INDIAN INSTITUTE of TECHNOLOGY, BOMBAY

DIGITAL SIGNAL PROCESSING

EE - 338

Filter Design Assignment

Name : Shreya Laddha

Roll Number : 180070054

November 11, 2020

Contents

1 Filter Specifications

Name – Shreya Laddha

Roll Number - 180070054

Filter Number – 63

Group Number – 42

Reviewer – Shreyan Jabade

2 Filter-1(Bandpass) Details

2.1 Un-normalized Discrete Time Filter Specifications

Since filter number is < 80 , m = 63 and passband will be monotonic

q(m) = greatest integer less than [0.1*63] = 6

r(m) = m - 10*q(m) = 63 - 10*6 = 3

BL(m) = 25 + 1.7*q(m) + 6.1*r(m) = 5 + 1.7*6 + 6.1*3 = 53.5

BH (m) = BL(m) + 20 = 33.5 + 20 = 73.5

The first filter is given to be a Band-Pass filter with passband from BL(m) kHz to BH (m) kHz. Therefore the specifications are :-

- Passband : 53.5 kHz to 73.5 kHz
- Transition Band : 4 kHz on either side of passband
- Stopband : 0-49.5 kHz and 77.5-165 kHz (∵ Sampling rate is 330 kHz)
- Tolerance : 0.15 in magnitude for both Passband and Stopband
- Passband Nature : Monotonic
- Stopband Nature : Monotonic

2.2 Normalized Digital Filter Specifications

Sampling Rate = 330 kHz

In the normalized frequency axis, sampling rate corresponds to 2π

Thus, any frequency (Ω) upto 165 kHz (Sampling rate/2) can be represented on the normalized axis(ω) as :-

$$\omega = \frac{\Omega * 2\pi}{\Omega_s(SamplingRate)}$$

Therefore the corresponding normalized discrete filter specifications are :-

- **Passband** : 0.324π to 0.4454π
- **Transition Band** : 0.024π on either side of passband
- **Stopband** : 0 - 0.3π and 0.47π - π
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband
- **Passband Nature** : Monotonic
- **Stopband Nature** : Monotonic

2.3 Analog filter specifications for Band-pass analog filter using Bilinear Transformation

The bilinear transformation is given as :-

$$\Omega = tan\ (\omega/2)$$

Applying the Bilinear Transformation to the frequencies at the band-edges, we get :-

ω	Ω
0.324π	0.558
0.4454π	0.8417
0.3π	0.5095
0.47π	0.9099
0	0
π	∞

Therefore the corresponding analog filter specifications for the same type of analog filter using the bilinear transformation are :-

- **Passband** : 0.558 (Ω_{P_1}) to 0.8417 (Ω_{P_2})
- **Transition Band** : 0.5095 to 0.558 and 0.8417 to 0.9099

- **Stopband** : 0 to 0.5095 (Ω_{S_1}) and 0.9099 (Ω_{S_2}) to ∞
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband
- **Passband Nature** : Monotonic
- **Stopband Nature** : Monotonic

2.4 Frequency Transformation & Relevant Parameters

We need to transform a Band-Pass analog filter to a Lowpass analog filter. We require two parameters in such a case. We can make use of the Bandpass transformation which is given as :-

$$\Omega_L = (\Omega^2 - \Omega_0^2) / B\Omega$$

The two parameters in the above equation are B and Ω_0. They can be determined using the specifications of the bandpass analog filter using the following relations :-

$$\Omega_0 = \sqrt{(\Omega_{P1} * \Omega_{P2})} = \sqrt{(1.620*5.729)} = 0.6853$$

$$B = \Omega_{P2} - \Omega_{P1} = 0.2837$$

Ω	Ω_L
0^+	-∞
0.5095 (Ω_{S_1})	1.453 ($\Omega_{L_{S1}}$)
0.558 (Ω_{P_1})	-1($\Omega_{L_{P1}}$)
0.6853 (Ω_0)	0
0.8417 (Ω_{P_2})	1 ($\Omega_{L_{P2}}$)
0.9099 (Ω_{S_2})	1.388 ($\Omega_{L_{S2}}$)
∞	∞

2.5 Frequency Transformed Lowpass Analog Filter Specifications

- **Passband Edge** : 1 (Ω_{L_P})
- **Stopband Edge** : min(-$\Omega_{L_{S_1}}$,$\Omega_{L_{S_2}}$) = min(1.453, 1.388) = 1.388 (Ω_{L_S})
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband
- **Passband Nature** : Monotonic

- **Stopband Nature** : Monotonic

2.6 Butterworth Filter: Analog Lowpass Transfer Function

Analog Filter which has an *monotonic passband* and a *monotonic stopband* is required. This is done using the **Butterworth** approximation. Since the tolerance (δ) in both passband and stopband is 0.15, we define :

$$D_1 = \frac{1}{(1-\delta)^2} - 1 = \frac{1}{0.85^2} - 1 = 0.3841$$

$$D_2 = \frac{1}{\delta^2} - 1 = \frac{1}{0.15^2} - 1 = 43.444$$

Now using the inequality on the order N of the filter for the Butterworth Approximation we get ($\Omega s = 1.388$ and $\Omega p = 1$) :-

$$N_{min} = \left\lceil \log\sqrt{(D2/D1)} \,/\, \log(\Omega s/\Omega p) \right\rceil$$

$$N_{min} = 8$$

The cut-off frequency (Ω_c) of the Analog LPF should satisfy the following constraint :-

$$\Omega_P / (D_1^{1/2N}) \leq \Omega_c \leq \Omega_S / (D_2^{1/2N})$$

$$1.0616 \leq \Omega_c \leq 1.096$$

Taking $\Omega_c = 1.0788$ (arithematic average of the two limits). Now, the poles of the transfer function can be obtained by solving the equation :

$$1+\left(\frac{s}{j\Omega_c}\right)^{2N} = 1+\left(\frac{s}{j\,1.0788}\right)^{16} = 0$$

Solving for the roots(using Wolfram) we get :-

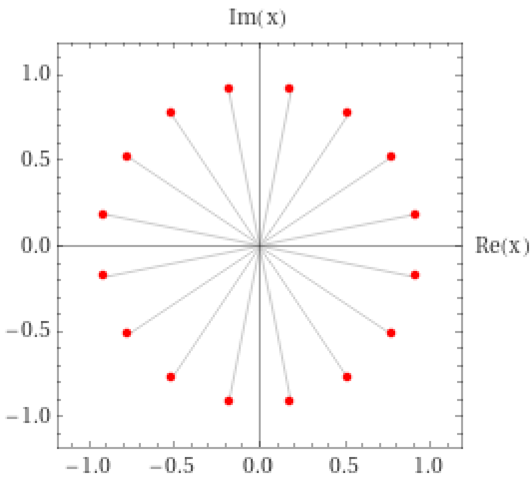

Figure 1: Poles of Magnitude Plot of Analog LPF

Above figure shows the poles of the Magnitude Plot of the Transfer Function. In order to get a stable Analog LPF, only the poles lying in the Left Half Plane are included in the Transfer Function. (The poles are symmetric about origin and we can pick one from each pair to be a part of our Transfer Function). These poles are (printed in command window in matlab) -

```
>> Butterworth_Filter_1
p1 = -2.104634e-01+ i 1.058071e+00
p2 = -2.104634e-01+ i -1.058071e+00
p3 = -5.993492e-01+ i 8.969894e-01
p4 = -5.993492e-01+ i -8.969894e-01
p5 = -8.969894e-01+ i 5.993492e-01
p6 = -8.969894e-01+ i -5.993492e-01
p7 = -1.058071e+00+ i 2.104634e-01
p8 = -1.058071e+00+ i -2.104634e-01
```

Using the above poles which are in the left half plane we can write the Analog Lowpass Transfer function as :-

$$H_{analog,LPF}(s_L) = \frac{(\Omega_c)^N}{(s_L - p_1)\,(s_L - p_2)\,(s_L - p_3)\,(s_L - p_4)\,(s_L - p_5)\,(s_L - p_6)\,(s_L - p_7)\,(s_L - p_8)}$$

2.7 Analog Bandpass Transfer Function

The transformation equation is given by :-

$$s_L = \frac{s^2 + \Omega_o^2}{Bs}$$

Substituting the values of the parameters B (0.2837) and Ω_0 (0.6853), we get :-

$$S_L = s^2 + 0.469 / 0.2837s$$

Thus we get the transfer function as:

$$H_{analog,BPF}(s) = \frac{\Omega_c^8 (Bs)^8}{\prod_{i=0}^{7}(s^2 - p_i Bs + \Omega_o^2)}$$

Note that in this equation, the zero of is at s = 0 of order 8. And in the denominator there are 8 product terms which are second order polynomials. Thus factorizing each of the second order polynomials, we expect to get a total of 16 poles, provided there is no overlapping of the poles. The poles can be calculated as (here p_i are poles of analog LPF) -

$$Poles = \frac{p_i B \pm \sqrt{(p_i B)^2 - 4\Omega_o^2}}{2}$$

Thus from the new poles obtained, we can write our effective band-pass filter transfer function as (here p_i are the poles of analog BPF):

$$H_{analog,BPF}(s) = \frac{\Omega_c^8 (Bs)^8}{\prod_{i=0}^{15}(s - p_i)}$$

Obtained Numerator(s) of transfer function= $10^{-4} * 0.7621\ s^8$

Obtained Denominator(s) = $s^{16} + 1.5668\ s^{15} + 4.9884\ s^{14} + 5.78\ s^{13} + 9.8749\ s^{12} + 8.7969\ s^{11} + 10.3202\ s^{10} + 7.1603\ s^9 + 6.2775\ s^8 + 3.3662\ s^7 + 2.2809\ s^6 + 0.914\ s^5 + 0.4824\ s^4 + 0.1327\ s^3 + 0.0539\ s^2 + 0.008\ s^1 + 0.0024$

