as presented in Table 2.

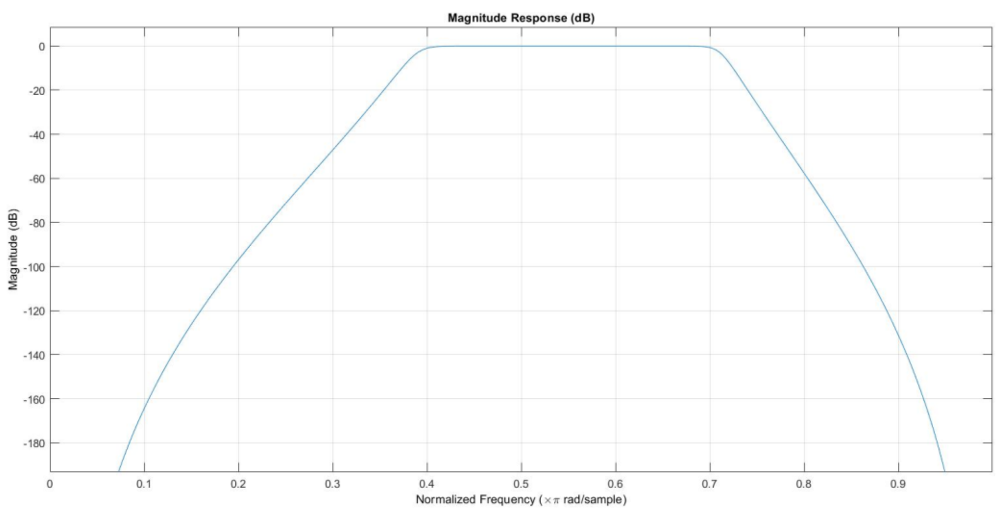
Table 2: IIR filter approximation method.

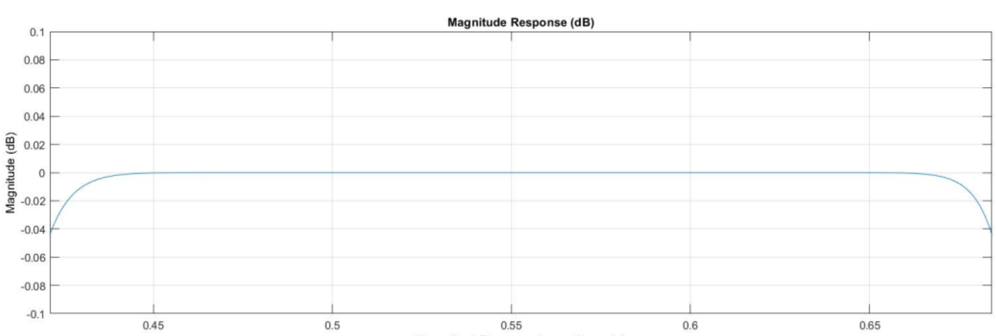
D	**Approximation method**
0	0 Butterworth
1	Chebyshev
2	Inverse-Chebyshev
3	Elliptic

a) Tabulate the coefficients of the transfer function of the IIR filter.
b) Plot the magnitude response of the digital filter for π ≤ ω < π rad/sample.
c) Plot the magnitude response for ωp1 ≤ ω ≤ ωp2 (in the passband), where ωp1 and ωp2 are the passband edges in the discrete-time angular frequency domain.



```
Numerator:
 0.000085662182185752833163405739469453692
 0.000000000000004440892098500626161694527
-0.000856621821847980413622281048446893692
 0.000000000000021316282072803005576133728
 0.00385479819829193104396836133673787117
 0.000000000000145661260830820538103580475
-0.010279461862605643318602233193814754486
 0.000000000000323296944770845584571361542
 0.017989058258248036281656823121011257172
 0.000000000000607514039074885658919811249
-0.021586869911610406802537909243255853653
 0.000000000000252242671194835565984249115
 0.0179890582593795755588354782667011022568
-0.000000000000841549052665868657641112804
-0.010279461861422145574351816321723163128
-0.000000000000718980430747251375578343868
 0.003854798198675513098976352921454235911
-0.00000000000013497536421880340640200302
-0.000856621821805975819374978641462803353
-0.00000000000000850014503228635476261843
 0.000085662182186671802924804453027718409
Denominator:
 1
 2.439101642697648397728471536538563668728
 6.280556639928710715992110635852441191673
10.006687096388986901729367673397064208984
15.985357438828204479364103463012725114822
19.488046247160959012489911401644349098206
23.328930785612769227554963435977697372437
22.873773790514306369914265815168619155884
21.891990042810892447278092731721699237823
17.624382907625168570575624471530318260193
13.817620119339434126004562131129205226898
 9.157245940702313902193054673261940479279
 5.913587668106354833241766755236312747002
 3.185505392156269266479284851811826229095
 1.678652018601475504055997589603066444397
 0.710374364378721478985312387521844357252
 0.297578295200915998730550882100942544639
 0.09147034056016416225798337791275116615
 0.028843278629384405675750713271554559469
 0.005134580534055626678535677598347319872
 0.001080926703714101813277870256513324421
```





3) Compare the order and the number of multiplications and additions required to process a sample by the two designed filters. Assume that the two filters are implemented using the difference equations, and the symmetry of coefficients can be exploited to reduce the number of multiplications.

