Digital Signal Processing
EE-338
Filter Design Assignment

VINAY

19D070068
Autumn 2022
IIT BOMBAY

March 31, 2022

Contents

1 IIR Bandpass Filter

The filter number m = 72.So,

q(m) = $\lceil 0.1m \rceil$ = 7

r(m) = m-10q(m) = 2

BL(m) = 10+5q(m)+13r(m) = 71kHz

‘ BL(h) = BL(m)+45 = 116kHZ

1.1 Unnormalized Specifications

- Upper Passband edge = 116kHz
- Lower Passband edge = 71kHz
- Passband tolerance = 0.15
- Stopband tolerance = 0.15
- Transition bandwidth = 3kHz
- Sampling frequency = 540kHz
- Passband nature = Monotonic
- Stopband nature = Monotonic

1.2 Normalized Specifications

The sampling angular frequency 2πfs should map to 2π after normalization. Hence the transformation used is

$$\omega = 2\pi \frac{f}{f_s}$$

Table 1: Specification Table

Parameter	Value
ω_{p2}	0.4296π
ω_{p1}	0.2629π
ω_{s1}	0.2518π
ω_{s2}	0.4407π

- Passband and Stopband tolerance = 0.15
- Passband and Stopband nature = Monotonic

1.3 Analog Filter Specifications using Bilinear Transform

We use the following transformation to move from the z domain to the s domain.

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

If we put z = jω and s = jΩ, we get

$$\Omega = tan(\frac{\omega}{2})$$

The filter specifications now are as follows

Table 2: Specification Table

Parameter	Value
Ω_{p2}	0.80012
Ω_{p1}	0.4381
Ω_{s1}	0.4175
Ω_{s2}	0.8291
δ_1	0.15
δ_2	0.15

1.4 Analog Frequency Transformation

Now we will do Bandpass to Lowpass frequency transformation:

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

Where

$$\Omega_0 = \sqrt{\Omega_{p1}\Omega p2} = 0.592$$

and

$$B = \Omega_{p2} - \Omega p1 = 0.362$$

Using the above transformation we have

- Ω_{Ls1} = -1.1655
- Ω_{Ls2} = 1.1226
- Ω_{Lp1} = -1
- Ω_{Lp2} = +1
- Passband and Stopband nature = Monotonic

1.5 Equivalent Analog Lowpass Filter Specifications

Since we have done the analog frequency transformation above,now we will find the low pass filter specifications by using the most stringent conditions

- Passband and Stopband nature = Monotonic
- $\Omega_{Ls} = \min(\Omega_{Ls2},|\Omega_{Ls1}|) = 1.1226$
- $\Omega_{Lp} = +1$
- $\delta_1 = \delta_2 = 0.15$
- $D_1 = \frac{1}{(1-\delta_1)^2}$-1 = 0.384
- $D_2 = \frac{1}{\delta_2^2}$-1 = 43.44

1.6 Analog Lowpass Magnitude Response

We know that for a Butterworth filter,the response is as follows

$$H_{LPF}(-s_L)H_{LPF}(s_L) = \frac{1}{1+(\frac{s_L}{j\Omega_C})^{2N}}$$

The parameters N and Ωc can be calculated by using the following relations:

$$N = \frac{log(\sqrt{D_2/D_1})}{log(\Omega_{Ls}/\Omega_{Lp})}$$

$$\frac{\Omega_{Lp}^{2N}}{D1} \leq \Omega_c^{2N} \leq \frac{\Omega_{Ls}^{2N}}{D2}$$

Using the above relations,We get

$$N = 21$$

$$1.02305 \leq \Omega_c \leq 1.026189$$

We choose $\Omega_c = 1.02462$. Using these values, we have

$$H_{LPF}(-s_L)H_{LPF}(s_L) = \frac{1}{1+(\frac{s_L}{j1.02462})^{42}}$$

The poles are plotted using the matlab as shown here

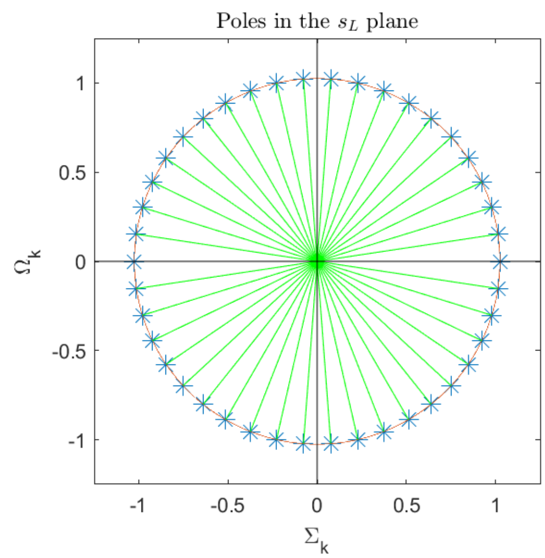

Figure 1: Poles in the s_L plane

The poles which lies in the left half plane are of our interest because only LHP ensures stability of our system and hence makes our filter achievable.

```
-0.2280 + 0.9989i

-0.3743 + 0.9538i

-0.5123 + 0.8873i

-0.6388 + 0.8011i

-0.7511 + 0.6969i

-0.8466 + 0.5772i

-0.9232 + 0.4446i

-0.9791 + 0.3020i
```

```
-1.0132 + 0.1527i

-1.0246 + 0.0000i

-1.0132 - 0.1527i

-0.9791 - 0.3020i

-0.9232 - 0.4446i

-0.8466 - 0.5772i

-0.7511 - 0.6969i

-0.6388 - 0.8011i

-0.5123 - 0.8873i

-0.3743 - 0.9538i

-0.2280 - 0.9989i

-0.0766 - 1.0218i

-0.0766 + 1.0218i
```

We can calculate the equivalent low pass filter in analog domain by using the LHP poles calculated above by the use of the following equations

$$H_{LPF}(s_L) = \frac{\Omega_c^N}{\prod_{k=1}^{N}(s_L - s_k)} = \frac{1.6665}{\sum_{k=1}^{21} a_k s_L^k}$$

where s_k are the LHP poles

The Magnitude and Phase response of the equivalent Butterworth LPF are

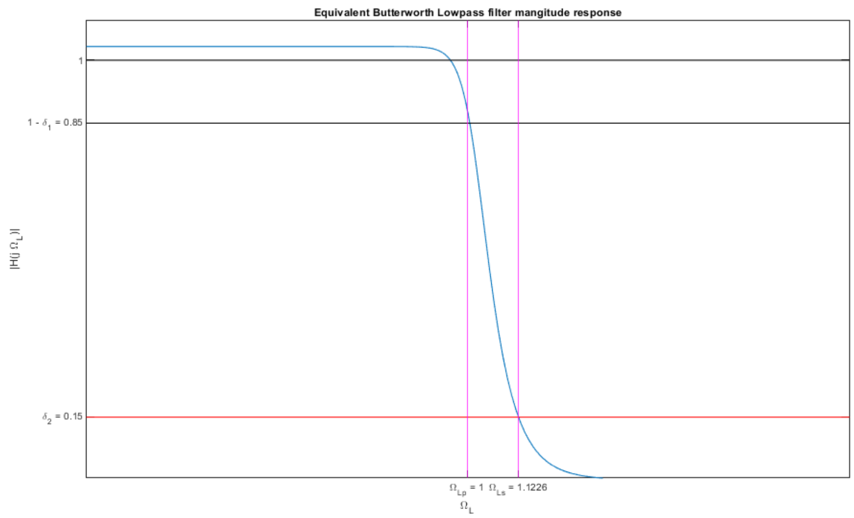

Figure 2: Magnitude Response

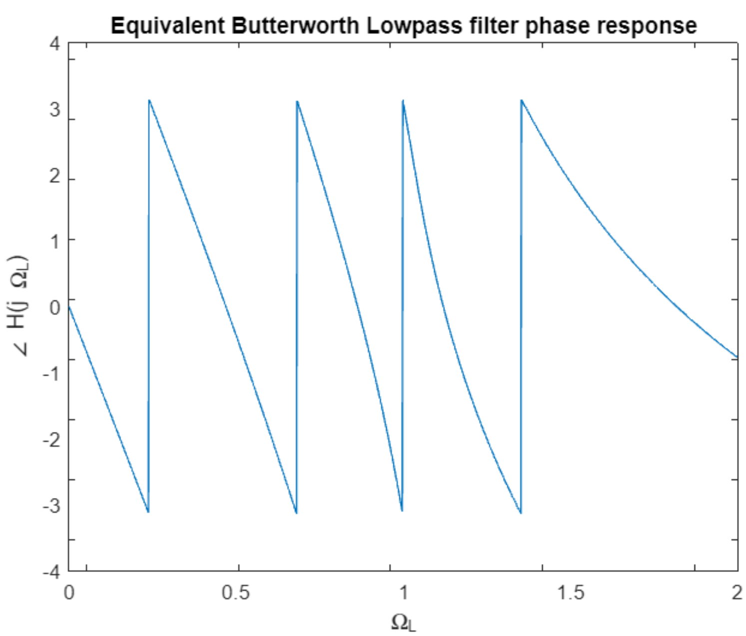

Figure 3: Phase Response

1.7 Analog Bandpass Transfer Function

We now use the inverse transformation to convert the equivalent Butterworth lowpass transfer function to get the Bandpass transfer function using the relation:

$$s_L \longleftarrow \frac{s^2 + \Omega_0^2}{Bs}$$

Where Ω and B are the values we found in section 1.1

$$H_{LPF}(\frac{s^2 + \Omega_0^2}{Bs}) = H_{BFP} = \frac{1.6665}{\sum_{k=1}^{21} a_k s_L^k}$$

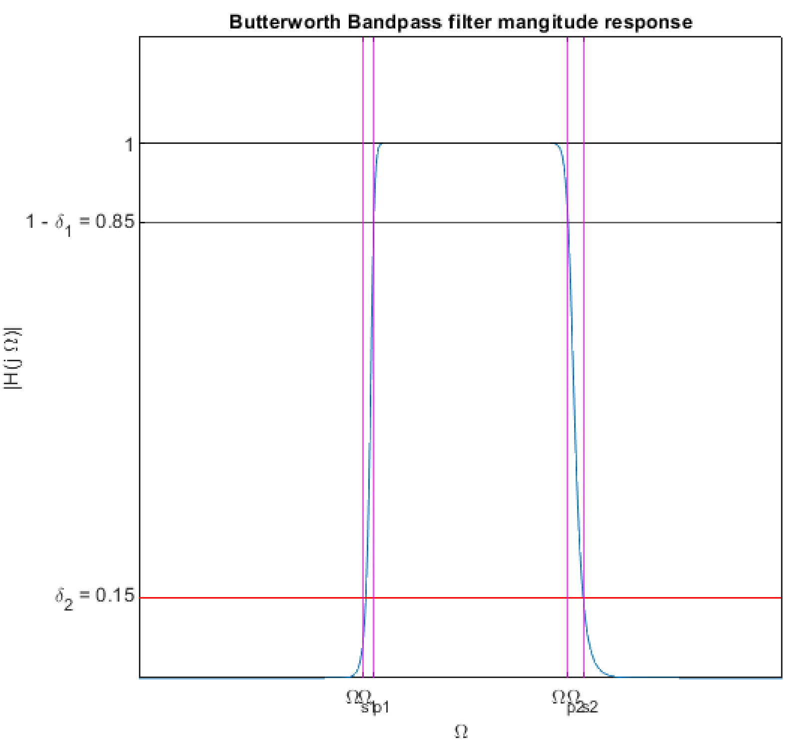

Figure 4: Magnitude Response

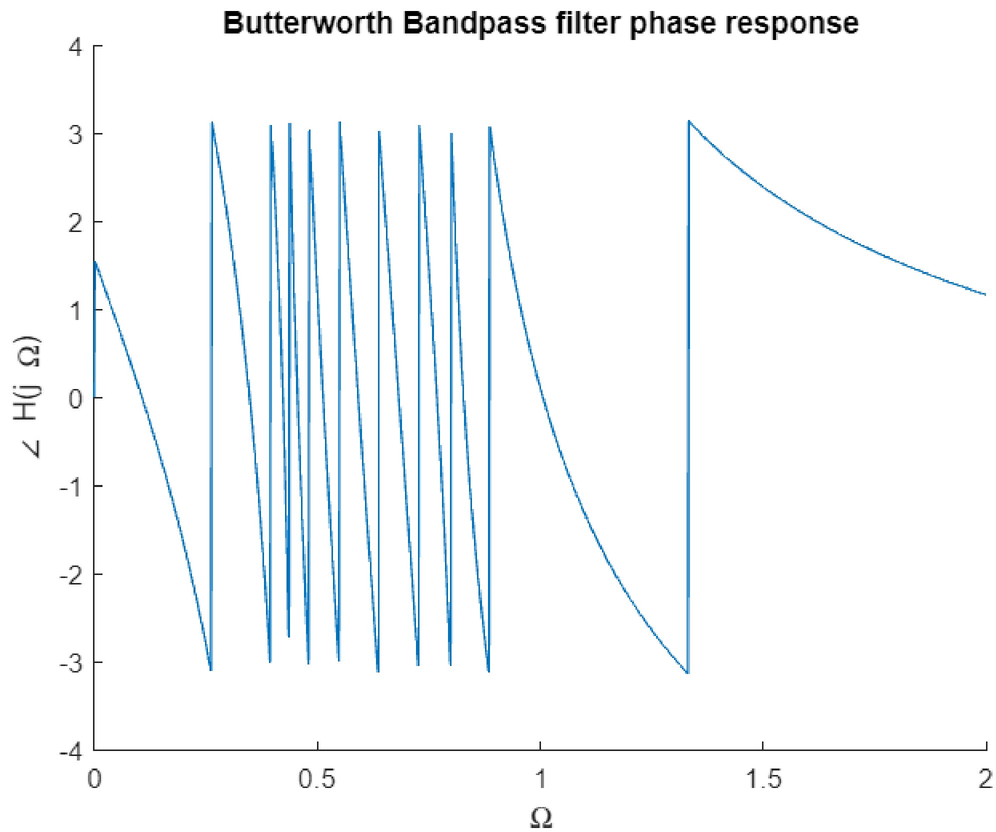

Figure 5: Phase Response

1.8 Discrete time Bandpass Transformation

Now we will convert the analog bandpass filter into the discrete bandpass filter by using the bilinear transformation in the normalized angular frequency domain.