2.8 Discrete Time Filter Transfer Function

To transform the analog domain transfer function into the discrete domain, we need to make use of the Bilinear Transformation which is given as :-

$$s = \frac{1 - z^{-1}}{1 + z^{-1}}$$

Using above equation we get $H_{discrete,BP\ F}\,(z)$ from $H_{analog,BP\ F}\,(s)$ as :-

$$H(z) = \frac{(\Omega_c B)^8 (1 - z^{-1})^8 (1 + z^{-1})^8}{\prod_{i=0}^{15} ((1 - p_i) - (1 + p_i) z^{-1})}$$

there are two zeros of the transfer function. at z = 1, and the other one at z = -1. Both the zeros have the order 8. The poles are (p_i are poles of analog BPF) -

$$z_i = \frac{1 + p_i}{1 - p_i}$$

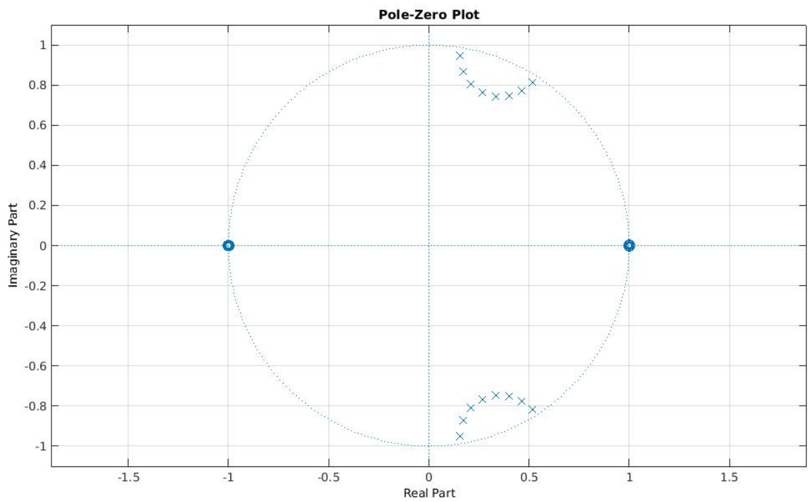

Figure2: Pole – zero plot of discrete BPF filter

We can observe that all the poles lie within the unit circle. Hence the system we have designed is stable.

We can get the frequency response of the discrete time band-pass system by putting the $z = e^{jw}$

$$H(\omega) = \frac{(\Omega_c B)^8 (1 - e^{-j\omega})^8 (1 + e^{-j\omega})^8}{\prod_{i=0}^{15} ((1 - p_i) - (1 + p_i) e^{-j\omega})}$$

The final rational transfer function obtained is given as:

$$H(z) = \frac{(\Omega_c B)^8 (1 - z^{-2})^8}{\prod_{i=0}^{15} ((1 - p_i) - (1 + p_i) z^{-1})}$$

We get Numerator(z) = 1.0e-04 * (0.0121 – 0.0968 z^{-2} + 0.3387 z^{-4} - 0.6774 z^{-6} + 0.8467 z^{-8} -0.6774 z^{-10} + 0.3387 z^{-12} - 0.0968 z^{-14} + 0.0121 z^{-16})

and Denominator(z) = 1 – 5.01 z^{-1} 16.9091 z^{-2} -39.8758 z^{-3} + 75.6550 z^{-4} – 116.1858 z^{-5} + 151.1899 z^{-6} - 166.1997 z^{-7} + 157.6139 z^{-8} – 127.6136 z^{-9} + 89.1263 z^{-10} – 52.5640 z^{-11} + 26.2606 z^{-12} – 10.6111 z^{-13} + 3.4494 z^{-14} – 0.7823 z^{-15} + 0.1199 z^{-16}

2.9 FIR Filter Transfer Function using Kaiser Window

The tolerance in both the stopband and passband is given to be 0.15.

Therefore δ = 0.15 and we get the minimum stopband attenuation to be :-

$$A = -20 \log(0.15) = 16.4782\ dB$$

Since A < 21, we get β to be 0 where β is the shape parameter of Kaiser window.

Now to estimate the window length required, we use the empirical formula for the lower bound on the window length.

$$2N + 1 >= 1 + (A - 8 / 2.285 * \Delta\omega_T)$$

Here $\Delta\omega_T$ is the minimum transition width. In our case, the transition width is the same on either side of the passband.

$$\Delta\omega_T = \frac{2\pi}{f_s} \times 4kHz = \frac{8\pi}{330}$$

$$N \geq \frac{A - 8}{4.57 \Delta\omega_T}$$

$$N_{min} = 25$$

The above equation gives a loose bound on the window length when the tolerance is not very stringent. Choosing N=45 after checking various N. Also, since β is 0, the window is a rectangular window.

In order to find the time domain coefficients, first the ideal impulse response samples for the same length as that of the window are generated. Then, the Kaiser Window is generated using the MATLAB function and applied on the ideal impulse response samples. For generating the ideal impulse response a separate function was made to generate the impulse response of Low-Pass filter. It took the cutoff value and the number of samples as input argument. The band pass impulse response samples are then generated as the difference between two low-pass filters. Since there are 69 coefficients, they are listed here for brevity.

```
>> Bandpass_FIR1
  Columns 1 through 20

   -0.0180   -0.0108    0.0094    0.0145    0.0017   -0.0067   -0.0022   -0.0008   -0.0080   -0.0050    0.0139    0.0211   -0.0029   -0.0292   -0.0183    0.0174    0.0286    0.0034   -0.0174

  Columns 21 through 40

    0.0010   -0.0081   -0.0074    0.0251    0.0426   -0.0071   -0.0746   -0.0520    0.0587    0.1127    0.0156   -0.1190   -0.1054    0.0511    0.1457    0.0511   -0.1054   -0.1190    0.0156

  Columns 41 through 60

    0.0587   -0.0520   -0.0746   -0.0071    0.0426    0.0251   -0.0074   -0.0081    0.0010   -0.0092   -0.0174    0.0034    0.0286    0.0174   -0.0183   -0.0292   -0.0029    0.0211    0.0139

  Columns 61 through 69

   -0.0080   -0.0008   -0.0022   -0.0067    0.0017    0.0145    0.0094   -0.0108   -0.0180
```

Figure3 : Values of the FIR band-pass impulse sequence

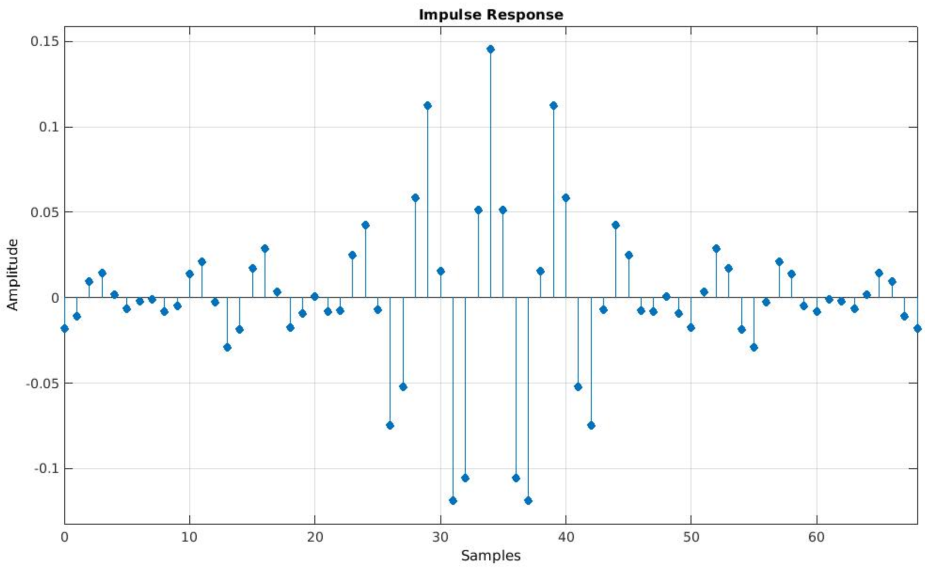

Figure4 - Plot of the FIR band-pass impulse sequence

2.10 Comparison between FIR and IIR realizations

We have realized a bandpass filter in terms of finite impulses. In case of FIR, the number of impulses are 69. Thus, we will have terms from 1 to z^{-68}. This filter is definitely stable and causal. As we have shifted the impulses, we expect to get a linear phase response in the pass-band region.

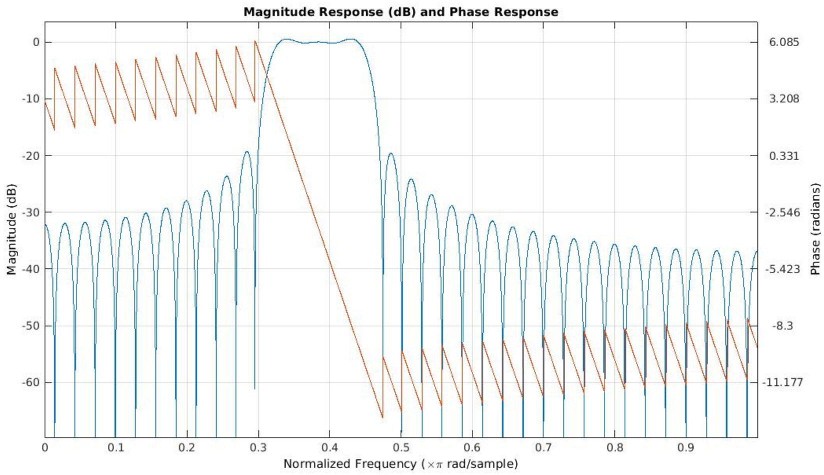

Figure5 : Frequency response of FIR band-pass filter

The phase response is linear as shown. This was not the case for the Butterworth implementation, which had a slight non-linearity.