When the number of computations for the two filter design strategies is compared, the Kaiser window requires fewer computations to design the filter than the Bilinear transformation method. This was accomplished by taking use of the symmetry of the coefficients of the FIR filter built using the Kaiser window method. The filter created with the Kaiser window approach has a higher order than the filter created with the Bilinear transformation method. When the magnitude responses of the two filters are compared, we can observe that the IIR filter has a substantially less passband ripple than the FIR filter. To compute an output sample, the FIR filter also requires a non-recursive approach, but the IIR filter requires a recursive one.

```
Discrete-Time IIR Filter (real)
-------------------------------
Filter Structure    : Direct-Form II Transposed
Numerator Length    : 21
Denominator Length  : 21
Stable              : Yes
Linear Phase        : No

Implementation Cost
Number of Multipliers            : 41
Number of Adders                 : 40
Number of States                 : 20
Multiplications per Input Sample : 41
Additions per Input Sample       : 40
```

```
Discrete-Time FIR Filter (real)
-------------------------------
Filter Structure    : Direct-Form II Transposed
Numerator Length    : 46
Denominator Length  : 1
Stable              : Yes
Linear Phase        : Yes (Type 2)

Implementation Cost
Number of Multipliers            : 46
Number of Adders                 : 45
Number of States                 : 45
Multiplications per Input Sample : 46
Additions per Input Sample       : 45
```

REFERENCES

• Signals, Systems, and Filters in Digital Signal Processing Kaiser Window, First Edition, Andreas Antoniou
• https://en.wikipedia.org/Kaiser Window
• https://en.wikipedia.org/

Appendix

MATLAB Code for the FIR filter

Impulse response of the Digital Filter Code

```
%{
Registartion_Number = 200014B
A = 0, B = 1, C = 0

Maximum passband ripple, A~p = 0.1 dB
Minimum stopband attenuation, A~a = 51 dB
Lower passband edge, ?p1 = 800 rad/s
Upper passband edge, ?p2 = 1300 rad/s
Lower stopband edge, ?s1 = 500 rad/s
Upper stopband edge, ?s2 = 1500 rad/s
Sampling frequency, ?sm = 3800 rad/s
%}

fsamp = 3800;
fcuts = [500 800 1300 1500];
mags = [0 1 0];
devs = [1./(10.^(51./20)) 1./(10.^(0.1./20))  1./(10.^(51./20))];

[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,fsamp);
n = n + rem(n,2);
hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale');

figure;
stem(hh,'filled');
title('Impulse Response of the Digital Filter');
xlabel('n (Samples)');
ylabel('Amplitude');
```

Magnitude of the Digital Filter Code

```
%{
Registartion_Number = 200014B
A = 0, B = 1, C = 0

Maximum passband ripple, A~p = 0.1 dB
Minimum stopband attenuation, A~a = 51 dB
Lower passband edge, ?p1 = 800 rad/s
Upper passband edge, ?p2 = 1300 rad/s
Lower stopband edge, ?s1 = 500 rad/s
```

```
Upper stopband edge, ?s2 = 1500 rad/s
Sampling frequency, ?sm = 3800 rad/s
%}

fsamp = 3800;
fcuts = [500 800 1300 1500];
mags = [0 1 0];
devs = [1./(10.^(51./20)) 1./(10.^(0.1./20))  1./(10.^(51./20))];

[n,Wn,beta,ftype] = kaiserord(fcutoffs,mags,devs,fsamp);
n = n + rem(n,2);
hh = fir1(n,Wn,ftype,kaiser(n+1,beta));

[H,f] = freqz(hh,1);

plot(f/pi,20*log10(abs(H)))
title("Magnitude Response of the Digital Filter")
ax = gca;
ax.YLim = [-150 20];
ax.XTick = 0:.25:1;

xlabel('Normalized Frequency (\times\pi rad/sample)')
ylabel('Magnitude (dB)')
n
```

Magnitude response for $\omega p1 \leq \omega \leq \omega p2$ (in the passband)

```
%{
Registartion_Number = 200014B
A = 0, B = 1, C = 0

Maximum passband ripple, A~p = 0.1 dB
Minimum stopband attenuation, A~a = 51 dB
Lower passband edge, ?p1 = 800 rad/s
Upper passband edge, ?p2 = 1300 rad/s
Lower stopband edge, ?s1 = 500 rad/s
Upper stopband edge, ?s2 = 1500 rad/s
Sampling frequency, ?sm = 3800 rad/s
%}

fsamp = 3800;
fcutoffs = [500 800 1300 1500];
mags = [0 1 0];
devs = [1./(10.^(51./20)) 1./(10.^(0.1./20))  1./(10.^(51./20))];

[n,Wn,beta,ftype] = kaiserord(fcutoffs,mags,devs,fsamp);
```

```
n = n + rem(n,2);
hh = fir1(n,Wn,ftype,kaiser(n+1,beta));

[H,f] = freqz(hh,1);

Wp1 = 800/1900;
Wp2 = 1300/1900;

plot(f/pi,20*log10(abs(H)))
title("Magnitude Response of the passband")
ax = gca;
ax.YLim = [-0.04 0.04];
ax.XLim = [Wp1 Wp2];

xlabel('Normalized Frequency (\times\pi rad/sample)')
ylabel('Magnitude (dB)')
```

MATLAB Code for the IIR filter

```
%specifications required for digital filter
fsamp = 3800/(2*pi);%Sampling frequency in Hz
%Passband and Stopband frequencies in samples/sec
%Analog specifications are in rads-1
Wp = [800 1300]/(fsamp);
Ws = [500 1500]/(fsamp);

%Specifications after prewarping
Wp = 2./(1./fsamp)*tan(Wp/2);
Ws = 2./(1./fsamp)*tan(Ws/2);

%Finding the corresponding analog filter

Rp = 0.1;%Passband ripple in dB
Rs = 51;%Stopband attenuation in dB
[n,Wn] = buttord(Wp,Ws,Rp,Rs,'s');
[n,d] = butter(n,Wn,'s');

%Using bilinear transform to find the filter coefficients for the digital
%filter
[nd,dd] = bilinear(n,d,fsamp);

%Visualizing the filter
fvtool(nd,dd)
```