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

The discrete time BPF transfer function is

$$H_{BPF}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right) = \frac{\sum_{k=0}^{42} a_k z^k}{\sum_{k=0}^{42} b_k z^k}$$

The coefficients for the Discrete Time Bandpass filter is given the table 3

Table 3: Coefficient Table

Coefficient	Value	Coefficient	Value	Coefficient	Value	Coefficient	Value
a_{42}	0	a_{21}	-3.2280	b_{42}	-1	b_{21}	0
a_{41}	0	a_{20}	2.4553	b_{41}	0	b_{20}	352716
a_{40}	0	a_{19}	-1.7456	b_{40}	21	b_{19}	0
a_{39}	0.0002	a_{18}	1.1589	b_{39}	0	b_{18}	-293930
a_{38}	-0.0010	a_{17}	-0.7176	b_{38}	-210	b_{17}	0
a_{37}	0.0037	a_{16}	0.4138	b_{37}	0	b_{16}	203490
a_{36}	-0.0119	a_{15}	-0.2217	b_{36}	1330	b_{15}	0
a_{35}	0.0336	a_{14}	0.1101	b_{35}	0	b_{14}	-116280
a_{34}	-0.0835	a_{13}	-0.0505	b_{34}	-5985	b_{13}	0
a_{33}	0.1857	a_{12}	0.0213	b_{33}	0	b_{12}	54264
a_{32}	-0.3731	a_{11}	-0.0082	b_{32}	20349	b_{11}	0
a_{31}	0.6821	a_{10}	0.0029	b_{31}	0	b_{10}	-20349
a_{30}	-1.1421	a_9	-0.0009	b_{30}	-54264	b_9	0
a_{29}	1.7596	a_8	0.0003	b_{29}	0	b_8	5985
a_{28}	-2.5049	a_7	-0.0001	b_{28}	116280	b_7	0
a_{27}	3.3053	a_6	0	b_{27}	0	b_6	-1330
a_{26}	-4.0553	a_5	0	b_{26}	-203490	b_5	0
a_{25}	4.6292	a_4	0	b_{25}	0	b_4	210
a_{24}	-4.9317	a_3	0	b_{24}	293930	b_3	0
a_{23}	4.9067	a_2	0	b_{23}	0	b_2	-21
a_{22}	-4.5632	a_1	0	b_{22}	-352716	b_1	0
		a_0	3.9685			b_0	1

2 FIR Bandpass filter

The finite Impulse Response Bandpass filter is designed by approximating the infinitely long impulse response with a finite impulse response by using windowing methods. We will use the **Kaiser Window** to implement the above design, which is characterized by the width(M) and shape(β)

The normalized specification are as follows:

- $\omega_{p2} = 0.4296\pi$
- $\omega_{p1} = 0.2629\pi$
- $\omega_{s1} = 0.2518\pi$
- $\omega_{s2} = 0.4407\pi$
- Transition Bandwidth($\Delta\omega_t$) = 0.0111π
- Passband and Stopband Tolernce(δ) = 0.15

We will use the kaiser window and multiply it with the impulse response of the ideal bandpass filter to get the FIR filter.

2.1 Implementation Method

The mean of the lower passband and lower stopband is taken as the lower passband edge of the ideal bandpass filter while the mean of the upper passband and the upper stopband is taken as the upper passband edge of the BP filter

2.2 Kaiser Window Parameters

The value of the parameter is less than 21 and therefore $\alpha = 0$ and so the shaping parameter β and therefore, the kaiser window will be rectangular in shape

$$A = -20\log(\delta) = 16.47817$$

Window width M is

$$M \geq 1 + \frac{A-8}{2.285\Delta\omega_t} = 107.29$$

I will take odd value of M:

$$M = 111$$

2.3 Magnitude and Phase Response

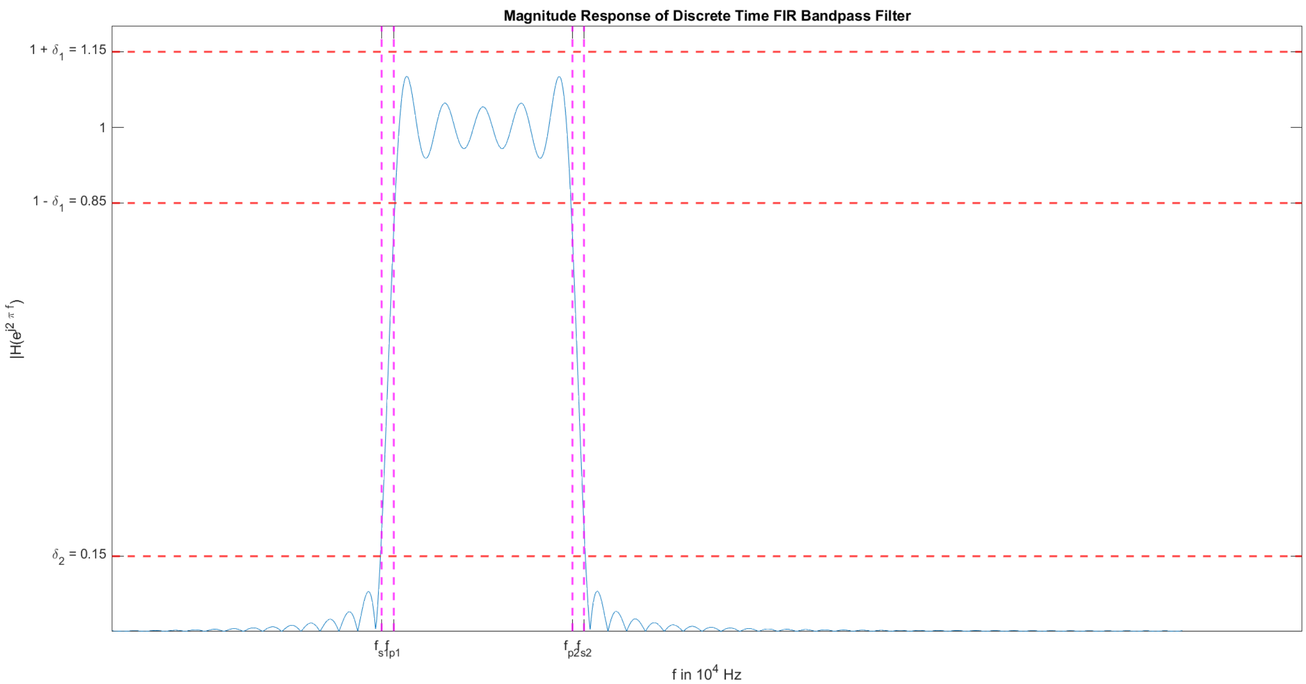

Figure 6: Magnitude Response of the FIR Bandpass Filter

The coefficents are:

```
Columns 1 through 13

 -0.0039   -0.0040    0.0043    0.0113    0.0059   -0.0070   -0.0116   -0.0036    0.0047    0.0037    0.0000    0.0029    0.0074

Columns 14 through 26

  0.0015   -0.0114   -0.0140   -0.0002    0.0137    0.0113   -0.0011   -0.0058   -0.0010   -0.0009   -0.0090   -0.0095    0.0064

Columns 27 through 39

  0.0211    0.0131   -0.0103   -0.0203   -0.0077    0.0061    0.0038   -0.0014    0.0081    0.0199    0.0068   -0.0251   -0.0351

Columns 40 through 52

 -0.0048    0.0302    0.0280    0.0001   -0.0098    0.0033   -0.0026   -0.0388   -0.0470    0.0200    0.1024    0.0833   -0.0506

Columns 53 through 65

 -0.1565   -0.0959    0.0815    0.1778    0.0815   -0.0959   -0.1565   -0.0506    0.0833    0.1024    0.0200   -0.0470   -0.0388

Columns 66 through 78

 -0.0026    0.0033   -0.0098    0.0001    0.0280    0.0302   -0.0048   -0.0351   -0.0251    0.0068    0.0199    0.0081   -0.0014

Columns 79 through 91

  0.0038    0.0061   -0.0077   -0.0203   -0.0103    0.0131    0.0211    0.0064   -0.0095   -0.0090   -0.0009   -0.0010   -0.0058

Columns 92 through 104

 -0.0011    0.0113    0.0137   -0.0002   -0.0140   -0.0114    0.0015    0.0074    0.0029    0.0000    0.0037    0.0047   -0.0036

Columns 105 through 111

 -0.0116   -0.0070    0.0059    0.0113    0.0043   -0.0040   -0.0039
```

Figure 7: Magnitude Response of the FIR Bandpass Filter

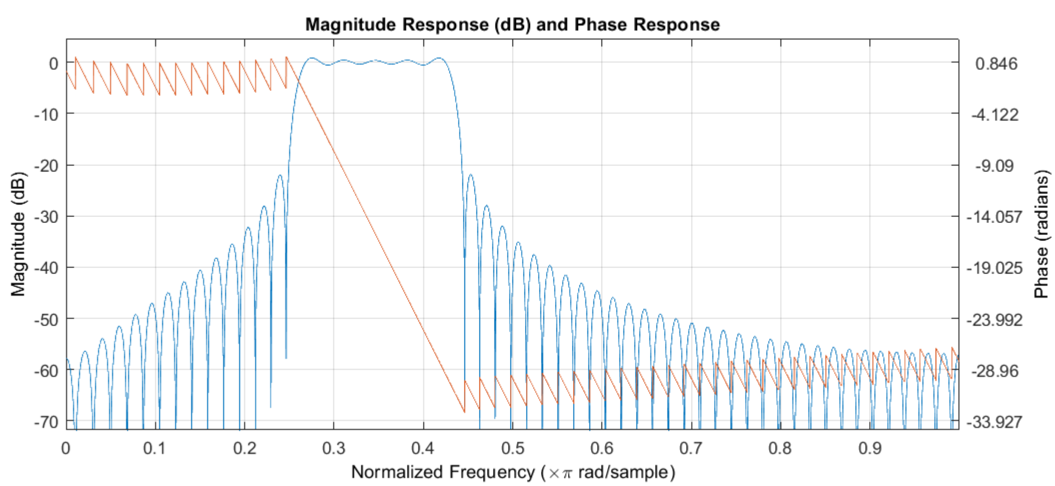

Figure 8: Phase Response of the FIR Bandpass Filter

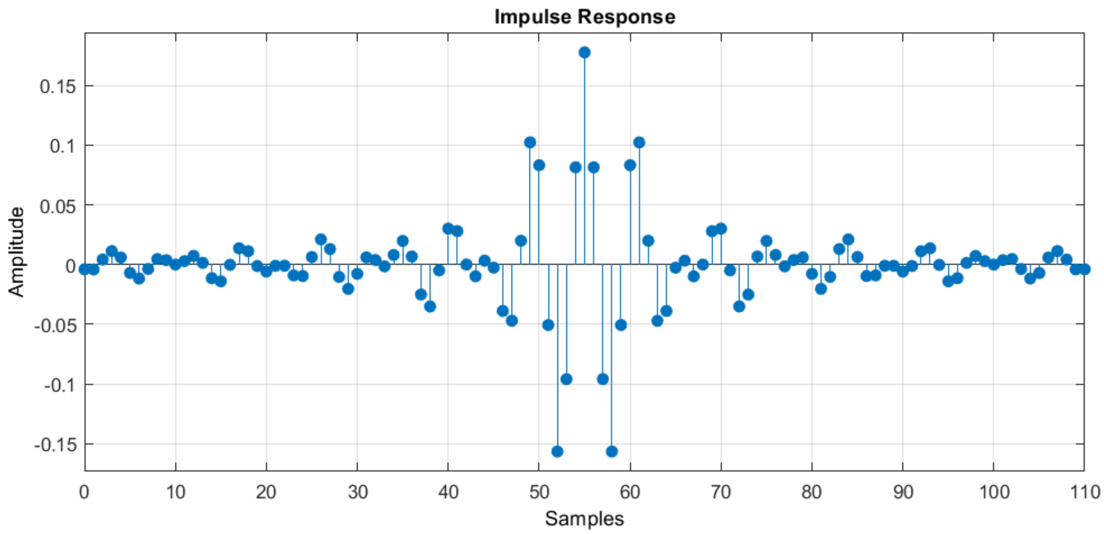

Figure 9: Impulse Response of the FIR Bandpass Filter

3 Infinite Impulse Response Bandstop Filter

The filter number m = 72.So,

q(m) = $\lceil 0.1m \rceil$ = 7

r(m) = m-10q(m) = 2

BL(m) = 10+3q(m)+11r(m) = 48kHz

BL(h) = BL(m)+25 = 73kHZ

3.1 Unnormalized Specifications

- Upper Stopband edge = 73kHz
- Lower Stopband edge = 48kHz
- Passband tolerance = 0.15
- Stopband tolerance = 0.15
- Transition bandwidth = 3kHz
- Sampling frequency = 400kHz
- Passband nature = Equiripple
- Stopband nature = Monotonic

3.2 Normalized Specifications

The sampling angular frequency 2πfs should map to 2π after normalization. Hence the transformation used is

$$\omega = 2\pi \frac{f}{f_s}$$

Table 4: Specification Table

Parameter	Value
ω_{p1}	0.7068
ω_{s1}	0.7539
ω_{s2}	1.1466
ω_{p2}	1.1938

- Passband and Stopband tolerance = 0.15
- Stopband nature = Monotonic
- Passband nature = Equiripple

3.3 Analog Filter Specifications using Bilinear Transform

We use the following transformation to move from the s domain to the z domain.