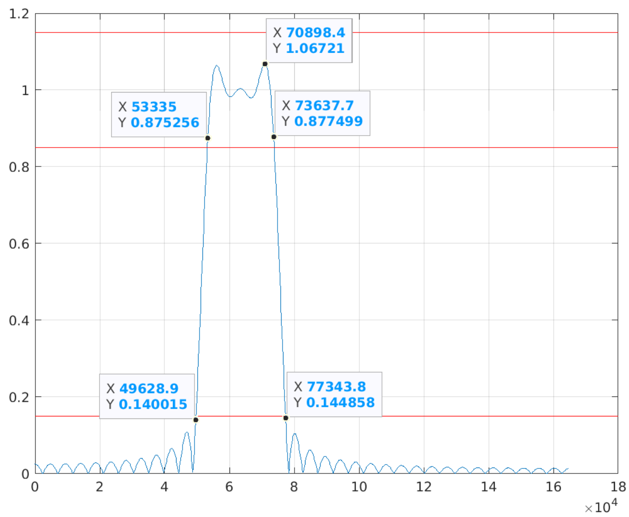

Figure6 : Magnitude response of FIR band-pass filter

For the Butterworth implementation, we had a order of 16 in both numerator and denominator. Thus around 30 delay stages are expexted for IIR case. For FIR, we need terms till z^{-68}, thus it needs 68 delay stages. Clearly, the resource demand for the FIR realization is much more than the IIR case.

Also, for FIR filter, we need to adjust the value of N to meet our design specifications, whereas no such tuning is required in IIR case, which is a disadvantage of FIR filters.

3 Filter-2(Bandstop) Details

3.1 Un-normalized Discrete Time Filter Specifications

Filter Number = 63

Since filter number is <80 , m = 63 and passband will be equiripple

B_L(m) = 25 + 1.9*q(m) + 4.1*r(m) = 25 + 1.9*6 + 4.1*3 = 48.7

B_H (m) = B_L(m) + 20 = 48.7 + 20 = 68.7

The second filter is given to be a ***Band-Stop*** filter with stopband from B_L(m) kHz to B_H (m) kHz. Therefore the specifications are :-

- **Stopband** : 48.7 kHz to 68.7 kHz
- **Transition Band** : 4 kHz on either side of stopband
- **Passband** : 0 to 44.7 kHz and 72.7 to 130 kHz (∵ Sampling rate is 260 kHz)
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband
- **Passband Nature** : Equiripple
- **Stopband Nature** : Monotonic

3.2 Normalized Digital Filter Specifications

Sampling Rate = 260 kHz

In the normalized frequency axis, sampling rate corresponds to 2π

Thus, any frequency(Ω) upto 50 kHz($\frac{SamplingRate}{2}$) can be represented on the normalized axis(ω) as :-

$$\omega = \frac{\Omega * 2\pi}{\Omega_s(SamplingRate)}$$

Therefore the corresponding normalized discrete filter specifications are :-

- **Stopband** : 0.375π to 0.528π
- **Transition Band** : 0.03π on either side of stopband
- **Passband** : 0 to 0.344π and 0.559π *to* π
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband

- **Passband Nature** : Equiripple
- **Stopband Nature** : Monotonic

3.3 Analog filter specifications for Band-stop analog filter using Bilinear Transformation

The bilinear transformation is given as :-

$$\Omega = tan\ (\omega/2)$$

Applying the Bilinear transform to the frequencies at the band-edges, we get :-

ω	Ω
0.375π	0.668
0.344π	0.6
0.528π	1.092
0.559π	1.205
0	0
π	∞

Therefore the corresponding analog filter specifications for the same type of analog filter using the bilinear transformation are :-

- **Stopband** : 0.668 (Ω_{S_1}) to 1.092 (Ω_{S_2})
- **Transition Band** : 0.6 to 0.668 & 1.092 to 1.205
- **Passband** : 0 to 0.6 (Ω_{P_1}) and 1.205 (Ω_{P_2}) to ∞
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband
- **Passband Nature** : Equiripple
- **Stopband Nature** : Monotonic

3.4 Frequency Transformation & Relevant Parameters

We need to transform a BandStop analog filter to a Lowpass analog filter. We require two parameters in such a case. We can make use of the Bandstop transformation which is given as :-

$$\Omega_L = \frac{B\Omega}{\Omega_0^2 - \Omega^2}$$

The two parameters in the above equation are B and Ω_0. They can be determined using the specifications of the bandpass analog filter using the following relations :-

$$\Omega_0 = \sqrt{(\Omega_{P1} * \Omega_{P2})} = 0.85$$

$$B = \Omega_{P2} - \Omega_{P1} = 0.605$$

Ω	Ω_L
0^+	0^+
$0.6(\Omega_{P_1})$	$+1\ (\Omega_{L_{P1}})$
$0.668(\Omega_{S_1})$	$+1.463(\Omega_{L_{S\,1}})$
$0.85(\Omega^-{}_0)$	∞
$0.85(\Omega^+{}_0)$	$-\infty$
$1.092(\Omega_{S_2})$	$-\ 1.406\ (\Omega_{L_{S\,2}})$
$1.205(\Omega_{P_2})$	$-\ 1\ (\Omega_{L_{P\,2}})$
∞	0^-

3.5 Frequency Transformed Lowpass Analog Filter Specifications

- **Passband Edge** : $1\ (\Omega_{L_P})$
- **Stopband Edge** : $\min(\Omega_{L_{S_1}}, -\Omega_{L_{S_2}}) = \min(1.463, 1.406) = 1.406\ (\Omega_{L_S})$
- **Tolerance** : 0.15 in magnitude for both Passband and Stopband
- **Passband Nature** : Equiripple
- **Stopband Nature** : Monotonic

3.6 Analog Lowpass Transfer Function

Analog Filter which has a equiripple passband and monotonic stopband is required. This is done using the **Chebyshev** approximation. Since the tolerance (δ) in both passband and stopband is 0.15, we have :-

$$D_1 = \frac{1}{(1-\delta)^2} - 1 = \frac{1}{0.85^2} - 1 = 0.3841$$

$$D_2 = \frac{1}{\delta^2} - 1 = \frac{1}{0.15^2} - 1 = 43.44$$

Now choosing the parameter ϵ of the Chebyshev filter to be $\sqrt{D1}$, we get the minimum value of N as:

$$N_{min} = \lceil \text{acosh}\,(\sqrt{(D2/D1)}) \,/\, \text{acosh}(\Omega_{L_S}/\Omega_{L_P}) \rceil$$

$$N_{min} = \lceil 3.499 \rceil = 4$$

Now, the poles of the transfer function can be obtained by solving the equation : -

$$1 + D_1 \cosh^2(N_{min} \cosh^{-1}(s/j)) = 1 + 0.3841 \cosh^2(4 \cosh^{-1}(s/j)) = 0$$

Solving for the roots (using wolfram) we get : -

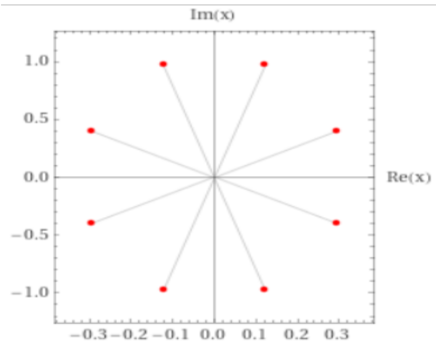

Figure 7: Poles of Magnitude Plot of Analog LPF

The plot shows the poles of the Magnitude Plot of the Transfer Function. In order to get a stable Analog LPF, only the poles lying in the Left Half Plane are included in the Transfer Function.The poles are -

```
>> Chebyshev_Filter_2
p1 = -1.221623e-01+ i 9.698117e-01
p2 = -1.221623e-01+ i -9.698117e-01
p3 = -2.949259e-01+ i 4.017091e-01
p4 = -2.949259e-01+ i -4.017091e-01
```

As in our case N = even, we will have a normalizing factor of $1/\sqrt{(1+\varepsilon^2)}$ in the denominator.

$\varepsilon = \sqrt{D_1}$ Thus, using the 4 poles as stated above, we can write our analog low-pass transfer function as:

$$H_{analog,LPF}(s_L) = \frac{\prod_{i=0}^{3}(-p_i)}{\sqrt{1+D_1} \times \prod_{i=0}^{3}(s_L - p_i)}$$

DC gain of $1/\sqrt{(1+\varepsilon^2)}$ is taken as N=4. In Chebyshev, the order is much lower as compared to the Butterworth case. But, the trade-off is that we get more non-linear phase response which is undesirable in many applications.

3.7 Analog Bandstop Transfer Function

The transformation equation is given by - $\Omega_0^2 + s^2$

$$s_L = Bs / (\Omega_0^2 + s^2)$$

Substituting the values of the parameters B (1.298) and Ω_0 (1.32566), we get :-

$$s_L = \frac{1.298s}{1.7574 + s^2}$$

To get poles and zeroes, we can also write (p_i are the poles of analog LPF obtained in previous section) -

$$H_{analog,BSF}(s) = \frac{1}{\sqrt{1+D_1}} \times \frac{(s^2 + \Omega_o^2)^4}{\prod_{i=0}^{3}(s^2 - \frac{Bs}{p_i} + \Omega_o^2)}$$

The zeroes are at $s = \pm j\Omega_0$ both of order 4. We get a total of 8 poles, which can be found using the following function -

$$Poles = \frac{\frac{B}{p_i} \pm \sqrt{(\frac{B}{p_i})^2 - 4\Omega_o^2}}{2}$$

and we can write our transfer function as (here p_i are the poles obtained above, of bandstop filter)-

$$H_{analog,BSF}(s) = \frac{1}{\sqrt{1+D_1}} \times \frac{(s^2 + \Omega_o^2)^4}{\prod_{i=0}^{7}(s - p_i)}$$

Substituting values into $H_{analog,LPF}$ (s_L), we get $H_{analog,BSF}$ (s) = Num(s) /Den(s)

where Num(s) = $0.85s^8 + 2.4582\ s^6 + 2.6658\ s^4 + 1.2849\ s^2 + 0.2322$

and Den(s) = $s^8 + 1.5949\ s^7 + 4.9797\ s^6 + 4.2425\ s^5 + 6.7244\ s^4 + 3.0673\ s^3 + 2.6030s^2 + 0.6027\ s + 0.2732$

The coefficients of odd powers of s in N(s) are all 0.

3.8 Discrete Time Filter Transfer Function

To transform the analog domain transfer function into the discrete domain, we need to make use of the Bilinear Transformation which is given as :-

$$s = \frac{1 - z^{-1}}{1 + z^{-1}}$$

Using above equation we get $H_{discrete,BSF}$ (z) = N (z) / D(z) =

$$H(z) = \frac{1}{\sqrt{1+D_1}} \times \frac{\{(1+j\Omega_o)+(j\Omega_o-1)z^{-1}\}^4\{(1-j\Omega_o)-(j\Omega_o+1)z^{-1}\}^4}{\prod_{i=0}^{7}((1-p_i)-(1+p_i)z^{-1})}$$

The p_i are poles of analog BSF transfer function. The form of the final discrete time band stop transfer function is similar to the one we obtained

in the case of the band pass filter. The zeroes of this system are both of 4 order (total 8 zeroes) given as -

$$zeros_{set1} = \frac{1-j\Omega_o}{1+j\Omega_o}$$

$$zeros_{set2} = \frac{1+j\Omega_o}{1-j\Omega_o}$$

The poles of the system are -

$$z_i = \frac{1+p_i}{1-p_i}$$

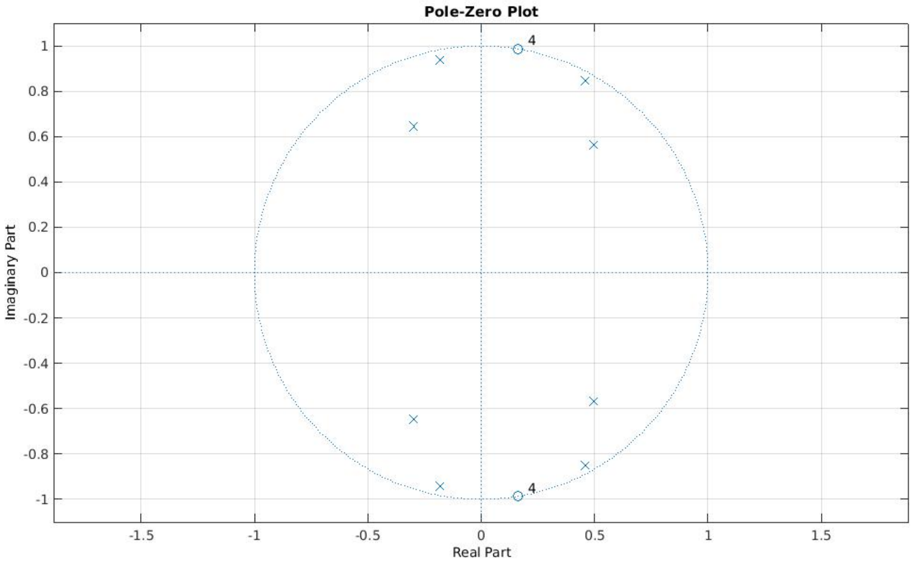

Figure8 : Poles and Zeros of the Transfer Function of the Filter

Values of the poles p_i are known and also the Ω_0 and B. Also, the denominator has poles in terms of complex conjugate pairs, thus the denominator can be expressed in terms of real coefficients of z^{-1} .

Hence, this system can be implemented using a finite number of delays. Thus, we have obtained a rational, stable and realizable system.

After subsituting values, we get

$N(z) = 0.2986 – 0.3841\ z^{-1} + 1.3796\ z^{-2} – 1.1919\ z^{-3} + 2.1653\ z^{--4} – 1.1919\ z^{-5} + 1.3796\ z^{-6} - 0.3841\ z^{-7} + 0.2986\ z^{-8}$

and $D(z) = 1- 0.9417\ z^{-1} + 2.2015\ z^{-2} – 1.5159\ z^{-3} + 2.1383\ z^{-4} – 0.9707\ z^{-5} + 0.9143\ z^{-6} -0.2797\ z^{-7} + 0.2421\ z^{-8}$

3.9 FIR Filter Transfer Function using Kaiser Window

The tolerance in both the stopband and passband is given to be 0.15.

Therefore $\delta = 0.15$ and we get the minimum stopband attenuation to be :-

$$A = -20\ \log(0.15) = 16.4782\ dB$$

Since A < 21, we get β to be 0 where β is the shape parameter of Kaiser window.

Now to estimate the window length required, we use

$$2N + 1 >= 1 + (A – 8 / 2.285 * \Delta\omega T)$$

Here $\Delta\omega_T$ is the minimum transition width.

$$N_{min} = 20$$

This gives a loose bound on the window length when the tolerance is not very stringent. Experimentally, it was found that a window length of 33 is required. Also, since β is 0, the window is actually a rectangular window.

In order to find the time domain coefficients, first the ideal impulse response samples for the same length as that of the window are generated. The Kaiser Window was generated using the MATLAB function and applied on the ideal impulse response samples. For generating the ideal impulse response a separate function was made to generate the impulse response of Low-Pass filter. It took the cutoff value and the number of samples as input argument. The band-stop impulse response samples were generated as the difference between three low-pass filters (all-pass - bandpass). Since there are 63 coefficients, they are listed here for brevity.

```
>> Bandstop_FIR2
  Columns 1 through 20

   -0.0092   -0.0020    0.0182    0.0095   -0.0195   -0.0158    0.0128    0.0140   -0.0035   -0.0021   -0.0004   -0.0153   -0.0058    0.0288    0.0192   -0.0302   -0.0298    0.0191    0.0259

  Columns 21 through 40

   -0.0022    0.0007   -0.0357   -0.0202    0.0716    0.0639   -0.0872   -0.1194    0.0714    0.1659   -0.0275    0.8160   -0.0275    0.1659    0.0714   -0.1194   -0.0872    0.0639    0.0716

  Columns 41 through 60

   -0.0357    0.0007   -0.0022   -0.0043    0.0259    0.0191   -0.0298   -0.0302    0.0192    0.0288   -0.0058   -0.0153   -0.0004   -0.0021   -0.0035    0.0140    0.0128   -0.0158   -0.0195

  Columns 61 through 63

    0.0182   -0.0020   -0.0092
```

Figure 9: Values of FIR band-stop impulse response

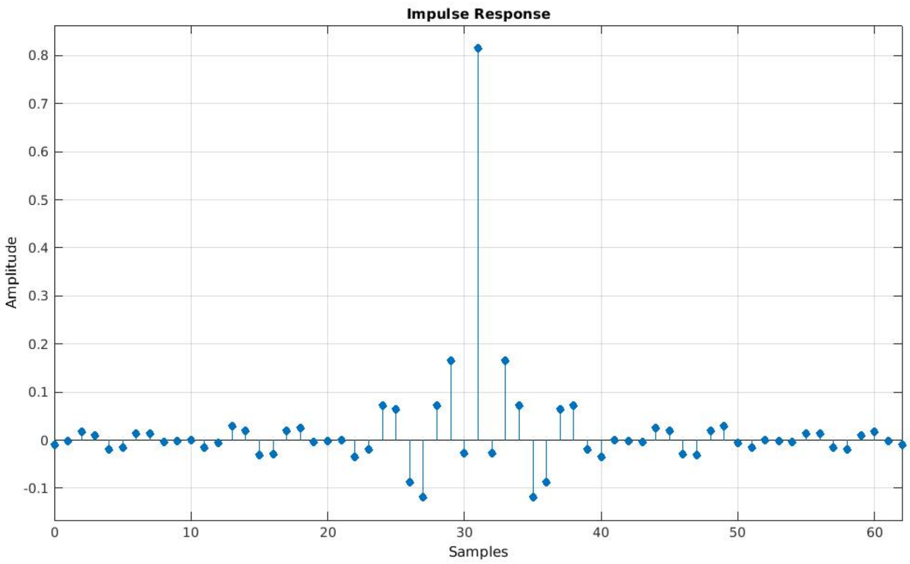

Figure 10: Plot of the FIR band-stop impulse sequence

3.10 Comparison between FIR and IIR realizations:

We have realized a band-stop filter in terms of both infinite and finite impulses. In FIR case, the number of impulses are 63. Hence for the transfer function, we will have terms from 1 to z^{-62} .The filter is stable as all poles lie within the unit cycle and causal. As we have shifted the impulses, we expect to get a linear phase

response in the stop-band region. Figure below shows the response.

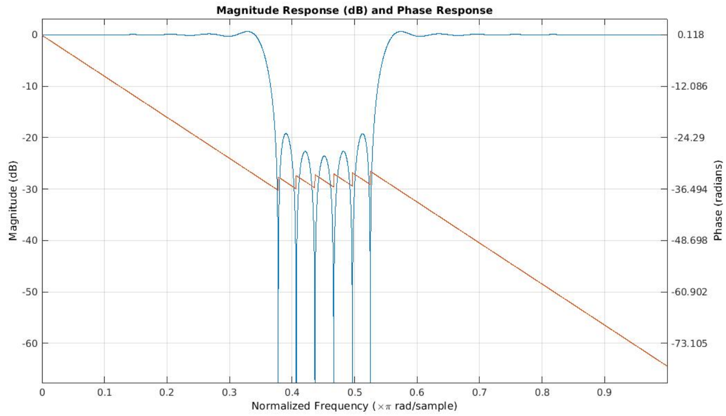

Figure 11: Frequency response of FIR band-stop filter

As seen in the figures, the phase is linear which was not the case in Chebyshev. There was an order of 8 in both numerator and denominator. Thus, around 14 delay stages are expected. However, the FIR design had order 63. Thus, it needs 62 delay stages. Hence, the resource demand for the FIR realization is much more than the IIR case.