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

If we put z = jω and s = jΩ, we get

$$\Omega = tan(\frac{\omega}{2})$$

The filter specifications now are as follows

Table 5: Specification Table

Parameter	Value
Ω_{p1}	0.3689
Ω_{s1}	0.3959
Ω_{s2}	0.6457
Ω_{p2}	0.6795
δ_1	0.15
δ_2	0.15

3.4 Analog Frequency Transformation

Now we will do Bandpass to Lowpass frequency transformation:

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

Where

$$\Omega_0 = \sqrt{\Omega_{p1}\Omega p2} = 0.5007$$

and

$$B = \Omega_{p2} - \Omega p1 = 0.31067$$

Using the above transformation we have

- Ω_{Ls1} = -1.30916
- Ω_{Ls2} = 1.20699
- Ω_{Lp1} = -1
- Ω_{Lp2} = +1
- Stopband nature = Monotonic
- Passband nature = Equiripple

3.5 Equivalent Analog Lowpass Filter Specifications

Since we have done the analog frequency transformation above,now we will find the low pass filter specifications by using the most stringent conditions

- Stopband nature = Monotonic
- Passband nature = Equiripple
- Ω_{Ls} = min(Ω_{Ls2},$|\Omega_{Ls1}|$) = 1.2069
- $\Omega_{Lp} = +1$
- $\delta_1 = \delta_2 = 0.15$
- $D_1 = \frac{1}{(1-\delta_1)^2}$-1 = 0.384
- $D_2 = \frac{1}{\delta_2^2}$-1 = 43.44

3.6 Analog Lowpass Magnitude Response

We know that for a Chebyshew filter,the response is as follows

$$H_{LPF}(-s_L)H_{LPF}(s_L) = \frac{1}{1 + \epsilon^2 C_N^2(\frac{s_L}{j\Omega_p})}$$

The parameters N and ϵ can be calculated by using the following relations:

$$\epsilon = \sqrt{D_1} = 0.6197$$

$$N = \frac{cosh^{-1}(\sqrt{D_2/D_1})}{cosh^{-1}(\Omega_{Ls}/\Omega_{Lp})}$$

Using the above relations,We get

$$N = 5$$

The poles are plotted using the matlab as shown here

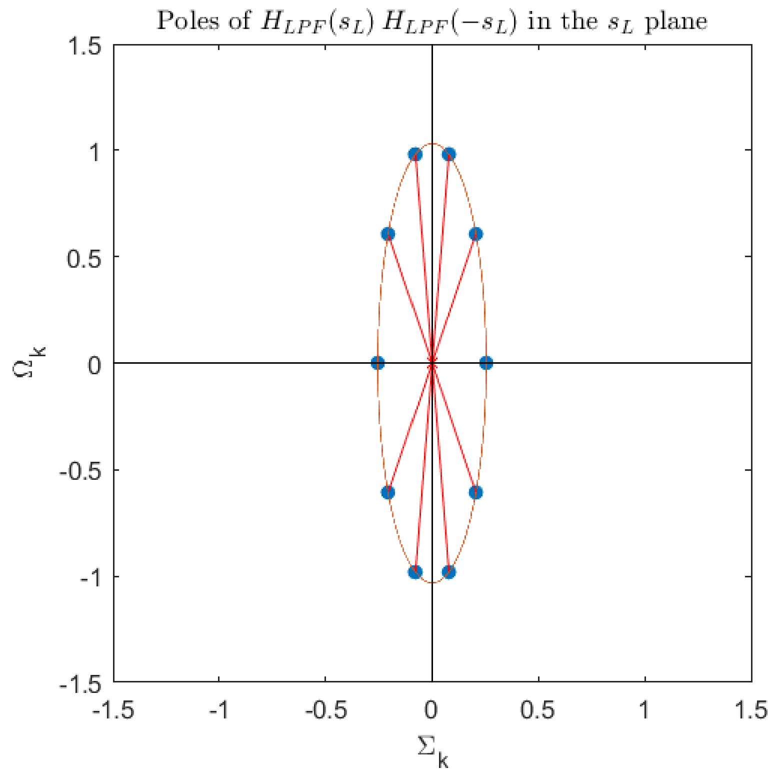

Figure 10: Poles in the s_L plane

The poles which lies in the left half plane are of our interest because only LHP ensures stability of our system and hence makes our filter achievable.

Table 6: Left Plane Poles

Pole	Value
s_1	-0.0785 + 0.9812i
s_2	-0.2054 + 0.6064i
s_3	-0.2539 + 0.0000i
s_4	-0.2054 - 0.6064i
s_5	-0.0785 - 0.9812i

Using the poles given in the tabel 6,the equivalent lowpass transfer can be written as

$$H_{LPF}(s_L) = \frac{A}{\prod_{k=1}^{N}(s_L - s_k)}$$

We need 1-δ_1 magnitude response at the passband edge,by applying the above condition we have the final analog low pass transfer function as

$$H_{LPF}(s_L) = \frac{\prod_{k=1}^{5} s_k}{\prod_{k=1}^{5}(s_L - s_k)}$$

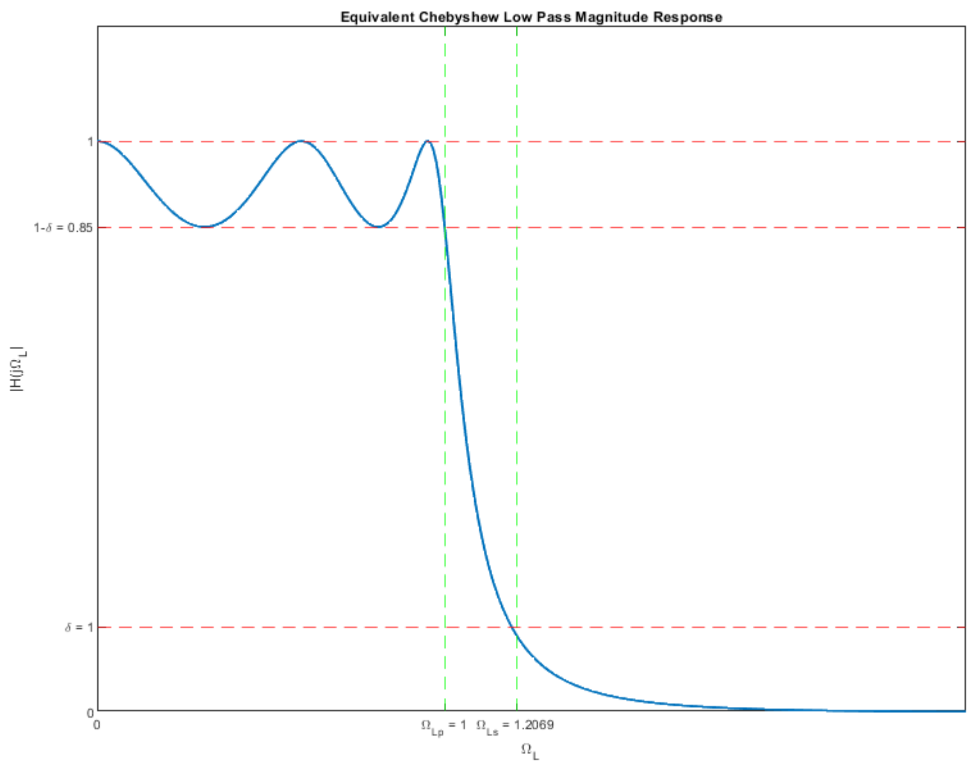

Figure 11: Magnitude Response

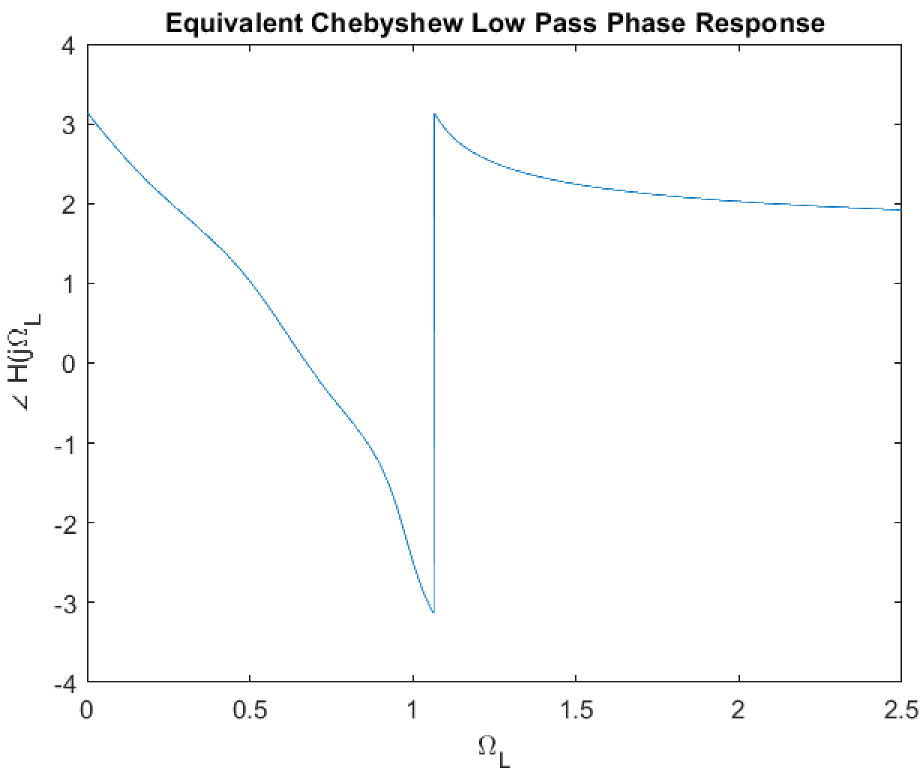

Figure 12: Phase Response

3.7 Analog Bandstop Transfer Function

We will now use inverse transformation to get the bandstop response from the chebyshew low pass response by using the following transformation