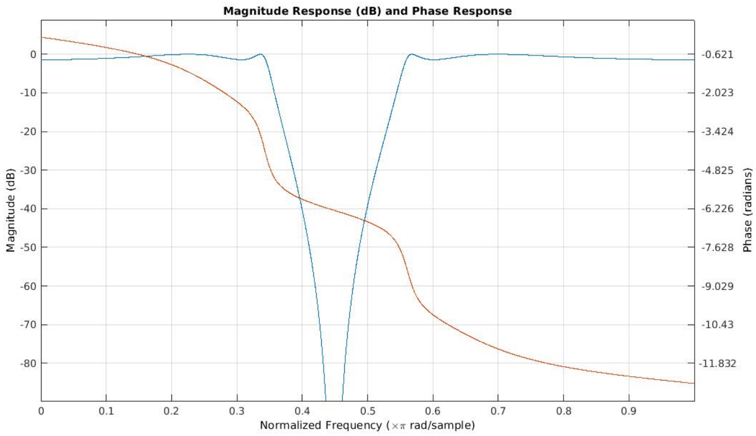

Figure 12: Magnitude response of IIR band-stop filter

The FIR design requires choosing N while implementation. This is not required in the IIR case

4 MATLAB Plots

4.1 Filter 1 - Bandpass

4.1.1 IIR Filter - Butterworth

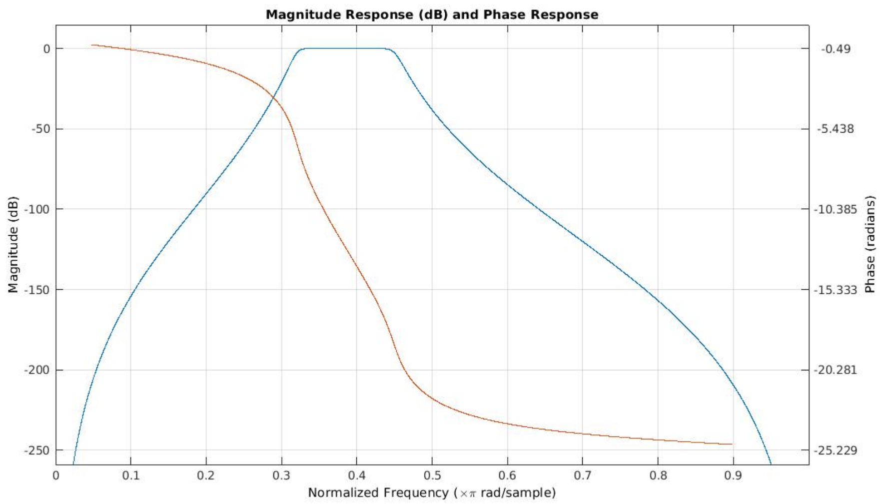

Figure 13: Frequency Response

From the above plot, it can be seen that the passband tolerance and stopband attenuation have been satisfied. Also, the **phase response** is **not linear**.

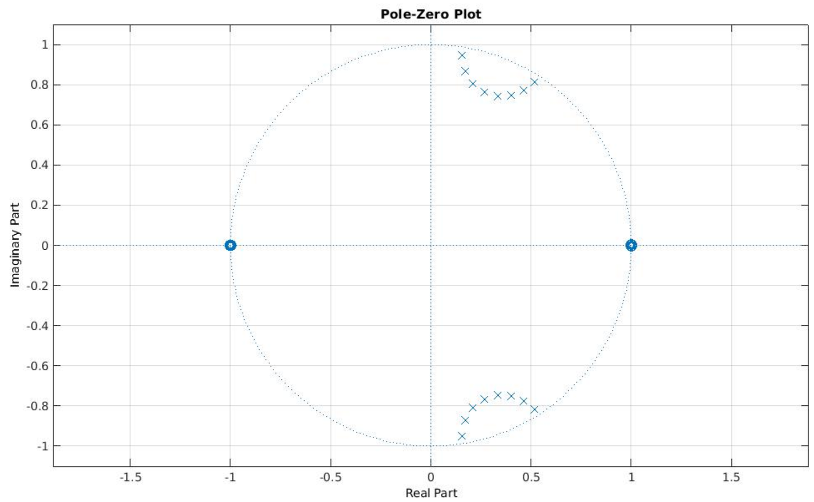

Figure 14: Pole-Zero map (all poles within unit circle, hence stable)

The zeroes are the circles(dots) and the poles are the crosses in the above pole-zero plot. As seen in the figure, all of the crosses lie inside the unit circle. This leads to the conclusion that it is indeed stable as required.

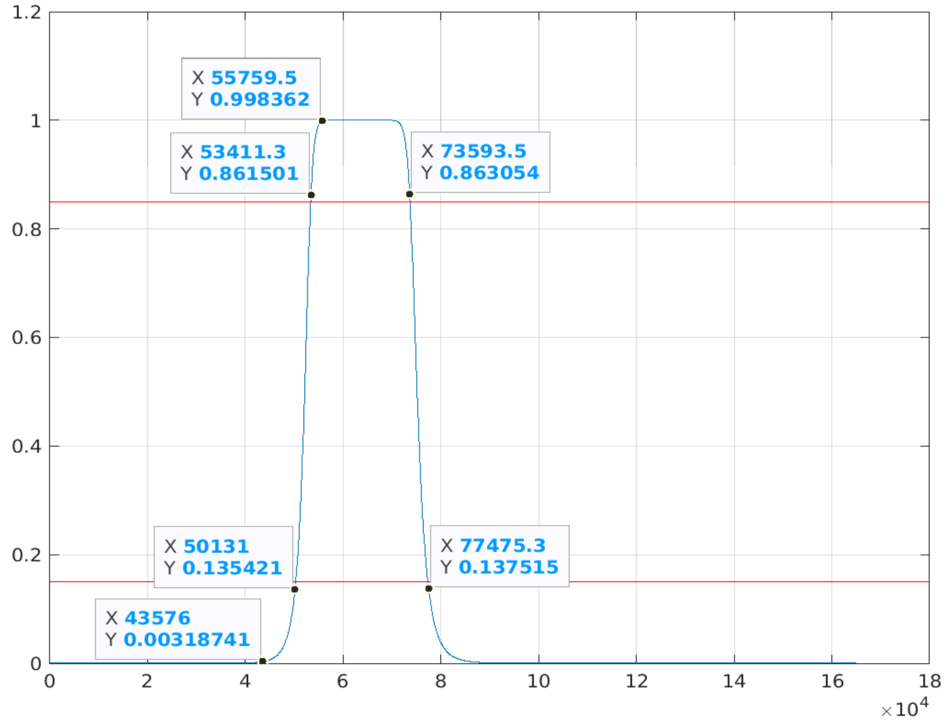

Figure 15: Magnitude Plot

In the above plot, the band edge frequencies have been marked. From the magnitude at these frequencies it can be seen that the specifications required in the passband and the stopband have been met satisfactorily.

4.1.2 FIR Filter

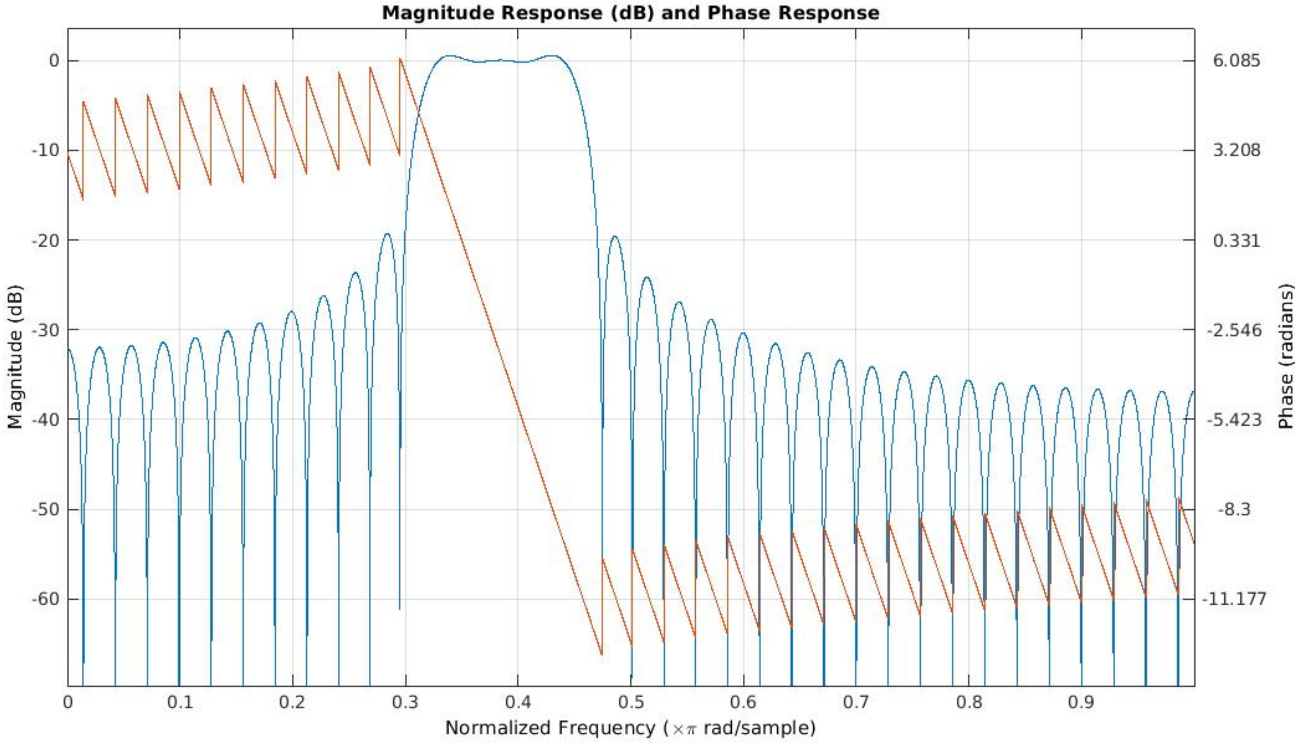

Figure 16: Frequency Response

From the above plot, it is verified that the passband tolerance and stopband attenuation have been satisfied. It can be seen that the FIR Filter is indeed giving us a **Linear Phase** response which is desired.

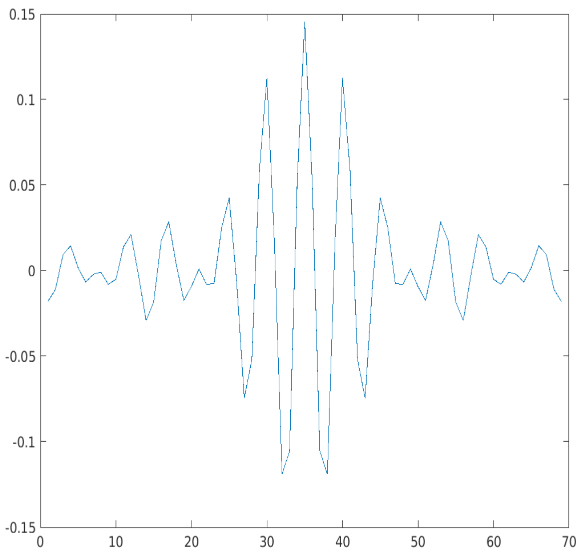

Figure 17: Time Domain Sequence

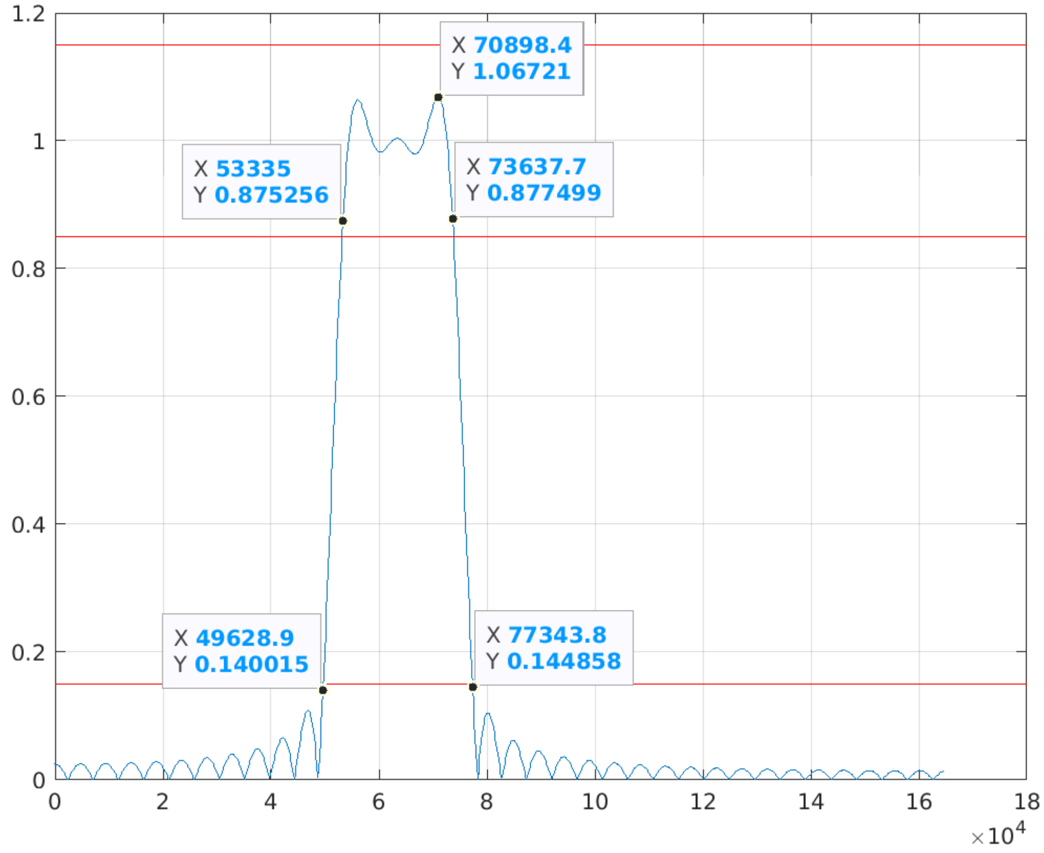

Figure 18: Magnitude Plot

In the diagram the horizontal lines are to represent the tolerance criteria and frequencies on the band-pass and band-stop edges are marked appropriately.

In the above plot, the band edge frequencies have been marked. From the magnitude at these frequencies it can be seen that the specifications required in the passband and the stopband have been met.

4.2 Filter 2 - Bandstop

4.2.1 IIR Filter

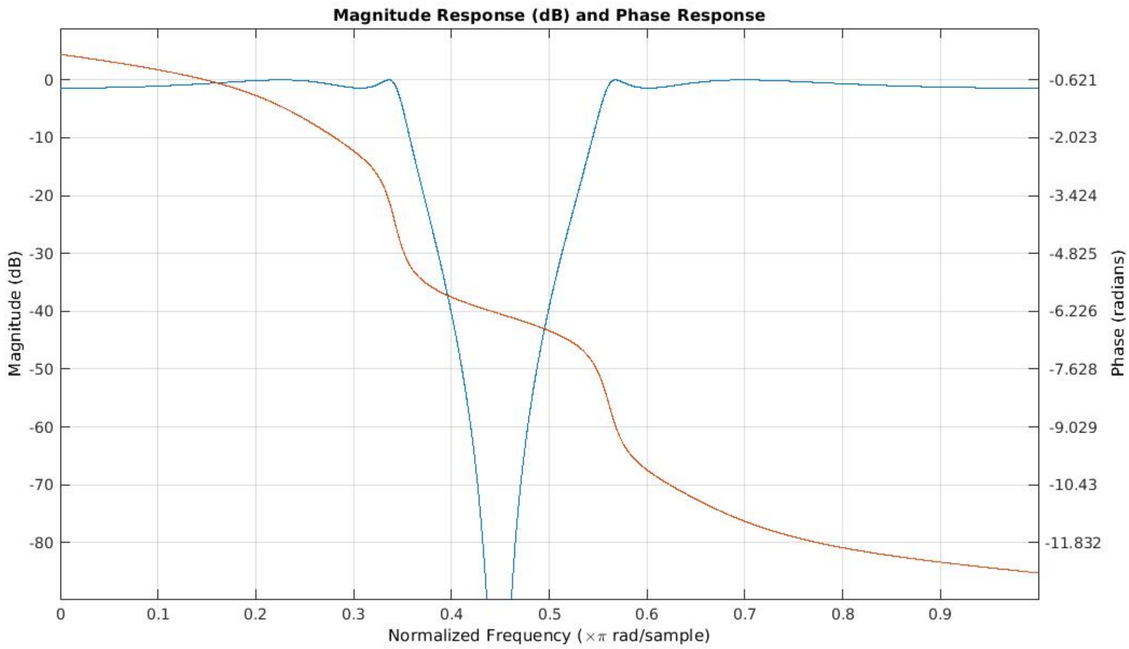

Figure 19: Frequency Response

From the above plot, it is verified that the passband tolerance and stopband attenuation have been satisfied. It can be seen that the **phase response** is **not linear**.

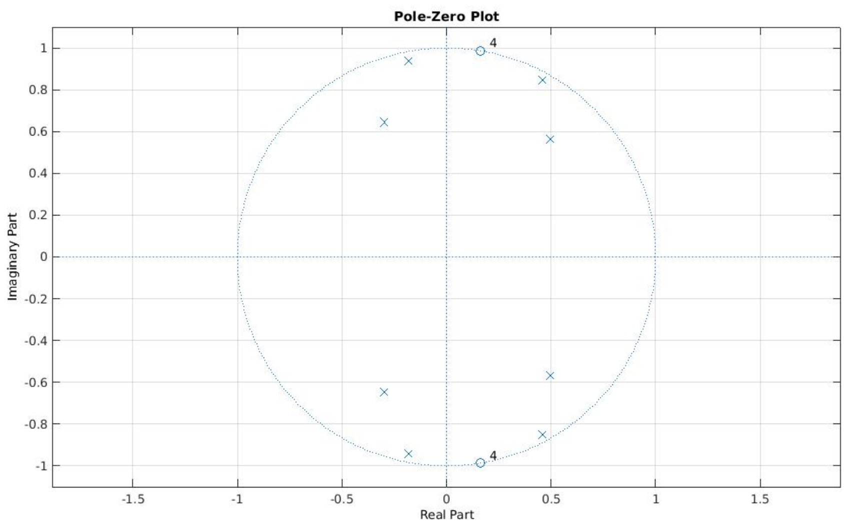

Figure 20: Pole-Zero map

The zeroes are the circles(dots) and the poles are the crosses in the above pole-zero plot. As seen in the figure, all of the crosses lie inside the unit circle. This leads to the conclusion that it is indeed stable as required.

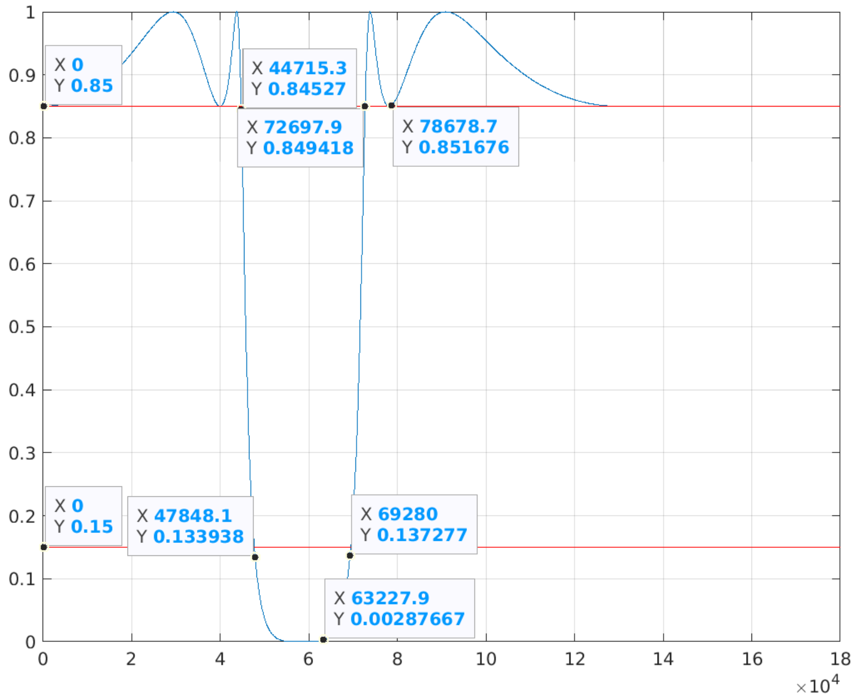

Figure 21: Magnitude Plot

In the diagram the horizontal lines are to represent the tolerance criteria and frequencies on the band-pass and band-stop edges are marked appropriately.