$$s_L \longleftarrow \frac{Bs}{s^2 + \Omega_0^2}$$

Where Ω and B are the values we found in section 3.4

$$H_{BSF} = H_{LPF}(\frac{Bs}{s^2 + \Omega_0^2})$$

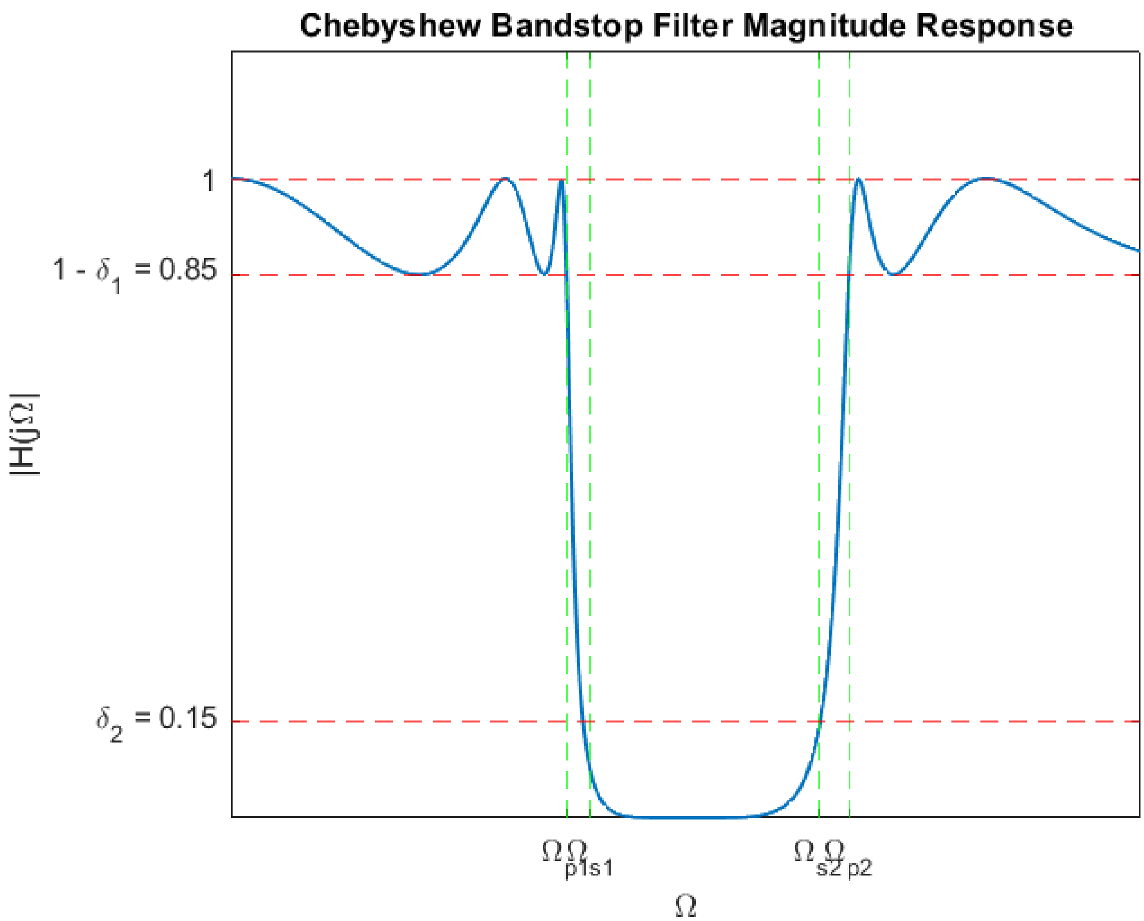

Figure 13: Magnitude Response

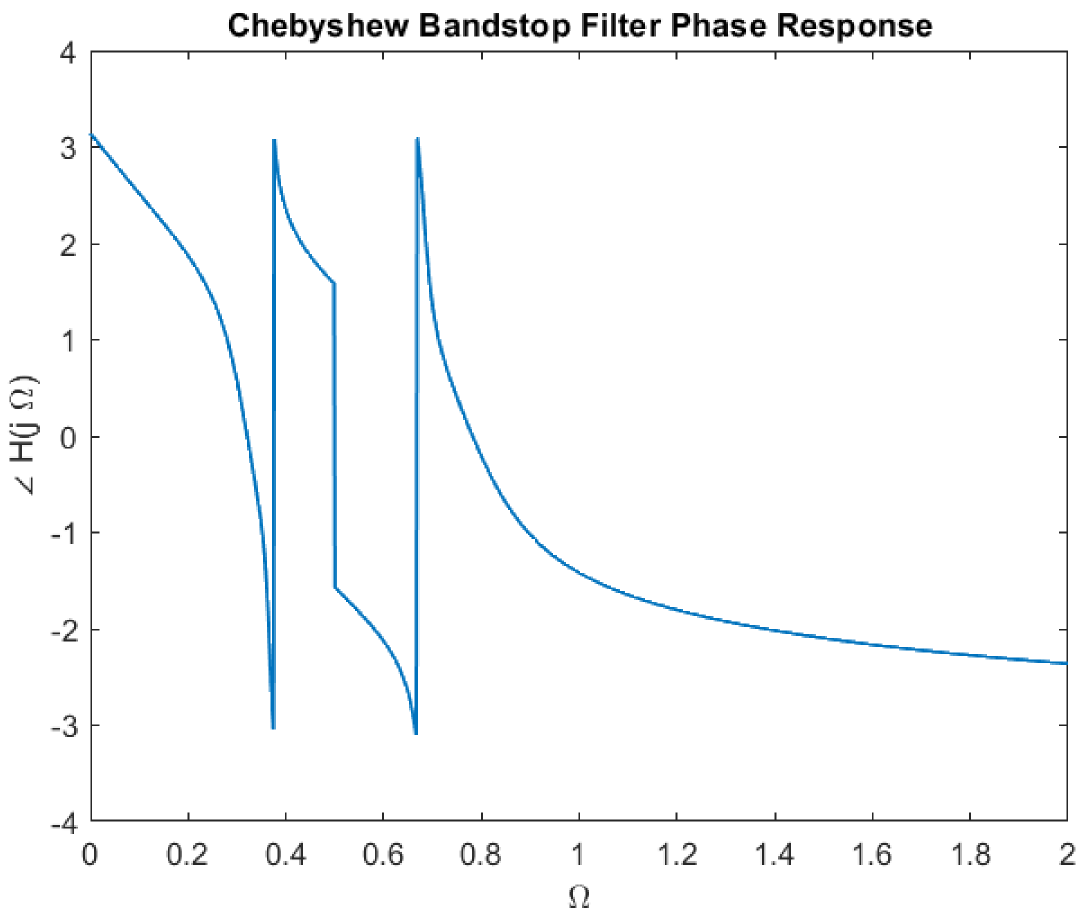

Figure 14: Phase Response

3.8 Discrete time Bandstop Transformation

Now we will convert the analog bandpass filter into the discrete bandpass filter by using the bilinear transformation in the normalized angular frequency domain.

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

The discrete time BSF magnitude and phase response

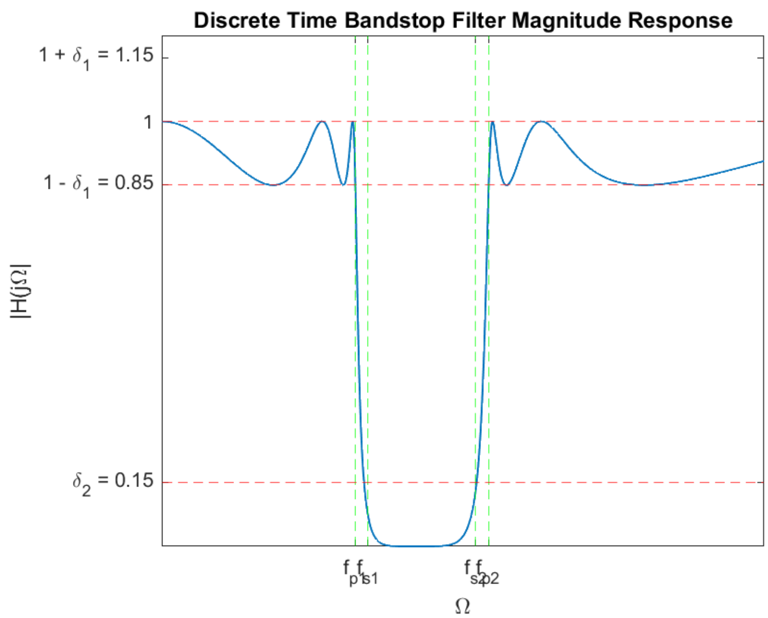

Figure 15: Magnitude Response

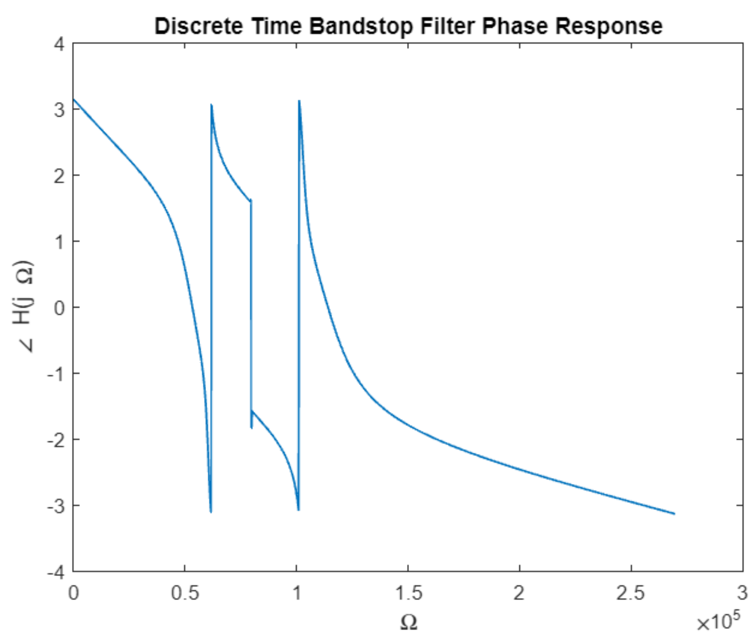

Figure 16: Phase Response

The normalized coefficients for the discrete time filter are shown below:

Table 7: Coefficient Table

Coefficient	Value	Coefficient	Value
a_{10}	-0.0546	a_{10}	0.1669
a_9	0.3271	a_9	-0.7920
a_8	-1.0570	a_8	2.0058
a_7	2.2479	a_7	-3.3639
a_6	-3.4606	a_6	4.0887
a_5	3.9763	a_5	-3.6942
a_4	-3.4606	a_4	2.4901
a_3	2.2479	a_3	-1.2109
a_2	-1.057	a_2	0.3918
a_1	0.3271	a_1	-0.0654
a_0	-0.0546	a_0	0.0011

4 FIR Bandstop filter

Finite Impulse Response Bandpass filter is designed by approximating the infinitely long impulse response with a finite impulse response by using windowing methods.Here we will use the **Kaiser Window** to implement the above method which is characterized by the width(M) and shape(β)
The normalized specification are as follows:

- $\omega_{p2} = 0.7068$
- $\omega_{p1} = 0.7539$
- $\omega_{s1} = 1.1466$
- $\omega_{s2} = 1.1938$
- Transition Bandwidth($\Delta\omega_t$) = 0.0471π
- Passband and Stopband Tolernce(δ) = 0.15

We will use the kaiser window and multiply it with the impulse response of the ideal bandstop filter to get the FIR filter.

4.1 Implementation Method

The mean of the lower passband and lower stopband is taken as the lower stopband edge of the ideal bandpass filter while the mean of the upper passband and the upper stopband is taken as the upper stopband edge of the BS filter

4.2 Kaiser Window Parameters

The value of the parameter is less than 21 and therefore $\alpha = 0$ and so the shaping parameter β and therefore, the kaiser window will be rectangular in shape

$$A = -20\log(\delta) = 16.47817$$

Window width M is

$$M \geq 1 + \frac{A - 8}{2.285\Delta\omega_t} = 79.73$$

I will take odd value of M:

$$M = 93$$

4.3 Magnitude and Phase Response

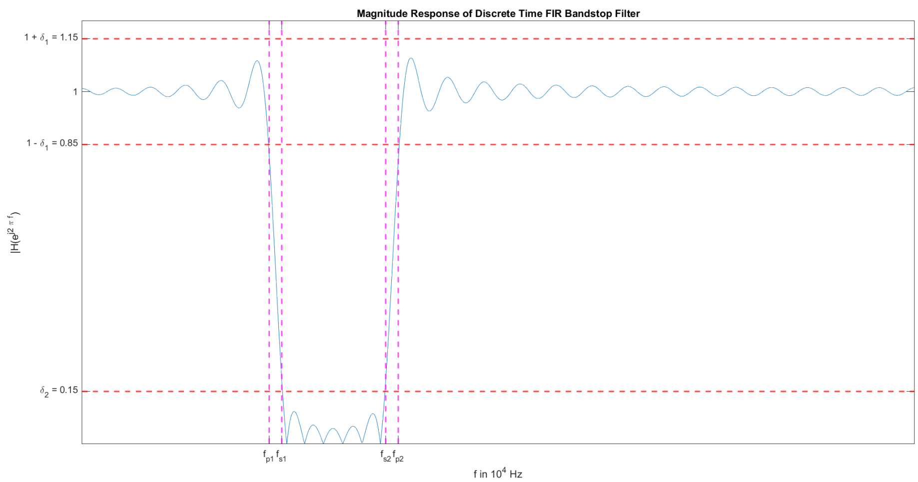

Figure 17: Magnitude Response of the FIR Bandstop Filter

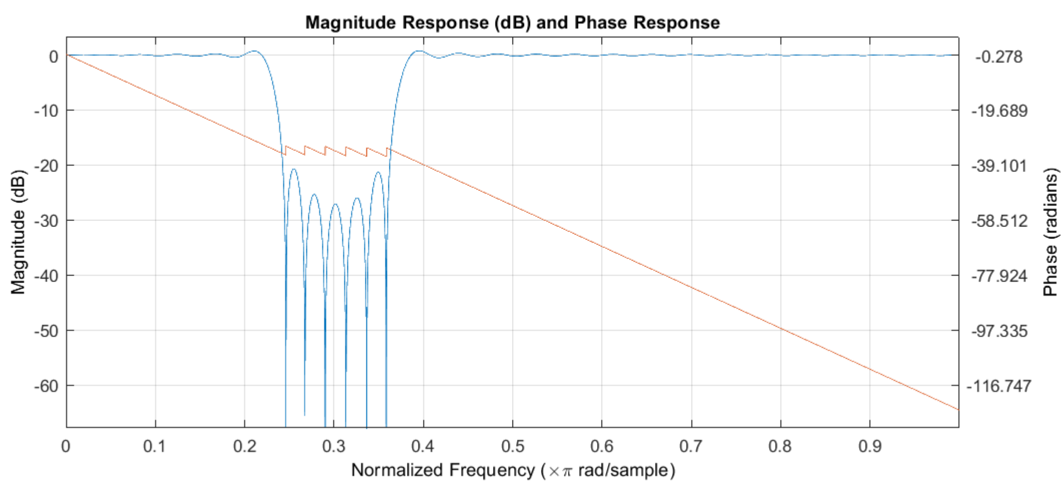

Figure 18: Phase Response of the FIR Bandstop Filter

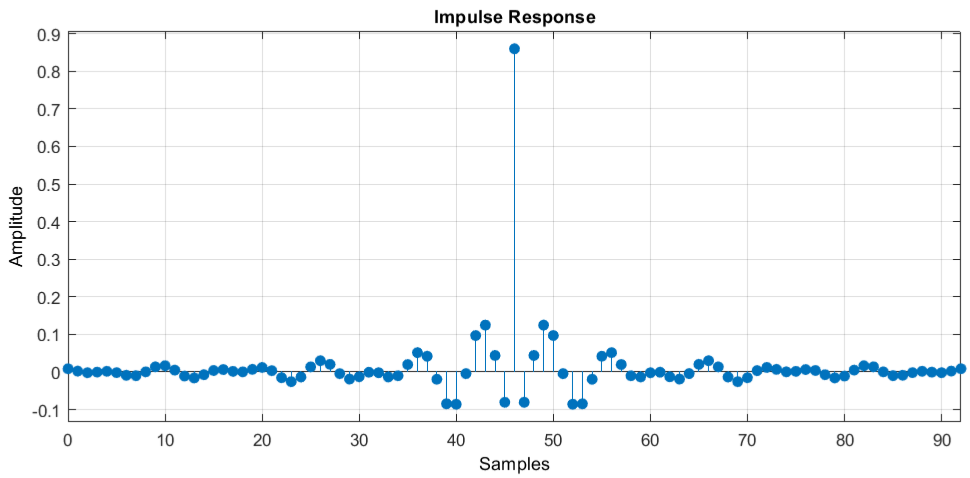

Figure 19: Impulse Response of the FIR Bandstop Filter

```
Columns 1 through 13

  0.0085    0.0022   -0.0020   -0.0005    0.0017   -0.0019   -0.0089   -0.0098    0.0002    0.0136    0.0166    0.0049   -0.0109

Columns 14 through 26

 -0.0159   -0.0073    0.0039    0.0064    0.0016    0.0003    0.0069    0.0120    0.0035   -0.0153   -0.0258   -0.0134    0.0135

Columns 27 through 39

  0.0299    0.0202   -0.0044   -0.0190   -0.0128   -0.0008   -0.0021   -0.0134   -0.0102    0.0197    0.0513    0.0418   -0.0194

Columns 40 through 52

 -0.0848   -0.0859   -0.0045    0.0969    0.1246    0.0439   -0.0807    0.8600   -0.0807    0.0439    0.1246    0.0969   -0.0045

Columns 53 through 65

 -0.0859   -0.0848   -0.0194    0.0418    0.0513    0.0197   -0.0102   -0.0134   -0.0021   -0.0008   -0.0128   -0.0190   -0.0044

Columns 66 through 78

  0.0202    0.0299    0.0135   -0.0134   -0.0258   -0.0153    0.0035    0.0120    0.0069    0.0003    0.0016    0.0064    0.0039

Columns 79 through 91

 -0.0073   -0.0159   -0.0109    0.0049    0.0166    0.0136    0.0002   -0.0098   -0.0089   -0.0019    0.0017   -0.0005   -0.0020

Columns 92 through 93

  0.0022    0.0085
```

Figure 20: Coefficients of FIR filter

5 Elliptical Bandpass Filter

5.1 Unnormalized Specifications

- Upper Passband edge = 116kHz
- Lower Passband edge = 71kHz
- Passband tolerance = 0.15
- Stopband tolerance = 0.15
- Transition bandwidth = 3kHz
- Sampling frequency = 540kHz
- Passband nature = Equiripple
- Stopband nature = Equiripple

5.2 Normalized Specifications

The sampling angular frequency 2πfs should map to 2π after normalization. Hence the transformation used is

$$\omega = 2\pi \frac{f}{f_s}$$

Table 8: Specifications

Parameter	Value
ω_{p2}	0.4296π
ω_{p1}	0.2629π
ω_{s1}	0.2518π
ω_{s2}	0.4407π

- Passband and Stopband tolerance = 0.15
- Passband and Stopband nature = Equiripple

5.3 Analog Filter Specifications using Bilinear Transform

We use the following transformation to move from the s domain to the z domain.