In the above plot, the band edge frequencies have been marked. From the magnitude at these frequencies it can be seen that the specifications required in the passband and the stopband have been met.

One can clearly observe from the two diagrams that the transition between passband and stop band is much steeper in the case for the Chebyshev implementation as compared to the Butterworth implementation.

4.2.2 FIR Filter

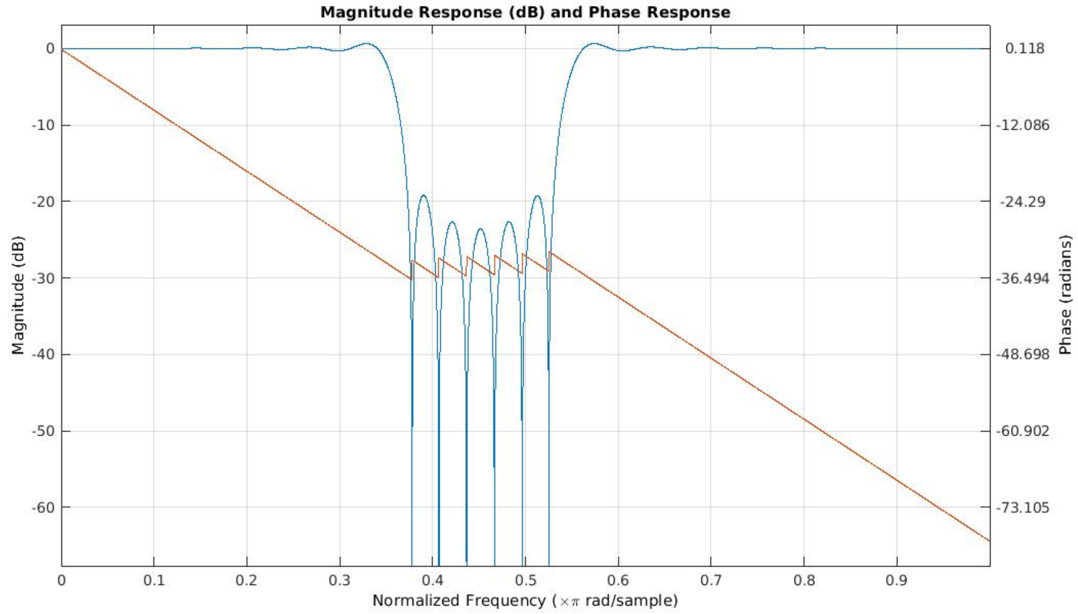

Figure 22: Frequency Response

From the above plot, it is verified that the passband tolerance and stopband attenuation have been satisfied. It can be seen that the FIR Filter indeed gives us a **Linear Phase** response which is desired.

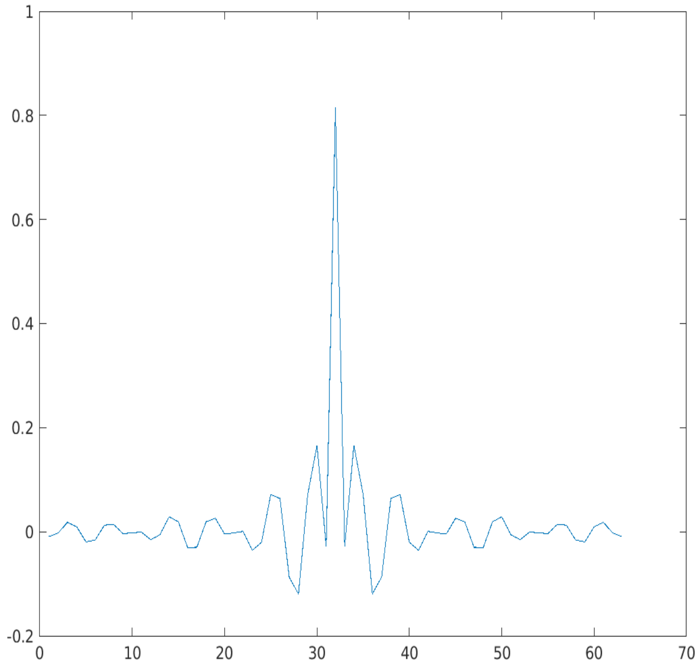

Figure 23: Time Domain Sequence

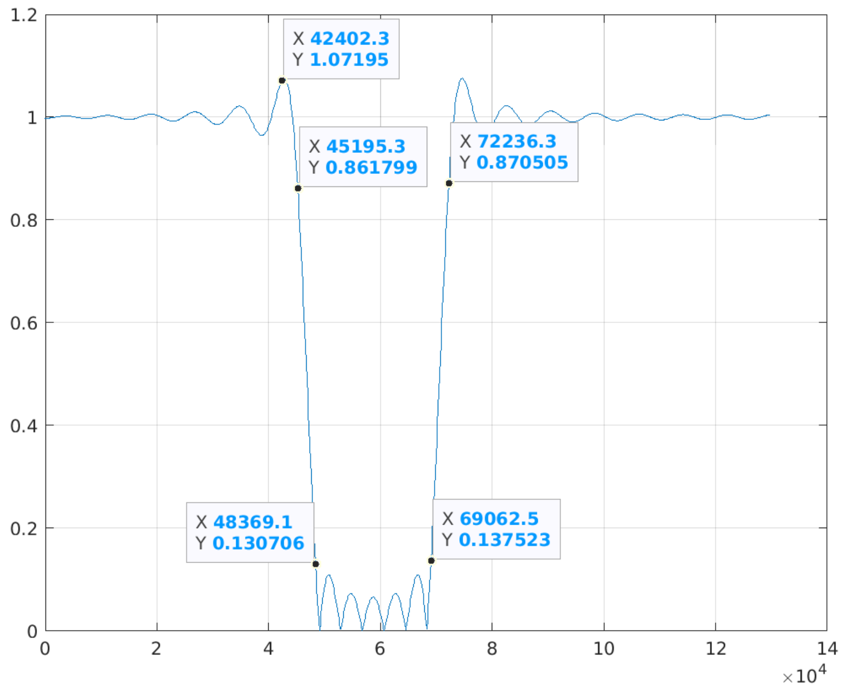

Figure 24: Magnitude Plot

In the above plot, the band edge frequencies have been marked. From the magnitude at these frequencies it can be seen that the specifications required in the passband and the stopband have been met.

We can see that the band-stop and band-pas edges criteria are properly met. Thus our design was correct for both the Butterworth band-pass design and the Chebyshev band-stop design.

5 Review:

Reviewer – Shreyan Jabade (180100055)

The specifications for both the band-pass and the band-stop filter are correct, the following analog transformations are carried out correctly. Also the calculations for the Kaiser window parameters are done correctly.

The frequency response for both IIR and FIR implementations are correct. For the chosen parameter values, the frequency response are within their tolerance limits. The important band edges and other points are also shown properly.

Hence the overall frequency respose specifications are correct. All the parts regarding the compulsary section of the filter design assignment are completed. Also the codes used in the filter designs are attached at the end.