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

If we put z = jω and s = jΩ, we get

$$\Omega = tan(\frac{\omega}{2})$$

The filter specifications now are as follows

Table 9: Specification Table

Parameter	Value
Ω_{p2}	0.80012
Ω_{p1}	0.4381
Ω_{s1}	0.4175
Ω_{s2}	0.8291
δ_1	0.15
δ_2	0.15

5.4 Analog Frequency Transformation

Now we will do Bandpass to Lowpass frequency transformation:

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

Where

$$\Omega_0 = \sqrt{\Omega_{p1}\Omega p2} = 0.592$$

and

$$B = \Omega_{p2} - \Omega p1 = 0.362$$

Using the above transformation we have

- Ω_{Ls1} = -1.1655
- Ω_{Ls2} = 1.1226
- Ω_{Lp1} = -1
- Ω_{Lp2} = +1
- Passband and Stopband nature = Equiripple

5.5 Equivalent Analog Lowpass Filter Specifications

Since we have done the analog frequency transformation above,now we will find the low pass filter specifications by using the most stringent conditions

- Passband and Stopband nature = Monotonic
- Ω_{Ls} = min(Ω_{Ls2},$|\Omega_{Ls1}|$) = 1.1226
- Ω_{Lp} = +1
- $\delta_1 = \delta_2 = 0.15$
- $D_1 = \frac{1}{(1-\delta_1)^2}$-1 = 0.384
- $D_2 = \frac{1}{\delta_2^2}$-1 = 43.44

5.6 Analog LPF Magnitude response

The Transfer function for an elliptic filter is given by

$$H_{LPF}(s_L)H_{LPF}(-s_L) = \frac{1}{1+\epsilon_p^2 F_N^2(\frac{s_L}{J\omega_p})}$$

Here N is the filter order and ϵ_p is the passband ripple factor with

$$F_N(\omega) = cd(NuK1, k1)$$

and

$$\omega = cd(uK, k)$$

where cd(x, k) denotes the Jacobian elliptic function cd with modulus k and real quarterperiod K

5.7 Jacobian Elliptic Functions

The elliptic function ω = sn(z, k) can be defined through the elliptic integral:

$$z = \int_0^{\phi} \frac{d\theta}{\sqrt{1 - k^2 sin^2(\theta)}}$$

$$z = \int_0^{\omega} \frac{dt}{\sqrt{(1 - k^2 t^2)(1 - t^2)}}$$

Where

$$\omega = sin(\phi(z, k))$$

The parameter k is called the elliptic modulus and is assumed to be a real number in the interval 0 $\leq$ k $\leq$ 1.
The three elliptic functions cn,dn,cd are defined as:

$$\omega = cn(z, k) = cos\phi(z, k)$$

$$\omega = dn(z, k) = cos\phi(z, k) = \sqrt{1 - k^2 sn^2(z, k}$$

$$\omega = cd(z, k) = \frac{cn(z, k)}{dn(z, k)}$$

The complete elliptical integral denoted by K(k) or K is defined as the value of z at $\phi = \pi/2$

$$K = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1 - k^2 sin^2(\theta)}}$$

At $\phi = \pi/2$ we have:

$$sn(K, k) = 1 \ \ and \ \ cd(K, k) = 0$$

Now we will define complementary elliptic modulus $k' = \sqrt{1 - k^2}$ and the associated complete elliptic integral as

$$K' = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1 - k'^2 sin^2(\theta)}}$$

5.8 Elliptic Filter Specification

The order of the ellliptic filter can be found the applying the constraints on the magnitude response at the passband and stopband.

$$N = \lceil \frac{KK_1'}{K'K_1} \rceil$$

Where K, K_1 are the complete elliptic integrals corresponding to the moduli k, k_1, and K' , K_1' are the complete elliptic integrals corresponding to the complementary moduli k' and k_1' For the given specifications of the Bandpass filter we have

$$k = \frac{\Omega_{Lp}}{\Omega_{Ls}} = 0.8908$$

$$k_1' = \frac{\epsilon_p}{\epsilon_s} = \sqrt{\frac{D_2}{D_1}} = 0.094$$

By applying the Jacobian for the above values of modulli we have

Parameter	Value
K	2.2427
K'	1.6629
K_1	1.5743
K_1'	3.7566

The poles and zeroes of the transfer function can be calculated by using the following equations:

$$\begin{gathered} z_i = j\Omega_{Lp}(k\zeta^{-1}) \quad i = 1, 2, 3, ...L \\ p_i = j\Omega_{Lp} cd((u_i - jv_o)K, k) \quad i = 1, 2, 3, ...L \\ p_o = j\Omega_{Lp} sn(jv_o K, k) \quad i = 1, 2, 3, ...L \end{gathered} \tag{1}$$

Where

$$u_i = \frac{2i-1}{N} \quad i = 1, 2, 3, ...L$$

$$\zeta_i = cd(u_i K, k) \quad i = 1, 2, 3, ...L$$

$$v_o = \frac{-j}{NK_1} sn^{-1}(\frac{j}{\epsilon_p}, k_1)$$

$$L = \lfloor \frac{N}{2} \rfloor$$

The zeros z_i are the poles of the $F_N(\omega)$ and the poles p_i are the zeroes of the denominator i.e. $1 + \epsilon_p^2 F_N^2(\omega) = 0$.
For the odd filter order N we have an additional real valued left hand s-plane pole p_o which can be obtained from eq 2
I get $\mathbf{N = 4}$ for the above filter specifications and the poles and zeroes of the lowpass filter are

Zeroes	Poles
0.0000 + 1.0639i	-0.3536 - 0.7071i
0.0000 + 1.7690i	-0.3536 + 0.7071i
0.0000 - 1.7690i	-0.0310 - 0.9995i
0.0000 - 1.0639i	-0.0310 + 0.9995i

$$H_LPF(s_L)H_LPF(-s_L) = \frac{0.85 * [(1 + 0.8835 * sl^2) * (1 + 0.3196 * sl^2)]}{(1 + 0.0619 * sl + sl^2) * (1 + 1.1314 * sl + 1.5998 * sl^2)}$$

NOTE:
I have updated the value of the k' by using the ellipdeg function in the matlab code.This function do the following operation:

$$k' = (k_1')^N \prod_{i=1}^{L} sn^4(u_i K_1', k_1')$$

By updating the value of the k' the filter response is more stringent in the stopband only.So if we do not update the value of k' the response will remain same but the magnitude in the stopband is more stringent

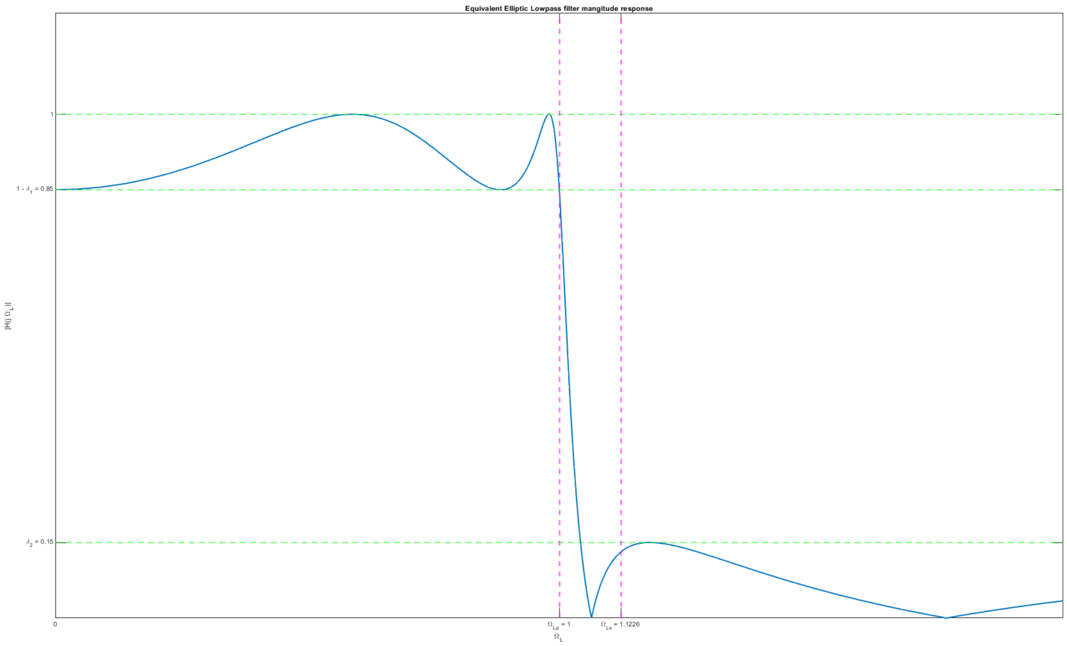

Figure 21: Magnitude Response

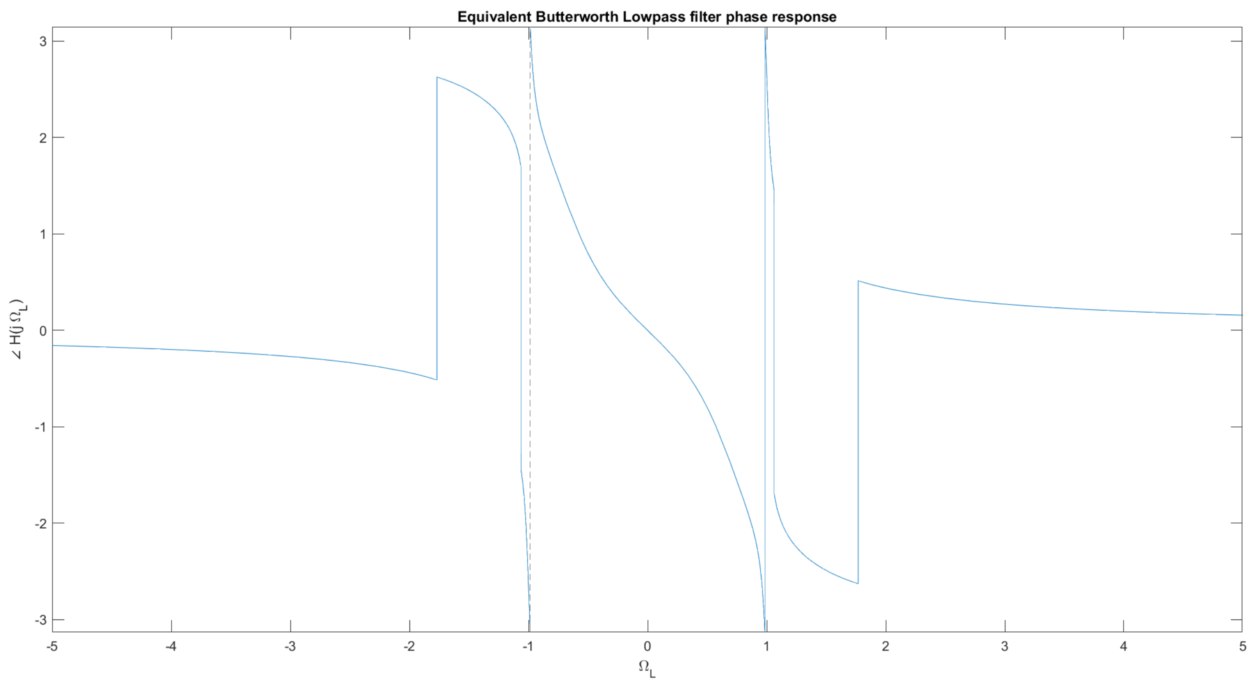

Figure 22: Phase Response

5.9 Analog Bandpass Transfer Function

We now use the inverse transformation to convert the equivalent Elliptic lowpass transfer function to get the Bandpass transfer function using the relation:

$$s_L \longleftarrow \frac{s^2 + \Omega^2}{Bs}$$

Where Ω and B are the values we found in section 5.4

$$H_{BFP} = H_{LPF}(\frac{s^2 + \Omega_0^2}{Bs})$$

$$H_{BPF} = \frac{0.15s^8 + 0.2941s^6 + 0.1784s^4 + 0.0361s^2 + 0.0023}{s^8 + 0.2784s^7 + 1.6205s^6 + 0.3281s^5 + 0.9010s^4 + 0.1150s^3 + 0.1990s^2 + 0.012s + 0.0151}$$