6 Appendix A: MATLAB codes

6.1 Filter 1 : Bandpass filter

6.1.1 IIR : Butterworth

```
%Butterworth Analog LPF parameters
Wc = 1.0788;                %cut-off frequency
N = 8;                    %order

%poles of Butterworth polynomial of degree 8 in the open CLHP
p1 = Wc*cos(pi/2 + pi/16) + Wc*sin(pi/2 + pi/16)*1i;
p2 = Wc*cos(pi/2 + pi/16) - i*Wc*sin(pi/2 + pi/16);
p3 = Wc*cos(pi/2 + pi/16+pi/8) + i*Wc*sin(pi/2 + pi/16+pi/8);
p4 = Wc*cos(pi/2 + pi/16+pi/8) - i*Wc*sin(pi/2 + pi/16+pi/8);
p5 = Wc*cos(pi/2 + pi/16+2*pi/8) + i*Wc*sin(pi/2 + pi/16+2*pi/8);
p6 = Wc*cos(pi/2 + pi/16+2*pi/8) - i*Wc*sin(pi/2 + pi/16+2*pi/8);
p7 = Wc*cos(pi/2 + pi/16+3*pi/8) + i*Wc*sin(pi/2 + pi/16+3*pi/8);
p8 = Wc*cos(pi/2 + pi/16+3*pi/8) - i*Wc*sin(pi/2 + pi/16+3*pi/8);

fprintf("p1 = %d", real(p1));
fprintf("+ i %d", imag(p1));
disp(" ");
fprintf("p2 = %d", real(p2));
fprintf("+ i %d", imag(p2));
disp(" ");
fprintf("p3 = %d", real(p3));
fprintf("+ i %d", imag(p3));
disp(" ");
fprintf("p4 = %d", real(p4));
fprintf("+ i %d", imag(p4));
disp(" ");
fprintf("p5 = %d", real(p5));
fprintf("+ i %d", imag(p5));
disp(" ");
fprintf("p6 = %d", real(p6));
fprintf("+ i %d", imag(p6));
disp(" ");
fprintf("p7 = %d", real(p7));
fprintf("+ i %d", imag(p7));
disp(" ");
fprintf("p8 = %d", real(p8));
fprintf("+ i %d", imag(p8));

%Band Edge speifications
fp1 = 53.5;
fs1 = 49.5;
fs2 = 77.5;
fp2 = 73.5;
```

```
%Transformed Band Edge specs using Bilinear Transformation
f_samp = 330;
wp1 = tan(fp1/f_samp*pi);
ws1 = tan(fs1/f_samp*pi);
ws2 = tan(fs2/f_samp*pi);
wp2 = tan(fp2/f_samp*pi);

%Parameters for Bandpass Transformation
W0 = sqrt(wp1*wp2);
B = wp2-wp1;

[num,den] = zp2tf([],[p1 p2 p3 p4 p5 p6 p7 p8],Wc^N);   %TF with poles p1-p8 and numerator Wc^N and no zeroes
                                                        %numerator chosen to make the DC Gain = 1

%Evaluating Frequency Response of Final Filter
syms s z;
analog_lpf(s) = poly2sym(num,s)/poly2sym(den,s);        %analog LPF Transfer Function
analog_bsf(s) = analog_lpf((s*s + W0*W0)/(B*s));        %bandpass transformation
discrete_bsf(z) = analog_bsf((z-1)/(z+1));              %bilinear transformation

%coeffs of analog bsf
[ns, ds] = numden(analog_bsf(s));                   %numerical simplification to collect coeffs
ns = sym2poly(expand(ns));
ds = sym2poly(expand(ds));                          %collect coeffs into matrix form
k = ds(1);
ds = ds/k;
ns = ns/k;
disp("Coefficient of analog bpf");
disp(ns);
disp(ds);
%coeffs of discrete bsf
[nz, dz] = numden(discrete_bsf(z));                     %numerical simplification to collect coeffs
nz = sym2poly(expand(nz));
dz = sym2poly(expand(dz));                              %coeffs to matrix form
k = dz(1);                                              %normalisation factor
dz = dz/k;
nz = nz/k;
disp("Coefficient of discrete bpf");
disp(nz);
disp(dz);
fvtool(nz,dz)                                           %frequency response
```

```
%magnitude plot (not in log scale)
[H,f] = freqz(nz,dz,1024*1024, 330e3);
figure;
plot(f,abs(H))
grid
line([0;18e4], [0.85;0.85], 'Color', 'red');
line([0;18e4], [0.15;0.15], 'Color', 'red');
```

6.1.2 FIR : Kaiser Window

```
f_samp = 330e3;
%Band Edge speifications
fp1 = 53.5;
fs1 = 49.5;
fs2 = 77.5;
fp2 = 73.5;

Wc1 = fp1*2*pi/f_samp;
Wc2  = fp2*2*pi/f_samp;

%Kaiser paramters
A = -20*log10(0.15);
if(A < 21)
    beta = 0;
elseif(A <51)
    beta = 0.5842*(A-21)^0.4 + 0.07886*(A-21);
else
    beta = 0.1102*(A-8.7);
end
N_min = ceil((A-8) / (2.285*0.02424*pi));     %empirical formula for N_min

%Window length for Kaiser Window
n=N_min+20;

%Ideal bandpass impulse response of length "n"
bp_ideal = ideal_lp(0.4577*pi,n) - ideal_lp(0.312*pi,n);

%Kaiser Window of length "n" with shape paramter beta calculated above
kaiser_win = (kaiser(n,beta))';

FIR_BandPass = bp_ideal .* kaiser_win;
fvtool(FIR_BandPass);         %frequency response
disp(bp_ideal);
figure;
plot(bp_ideal);
%magnitude response
[H,f] = freqz(FIR_BandPass,1,1024, f_samp);
figure;
plot(f,abs(H))
grid
line([0;18e4], [0.85;0.85], 'Color', 'red');
line([0;18e4], [1.15;1.15], 'Color', 'red');
line([0;18e4], [0.15;0.15], 'Color', 'red');
```

6.2 Filter 2 : Bandstop filter

6.2.1 IIR : Chebyshev

```
%Chebyshev LPF parameters
D1 = 1/(0.85*0.85)-1;       %since delta is 0.15
epsilon = sqrt(D1);         %epsilon was set to this value to satisfy required inequality
N = 4;

% Open CLHP Poles of the Chebyshev Polynomial of order 4
p1 = -sin(pi/(2*N))*sinh(asinh(1/epsilon)/N)+i*cos(pi/(2*N))*cosh(asinh(1/epsilon)/N);
p2 = -sin(pi/(2*N))*sinh(asinh(1/epsilon)/N)-i*cos(pi/(2*N))*cosh(asinh(1/epsilon)/N);
p3 = -sin(3*pi/(2*N))*sinh(asinh(1/epsilon)/N)+i*cos(3*pi/(2*N))*cosh(asinh(1/epsilon)/N);
p4 = -sin(3*pi/(2*N))*sinh(asinh(1/epsilon)/N)-i*cos(3*pi/(2*N))*cosh(asinh(1/epsilon)/N);

fprintf("p1 = %d", real(p1));
fprintf("+ i %d", imag(p1));
disp(" ");
fprintf("p2 = %d", real(p2));
fprintf("+ i %d", imag(p2));
disp(" ");
fprintf("p3 = %d", real(p3));
fprintf("+ i %d", imag(p3));
disp(" ");
fprintf("p4 = %d", real(p4));
fprintf("+ i %d", imag(p4));
disp(" ");

%evaluating the Transfer function of Chebyshev Analog LPF
n1 = [1 -p1-p2 p1*p2];
n2 = [1 -p3-p4 p3*p4];
den = conv(n1,n2);          %multiply n1 and n2, which are the two quadratic factors in the denominator
num = [den(5)*sqrt(1/(1+epsilon*epsilon))];       % even order, DC Gain set as 1/(1+ epsilon^2)^0.5

%Band Edge speifications
fs1 = 48.7;
fp1 = 44.7;
fp2 = 72.7;
fs2 = 68.7;

%Transformed Band Edge specs using Bilinear Transformation
f_samp = 260;
ws1 = tan(fs1/f_samp*pi);
wp1 = tan(fp1/f_samp*pi);
wp2 = tan(fp2/f_samp*pi);
ws2 = tan(fs2/f_samp*pi);
```

```
%Parameters for Bandstop Transformation
W0 = sqrt(wp1*wp2);
B = wp2-wp1;

%Evaluating Frequency Response of Final Filter
syms s z;
analog_lpf(s) = poly2sym(num,s)/poly2sym(den,s);    %analog lpf transfer function
analog_bpf(s) = analog_lpf((B*s)/(s*s +W0*W0));     %bandstop transformation
discrete_bpf(z) = analog_bpf((z-1)/(z+1));          %bilinear transformation

%coeffs of analog BPF
[ns, ds] = numden(analog_bpf(s));                   %numerical simplification
ns = sym2poly(expand(ns));
ds = sym2poly(expand(ds));                          %collect coeffs into matrix form
k = ds(1);
ds = ds/k;
ns = ns/k;
disp("Coefficient of analog bpf");
disp(ns);
disp(ds);
%coeffs of discrete BPF
[nz, dz] = numden(discrete_bpf(z));                 %numerical simplification
nz = sym2poly(expand(nz));
dz = sym2poly(expand(dz));                          %collect coeffs into matrix form
k = dz(1);                                          %normalisation factor
dz = dz/k;
nz = nz/k;
disp("Coefficient of discrete bpf");
disp(nz);
disp(dz);
fvtool(nz,dz)                                       %frequency response in dB

%magnitude plot (not in log scale)
[H,f] = freqz(nz,dz,1024*1024, 260e3);
figure;
plot(f,abs(H))
grid
line([0;18e4], [0.85;0.85], 'Color', 'red');
line([0;18e4], [0.15;0.15], 'Color', 'red');
```

6.2.2 FIR : Kaiser Window

```
f_samp = 260e3;
%Band Edge speifications
fs1 = 48.7e3;
fp1 = 44.7e3;
fp2 = 72.7e3;
fs2 = 68.7e3;

%Kaiser paramters
A = -20*log10(0.15);
if(A < 21)
    beta = 0;
elseif(A <51)
    beta = 0.5842*(A-21)^0.4 + 0.07886*(A-21);
else
    beta = 0.1102*(A-8.7);
end

Wn = [(fs1+fp1)/2 (fs2+fp2)/2]*2/f_samp;        %average value of the two paramters
N_min = ceil((A-7.95) / (2.285*0.02424*pi));       %empirical formula for N_min

%Window length for Kaiser Window
n=N_min + 13;

%Ideal bandstop impulse response of length "n"

bs_ideal =  ideal_lp(pi,n) -ideal_lp(0.5435*pi,n) + ideal_lp(0.3595*pi,n);

%Kaiser Window of length "n" with shape paramter beta calculated above
kaiser_win = (kaiser(n,beta))';

FIR_BandStop = bs_ideal .* kaiser_win;
fvtool(FIR_BandStop);          %frequency response
disp(bs_ideal);
figure;
plot(bs_ideal);
%magnitude response
[H,f] = freqz(FIR_BandStop,1,1024, f_samp);
figure;
plot(f,abs(H))
grid
line([0;18e4], [0.85;0.85], 'Color', 'red');
line([0;18e4], [1.15;1.15], 'Color', 'red');
line([0;18e4], [0.15;0.15], 'Color', 'red');
```

6.3 Review report of Mutual Peer review done by self

Review of report of Archishman Biswas, 180070009

- The specifications for both the filters are correct and the transformations are carried out correctly. Also the calculations for the Kaiser window parameters are done correctly.
- The frequency response for both IIR and FIR implementations are correct. The frequency response obtained are within their tolerance limits.
- All the parts regarding the compulsary section of the filter design assignment are completed and also some portion of the elliptical filter design. The codes for the design is provided in a link.