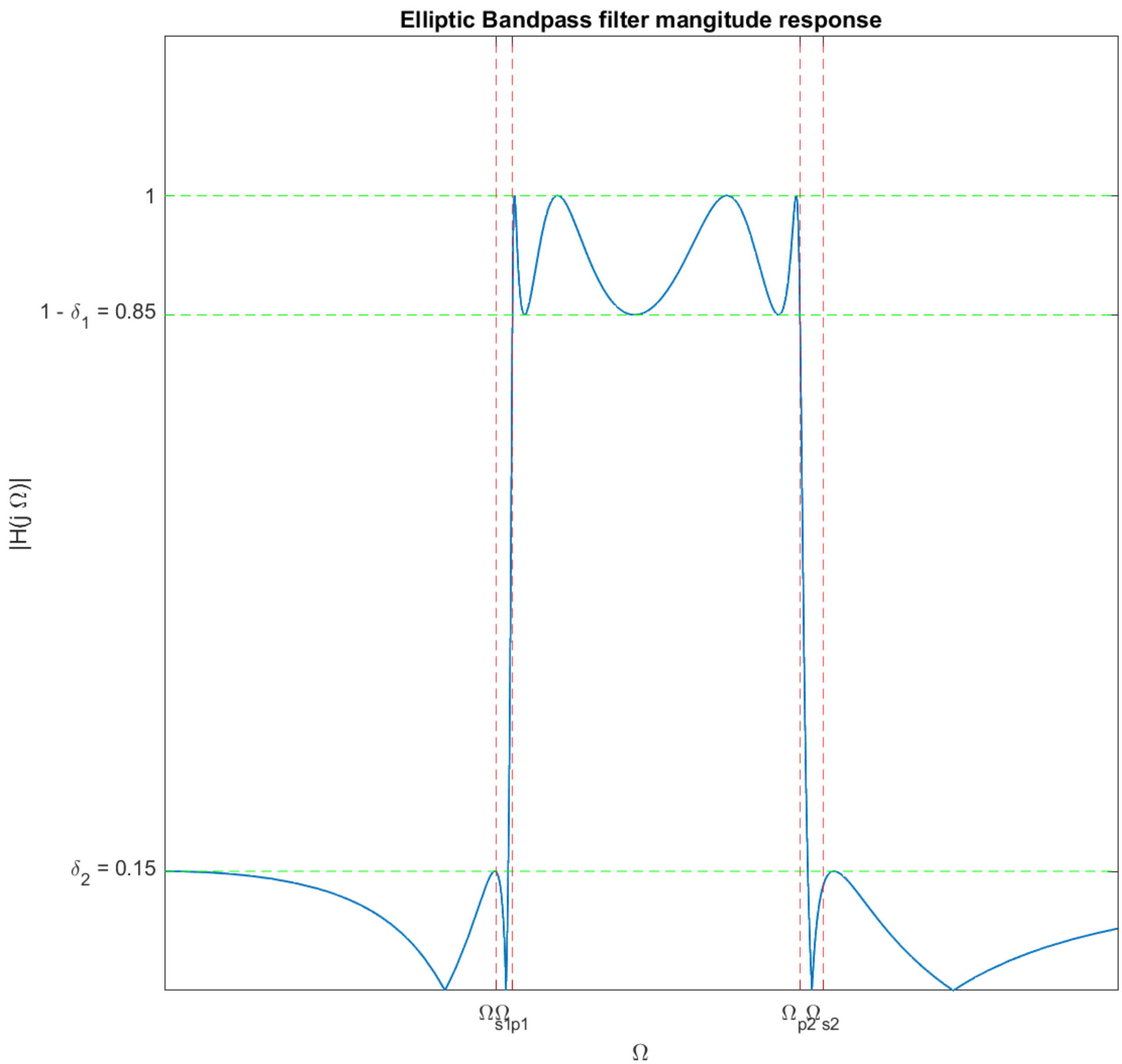

Figure 23: Magnitude Response

5.10 Discrete time Bandpass Transformation

Now we will convert the analog bandpass filter into the discrete bandpass filter by using the bilinear transformation in the normalized angular frequency domain.

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

The discrete time BPF transfer function is

$$H_{BPF}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)$$

$$H_{BPF} = \frac{0.1479z^8 - 0.4954z^7 + 1.09z^6 - 1.620z^5 + 1.886z^4 - 1.62z^3 + 1.09z^2 - 0.495z + 0.1479}{z^8 - 3.488z^7 + 7.893z^6 - 11.617z^5 + 13.037z^4 - 10.52z^3 + 6.47z^2 - 2.5823z + 0.6717}$$

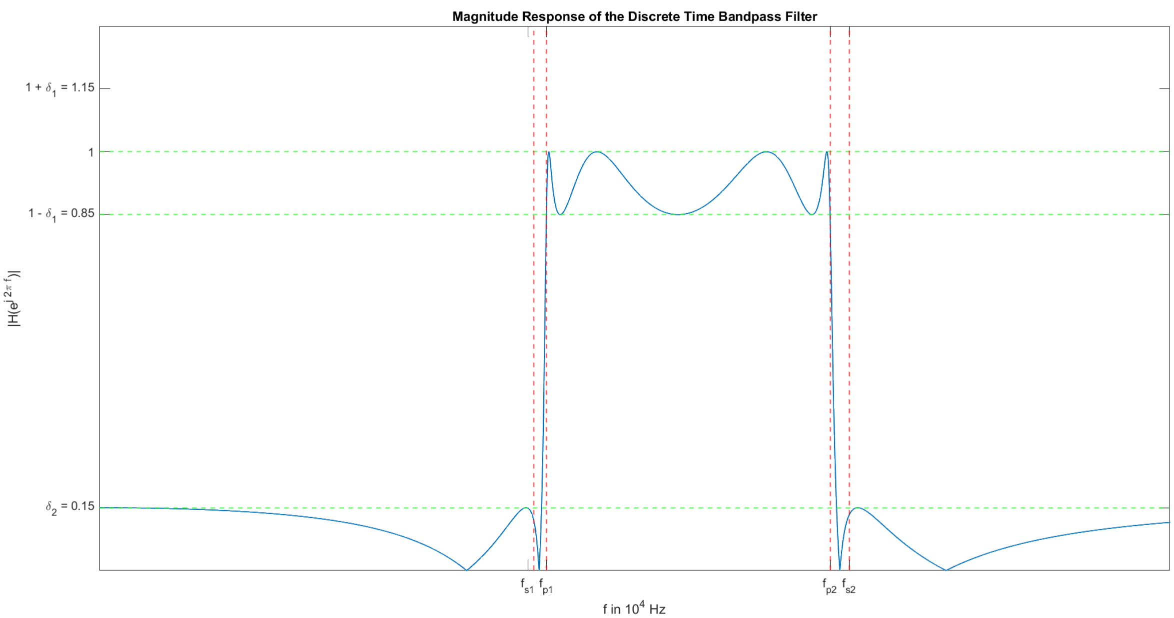

Figure 24: Magnitude Response

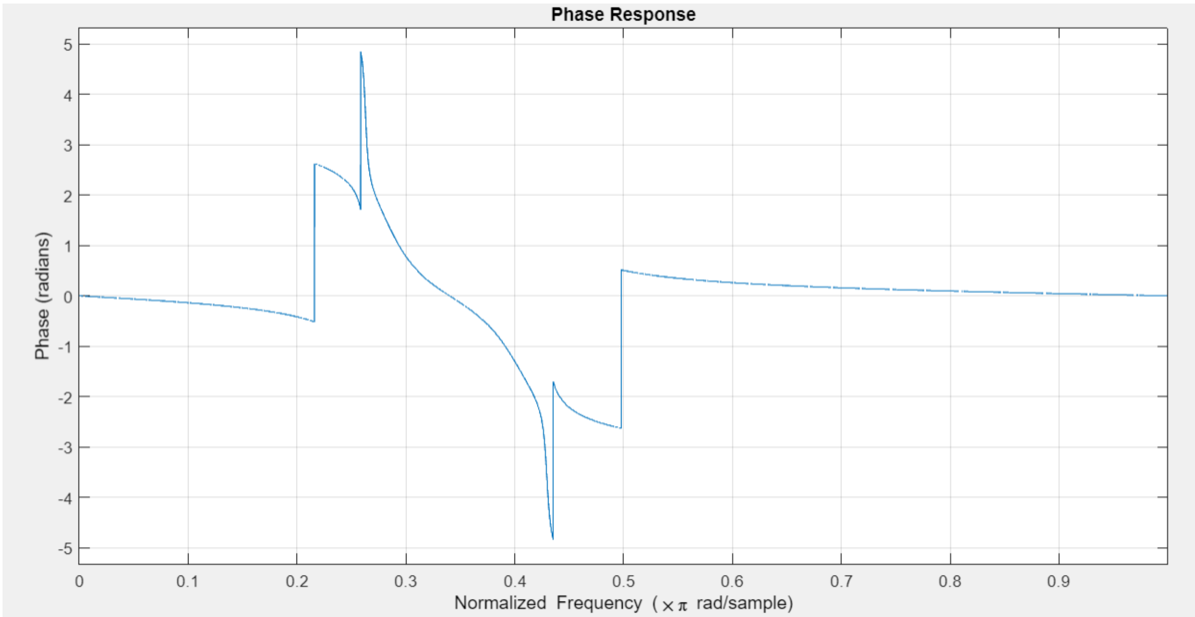

Figure 25: Phase Response

6 Elliptical Bandstop Filter

6.1 Unnormalized Specifications

- Upper Stopband edge = 73kHz
- Lower Stopband edge = 48kHz
- Passband tolerance = 0.15
- Stopband tolerance = 0.15
- Transition bandwidth = 3kHz
- Sampling frequency = 400kHz
- Passband nature = Equiripple
- Stopband nature = Equiripple

6.2 Normalized Specifications

The sampling angular frequency 2πfs should map to 2π after normalization. Hence the transformation used is

$$\omega = 2\pi\frac{f}{f_s}$$

Parameter	Value
ω_{p1}	0.7068
ω_{s1}	0.7539
ω_{s2}	1.1466
ω_{p2}	1.1938

- Passband and Stopband tolerance = 0.15
- Stopband nature = Equiripple
- Passband nature = Equiripple

6.3 Analog Filter Specifications using Bilinear Transform

We use the following transformation to move from the s domain to the z domain.

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

If we put z = jω and s = jΩ, we get

$$\Omega = tan(\frac{\omega}{2})$$

The filter specifications now are as follows

Parameter	Value
Ω_{p1}	0.3689
Ω_{s1}	0.3959
Ω_{s2}	0.6457
Ω_{p2}	0.6795
δ_1	0.15
δ_2	0.15

6.4 Analog Frequency Transformation

Now we will do Bandpass to Lowpass frequency transformation:

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

Where

$$\Omega_0 = \sqrt{\Omega_{p1}\Omega p2} = 0.5007$$

and

$$B = \Omega_{p2} - \Omega p1 = 0.31067$$

Using the above transformation we have

- Ω_{Ls1} = -1.30916
- Ω_{Ls2} = 1.20699
- Ω_{Lp1} = -1
- Ω_{Lp2} = +1
- Stopband nature = Equiripple
- Passband nature = Equiripple

6.5 Equivalent Analog Lowpass Filter Specifications

Since we have done the analog frequency transformation above,now we will find the low pass filter specifications by using the most stringent conditions

- Passband and Stopband nature = Equiripple
- $\Omega_{Ls} = \min(\Omega_{Ls2}, |\Omega_{Ls1}|) = 1.2069$
- $\Omega_{Lp} = +1$
- $\delta_1 = \delta_2 = 0.15$
- $D_1 = \frac{1}{(1-\delta_1)^2}$-1 = 0.384
- $D_2 = \frac{1}{\delta_2^2}$-1 = 43.44

6.6 Analog LPF Magnitude response

The Transfer function for an elliptic filter is given by

$$H_{LPF}(s_L)H_{LPF}(-s_L) = \frac{1}{1 + \epsilon_p^2 F_N^2(\frac{s_L}{J\omega_p})}$$

Here N is the filter order and ϵ_p is the passband ripple factor with

$$F_N(\omega) = cd(NuK1, k1)$$

and

$$\omega = cd(uK, k)$$

where cd(x, k) denotes the Jacobian elliptic function cd with modulus k and real quarterperiod K

6.7 Jacobian Elliptic Integrals

The elliptic function ω = sn(z, k) can be defined through the elliptic integral:

$$z = \int_0^{\phi} \frac{d\theta}{\sqrt{1 - k^2 sin^2(\theta)}}$$

$$z = \int_0^{\omega} \frac{dt}{\sqrt{(1-k^2t^2)(1-t^2)}}$$

Where

$$\omega = sin(\phi(z,k))$$

The parameter k is called the elliptic modulus and is assumed to be a real number in the interval 0 ≤ k ≤ 1.
The three elliptic functions cn,dn,cd are defined as:

$$\omega = cn(z,k) = cos\phi(z,k)$$

$$\omega = dn(z,k) = cos\phi(z,k) = \sqrt{1-k^2sn^2(z,k}$$

$$\omega = cd(z,k) = \frac{cn(z,k)}{dn(z,k)}$$

The complete elliptical integral denoted by K(k) or K is defined as the value of z at $\phi = \pi/2$

$$K = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1-k^2sin^2(\theta)}}$$

At $\phi = \pi/2$ we have:

$$sn(K,k) = 1 \ \ and \ \ cd(K,k) = 0$$

Now we will define complementary elliptic modulus $k' = \sqrt{1-k^2}$ and the associated complete elliptic integral as

$$K' = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1-k'^2sin^2(\theta)}}$$

6.8 Elliptic Filter Specification

The order of the ellliptic filter can be found the applying the constraints on the magnitude response at the passband and stopband.

$$N = \lceil \frac{KK_1'}{K'K_1} \rceil$$

Where K, K_1 are the complete elliptic integrals corresponding to the moduli k, k_1, and K' , K_1' are the complete elliptic integrals corresponding to the complementary moduli k' and k_1' For the given specifications of the Bandstop filter we have

$$k = \frac{\Omega_{Lp}}{\Omega_{Ls}} = 0.8286$$

$$k_1' = \frac{\epsilon_p}{\epsilon_s} = \sqrt{\frac{D_2}{D_1}} = 0.094$$

By applying the Jacobian for the above values of modulli we have The poles

Parameter	Value
K	2.056
K'	1.7219
K_1	1.5743
K_1'	3.7566

and zeroes of the transfer function can be calculated by using the following equations:

$$\begin{aligned} z_i &= j\Omega_{Lp}(k\zeta^{-1}) \quad i = 1, 2, 3, ...L \\ p_i &= j\Omega_{Lp} cd((u_i - jv_o)K, k) \quad i = 1, 2, 3, ...L \\ p_o &= j\Omega_{Lp} sn(jv_o K, k) \quad i = 1, 2, 3, ...L \end{aligned} \tag{2}$$

Where

$$u_i = \frac{2i-1}{N} \quad i = 1, 2, 3, ...L$$

$$\zeta_i = cd(u_i K, k) \quad i = 1, 2, 3, ...L$$

$$v_o = \frac{-j}{NK_1} sn^{-1}(\frac{j}{\epsilon_p}, k_1)$$

$$L = \lfloor \frac{N}{2} \rfloor$$

The zeros z_i are the poles of the $F_N(\omega)$ and the poles p_i are the zeroes of the denominator i.e. $1 + \epsilon_p^2 F_N^2(\omega) = 0$.
For the odd filter order N we have an additional real valued left hand s-plane pole p_o which can be obtained from eq 2
I get $\mathbf{N = 3}$ for the above filter specifications and the poles and zeroes are

Zeroes	Poles
0.0000 + 1.2604i	-0.1153 - 0.9936i
0.0000 - 1.2604i	-0.1153 + 0.9936i
	-0.6232 + 0.0000i

$$H_LPF(s_L)H_LPF(-s_L) = \frac{(1 + 0.6295 * sl^2)}{(1 + 1.6047 * sl) * (1 + 0.2305 * sl + 0.9994 * sl^2)}$$

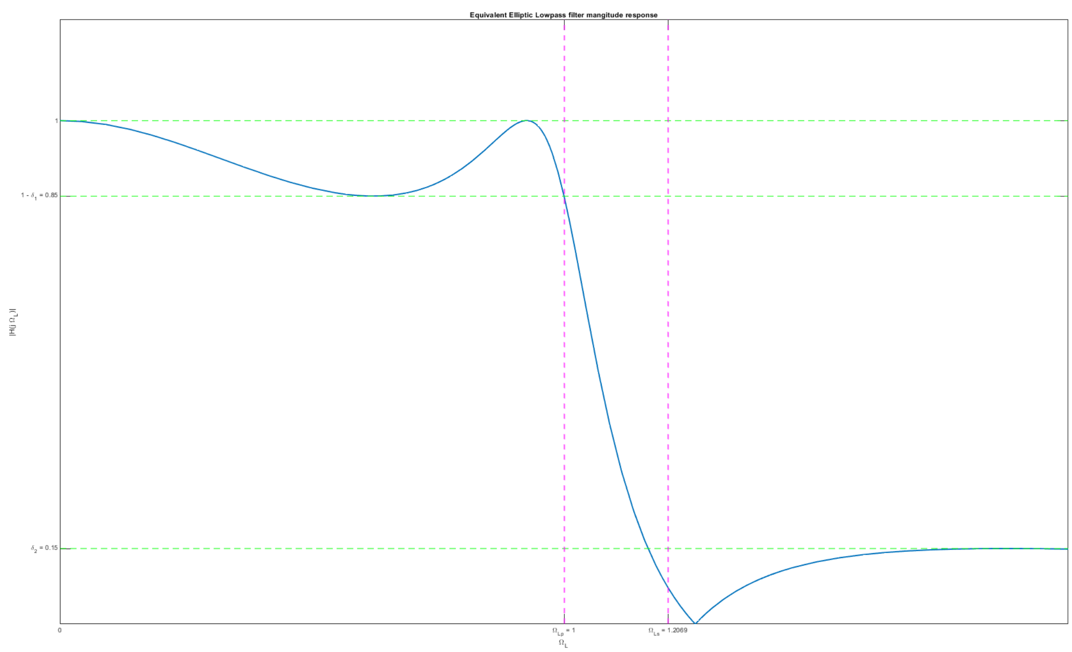

Figure 26: Magnitude Response

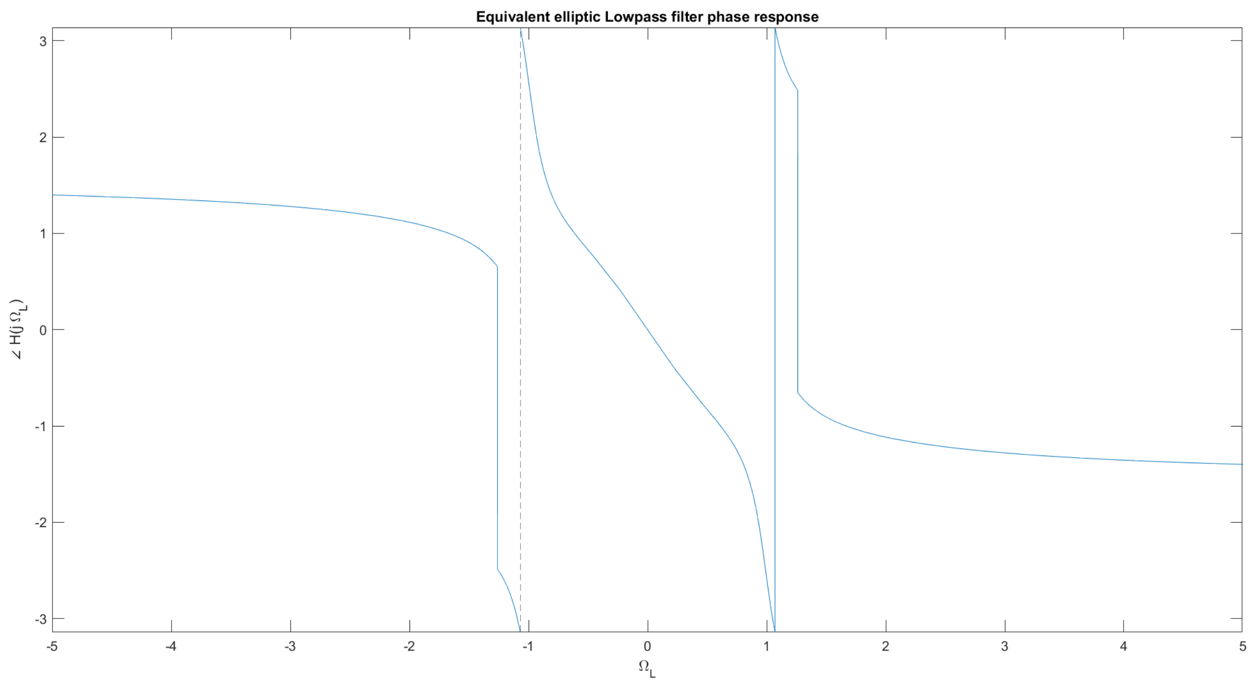

Figure 27: Phase Response

6.9 Analog Bandstop Transfer Function

We now use the inverse transformation to convert the equivalent Elliptic lowpass transfer function to get the Bandstop transfer function using the relation:

$$s_L \longleftarrow \frac{Bs}{s^2 + \Omega_0^2}$$

Where Ω and B are the values we found in section 5.4

$$H_{BSF} = H_{LPF}(\frac{Bs}{s^2 + \Omega_0^2})$$

$$H_{BPF} = \frac{s^6 + 0.8129s^4 + 0.2038s^2 + 0.01576}{s^6 + 0.5701s^5 + 0.8843s^4 + 0.334s^3 + 0.2217s^2 + 0.03583s + 0.01576}$$

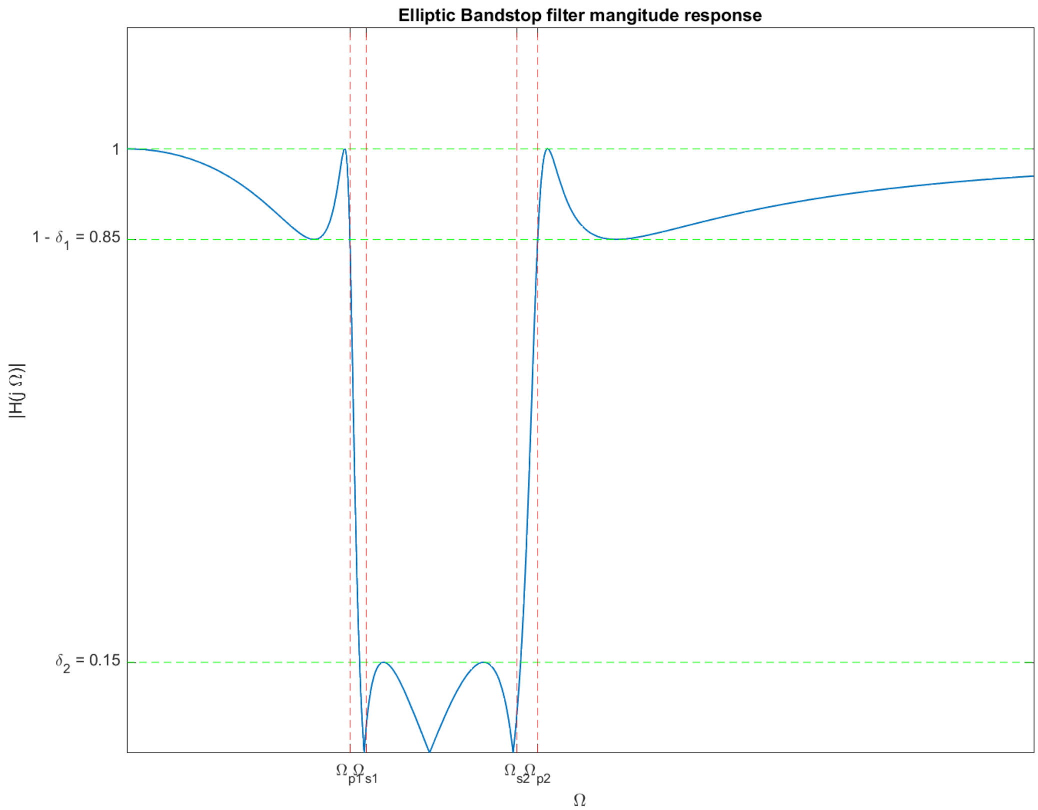

Figure 28: Magnitude Response

6.10 Discrete time Bandstop Transformation

Now we will convert the analog bandstop filter into the discrete bandstop filter by using the bilinear transformation in the normalized angular frequency domain.

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

The discrete time BSF transfer function is

$$H_{BSF}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right)$$

$$H_{BPF} = \frac{0.6638z^6 - 2.327z^5 + 4.644z^4 - 5.634z^3 + 4.644z^2 - 2.327z + 0.6638}{z^6 - 3.06z^5 + 5.278z^4 - 5.564z^3 + 3.953z^2 - 1.664z + 0.386}$$

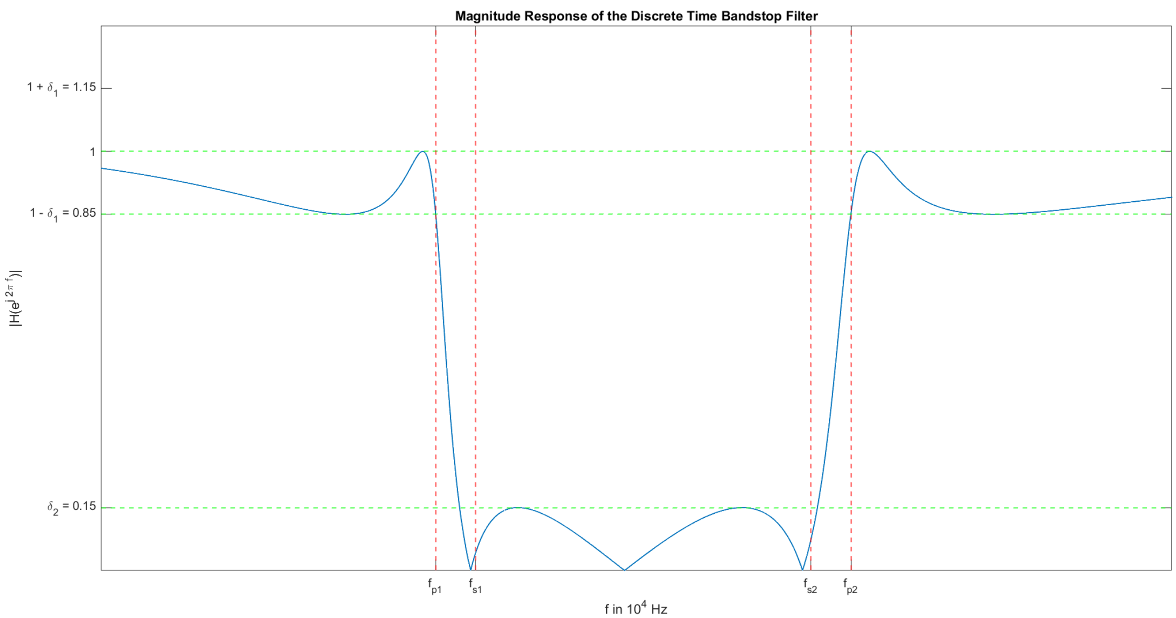

Figure 29: Magnitude Response

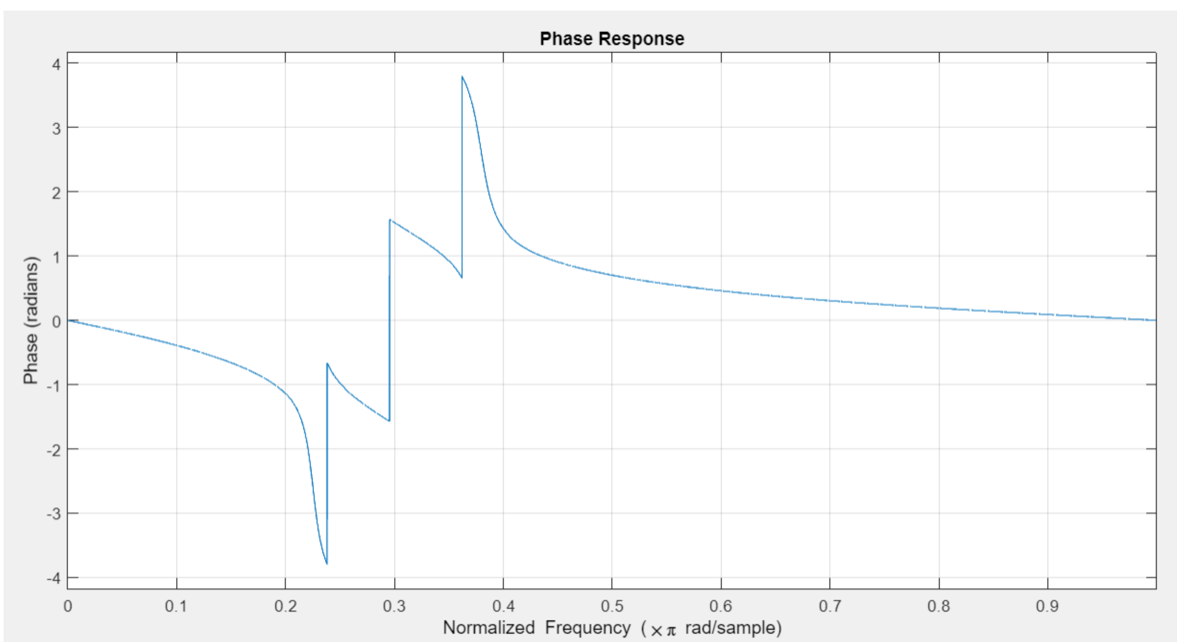

Figure 30: Phase Response

7 Comparison between IIR and FIR Filter

The FIR filters are always stable whereas IIR filters may be unsable.The FIR filters have a linear phase respose but IIR filters do not have a well defined phase response.IIR filters requires less number of taps for their implementation and have limited number of cycles when comapred with FIR filter.The order for the FIR design is ($\mathbf{N = 93}$) which is very high compared to the IIR order ($\mathbf{N = 21}$)

8 Comparison between Butterworth and Chebyshew Filter

The frequency response is monotonic in both passband and stopband but in case of chebyshew design we have equiripple in the passband edge.We can see from the magnitude response that butterworth filter has a slower roll-off and therefore it needs higher order to implement particular specifications.The phase response is more linear in the passband for butterworth filter.
Elliptic Filters are equiripple in both passband and stopband and elliptic filter requires lowest order to meet given specification in comparison to the other filters.For a given filter order,elliptic filters have minimum transition width between passband and stopband

- The use of elliptic approximation gives the minimum order for given specifications.The order calculated are as follows:
  Elliptic(4)<Chebyshew(5)<ButterWorth(21)<Kaiser-FIR(113)
- FIR design gives a linear phase response.The non-linearity in the phase response follows the given order:
  Elliptic>Chebyshew>ButterWorth>Kaiser FIR
- For the identical parameters, the elliptic filter has the sharpest transition from passband to stopband or vice versa. The diminishing order of transition band sharpness is:
  Elliptic>Chebyshev>Butterworth>Kaiser FIR

9 Peer Review

PEER REVIEW GIVEN TO BANDI DANY HEMANTH

I have **reviewed** the filter design report made by **Bandi Dany Hemanth**,I have checked that he has completed all the steps for designing the bandpass and bandstop filter by using IIR,FIR and elliptic methods but the bandpass IIR discrete response is not correct..The magnitude and phase response plots for IIR, FIR, and elliptic filters show that the specifications of the filters are satisfied.

PEER REVIEW REVIEWED FROM APOORVA HOTKAR

I have **reviewed** the filter design report made by **Vinay**,I have checked that he has completed all the steps for designing the bandpass and bandstop filter by using IIR,FIR and elliptic methods but the bandpass IIR discrete response is not correct.The magnitude and phase response plots for IIR, FIR, and elliptic filters show that the specifications of the filters are satisfied.

10 MATLAB Codes

10.1 BP-IIR FILTER

```
clc; close all;
tic;
%% Poles of H_{analog}(s_L) H_{analog}(-s_L)
N = 21;
Omega_c = 1.02462;
a = 1:2*N;
poles = Omega_c .* exp(1i .* (pi/2) .* (1 + (2.*a + 1)
    ./ N));
figure();
plot(poles, '*', 'MarkerSize', 10);
hold on;
x = linspace(-pi, pi, 10000);
a = Omega_c .* cos(x);
b = Omega_c .* sin(x);
plot(a,b);
hold on;
```

```
plot(0,0,'r*');
LHP = [];
for i=1:size(poles,2)
    hold on;
    plot([0, real(poles(1,i))], [0, imag(poles(1,i))], 'r-');
    if real(poles(1,i)) < 0
%         disp(poles(1,i));
        LHP = [LHP, poles(1,i)];
    end
end
plot([0, 0],[-1.5, 1.5], 'k-');
plot([-1.5, 1.5],[0, 0], 'k-');
title('Poles of $H_{LPF}(s_L) \, H_{LPF}(-s_L)$ in the $s_L$ plane', 'Interpreter', 'latex');
xlabel('\Sigma_k');
ylabel('\Omega_k');
%% Finding poles of H_{analog}(s_L)
syms s x;
temp = 1;
for i=1:size(LHP,2)
    temp = temp * (s - LHP(1,i));
end
%% Magnitude and Phase response of H_{analog}(s_L)
den_coeff = sym2poly(temp);
num = 1.665;
w = linspace(0,2,1000);
h = freqs(num,den_coeff,w);
figure();
plot(w,abs(h));
hold on;
fplot(s - s - 0.15 + 1, 'k-', 'Markersize', 10);
hold on;
fplot(x-x+1, 'k-', 'Markersize', 10);
hold on;
fplot(x - x + 0.15, 'r-', 'Markersize', 10);
xline(1, 'm-');
hold on;
```

```
xline(1.1226,'m--');
axis([0 2 0 1.2]);
set(gca,'Fontsize',5,'XTick',[0, 1, 1.1226], 'xticklabel', {'0', '\Omega_{Lp} = 1','\Omega_{Ls} = 1.1226'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
daspect([1 1 1]);
title('Equivalent Butterworth Lowpass filter mangitude response');
xlabel('\Omega_L');
ylabel('|H(j \Omega_L)|');
figure();
plot(w,angle(h));
xlabel('\Omega_L');
ylabel('\angle H(j \Omega_L)');
title('Equivalent Butterworth Lowpass filter phase response');




%% Bandpass analog frequency transformation
syms s1;
Omega_0 = 0.592;
B = 0.362;
s1 = (s^2 + Omega_0^2) ./ (B * s);
h = subs(temp, s, s1);
H = 1.6665/h;
%% Normalization of the coefficients
[num,den] = numden(H);
k = subs(num(1),s,1);   % numerator coefficient
num_coeff = sym2poly(num/k);
den_coeff = sym2poly(den/k);
%% magnitude and phase plot of BPF
w = linspace(0,2,1000);
[g,w] = freqs(num_coeff,den_coeff,w);
figure();
```

```
plot(w,abs(g));
set(gca,'Fontsize',4,'XTick', [0.4175, 0.4381, 0.80012, 0.8291], 'xticklabel', {'\Omega_{s1}', '\Omega_{p1}', '\Omega_{p2}', '\Omega_{s2}'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
hold on;
xline(0.4175, 'm--');hold on;
xline(0.4381, 'm--');hold on;
xline(0.80012, 'm--');hold on;
xline( 0.8291, 'm--');
hold on;
fplot(s - s - 0.15 + 1, 'k--', 'Markersize', 10);
hold on;
fplot(x-x+1, 'k--', 'Markersize', 10);
hold on;
fplot(x - x + 0.15, 'r--', 'Markersize', 10);
daspect([1 1 1]);
title('Butterworth Bandpass filter mangitude response');
xlabel('\Omega');
ylabel('|H(j \Omega)|');
axis([0 1.2 0 1.2]);

figure();
hold on;
plot(w,angle(g));
title('Butterworth Bandpass filter phase response');
xlabel('\Omega');
ylabel('\angle H(j \Omega)');

%% Discrete domain transformation
syms w;
c = subs(H, s, (1-1/w) / (1+1/w));
disp(c);
[num,den] = numden(c);
k = -subs(num(1),w,0);
num_coeff = sym2poly(num/k);
```

```
den_coeff = sym2poly(den/k);
% disp(num_coeff);
% disp(den_coeff);
w = linspace(0,2,1024);
[g,w] = freqz(num_coeff,den_coeff,1024,540e3);
figure();
plot(w,abs(g));
figure();
plot(w,angle(g));
```

10.2 BP-FIR FILTER

```matlab
clc; clear all; close all;
%% BP Filter Specifications
omega_s1 = 0.2518 * pi;
omega_s2 = 0.4407 * pi;
omega_p2 = 0.4296 * pi;
omega_p1 = 0.2629 * pi;
transition_bw = 0.0111 * pi;
delta = 0.15;
omega_c1 = (omega_p1 + omega_s1)/2;
omega_c2 = (omega_p2 + omega_s2)/2;
%% Kaiser window parameters
A = -20 * log10(delta);
min_width = ceil( 1 + ((A - 8)/(2.285 * transition_bw)));

alpha = -1;
if A < 21
    alpha = 0;
elseif A >=21 && A <= 50
    alpha = 0.5842 * (A - 21) ^ 0.4 + 0.07886 * (A - 21);
elseif A > 50
    alpha = 0.1102 * (A - 8.7);
else

end
beta = alpha /min_width;
%% kaiser window multiplication
M = min_width + 3;
K = kaiser(M,beta);
BPF_IDEAL = ideal_lpf(omega_c2, M) - ideal_lpf(omega_c1, M);
BPF_FIR= BPF_IDEAL .* K';
[H,f] = freqz(BPF_FIR,1,1024, 540e3);
figure();
plot(f, abs(H));
```

```
hold on;
xline(71e3, 'm--', 'LineWidth', 1.5);
hold on;
xline(116e3, 'm--', 'LineWidth', 1.5);
hold on;
xline(68e3, 'm--', 'LineWidth', 1.5);
hold on;
xline(119e3, 'm--', 'LineWidth', 1.5);
hold on;
yline(1.15, 'r--', 'LineWidth', 1.5);
hold on;
yline(0.85, 'r--', 'LineWidth', 1.5);
hold on;
yline(0.15, 'r--', 'LineWidth', 1.5);
xlabel('f in 10^4 Hz');
ylabel('|H(e^{j2 \pi f})');
title('Magnitude Response of Discrete Time FIR Bandpass Filter');
set(gca, 'XTick', [68e3, 71e3, 116e3, 119e3], 'xticklabel', {'f_{s1}', 'f_{p1}', 'f_{p2}', 'f_{s2}'});
set(gca, 'YTick', [0.15, 0.85, 1, 1.15], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1', '1 + \delta_1 = 1.15'});
%% displays the impulse and the phase response
fvtool(BPF_FIR, 'Analysis', 'Phase');
fvtool(BPF_FIR, 'Analysis', 'Impulse');

%% ideal lpf code
function lpass = ideal_lpf(wc,M);

alpha = (M-1)/2;
n = [0:1:(M-1)];
m = n - alpha + eps;
lpass = sin(wc*m) ./ (pi*m);
end
```

10.3 BS-IIR FILTER

```
clc; close all;
%% FILTER SPECIFICATIONS
wp1 = 0.706;
ws1 = 0.754;
ws2 = 1.146;
wp2 = 1.194;
op1= 0.3689194771098271;
os1 = 0.3959280087977212;
os2 = 0.645691821042554;
op2 = 0.6795992982245265;
%% Pole plot of LPF
N = 5;
Omega_0 = 0.5007;
B = 0.31067;
%% EQUIVALENT LOWPASS FILTER SPECIFICATIONS
D2 = 43.44;
D1 = 0.384;
Omega_ls = 1.2069;
Omega_lp = 1;
epsilon = sqrt(D1);
N = ceil((acosh(sqrt(D2/D1))) / (acosh(Omega_ls /
    Omega_lp)));
fprintf('N = %d\n', N);
%% LHP POLES
a = 0:2 * N - 1;
A = (2 * a + 1) .* (pi /(2 * N));
B = (1 / N) * asinh(1 / epsilon);
poles = (-1 * sin(A) * sinh(B)) + 1i * (cos(A) * cosh(B
    ));
figure();
plot(poles, '.', 'MarkerSize', 20);
daspect([1 1 1]);
hold on;
x = linspace(-pi, pi, 10000);
a = sinh(B) .* cos(x);
b = cosh(B) .* sin(x);
```

```
plot(a,b);
hold on;
plot(0,0,'g*');
LHP = [];
for i=1:size(poles,2)
    hold on;
    plot([0, real(poles(1,i))], [0, imag(poles(1,i))],
        'r-');
    if real(poles(1,i)) < 0
        disp(poles(1,i));
        LHP = [LHP, poles(1,i)];
    end
end
plot([0, 0],[-1.5, 1.5], 'k-');
plot([-1.5, 1.5],[0, 0], 'k-');
title('Poles of $H_{LPF}(s_L) \, H_{LPF}(-s_L)$ in the
    $s_L$ plane', 'Interpreter', 'latex');
xlabel('\Sigma_k');
ylabel('\Omega_k');
%% ANALOG LOWPASS TRansfer function
syms s;
den=1;
num =1;
for i=1:size(LHP, 2)
    num = num * LHP(1,i);
    den = den * (s - LHP(1,i));
end
H_LPF = num/den;
num_f = num;
den_f = den;
den = sym2poly(den);
w = linspace(0,2.5,1000);
h = freqs(num,den,w);
figure();
plot(w,abs(h),'LineWidth',1);
hold on;
yline(0.85,['--','r']);
xline(1,['--','g']);
```

```matlab
xline(1.2069,['--','g']);
yline(1.,['--','r']);
yline(0.15,['--','r']);
ylim([0,1.2]);
title("Equivalent Chebyshew Low Pass Magnitude Response
    ");
xlabel("\Omega_L");
ylabel("|H(j\Omega_L|");
set(gca,'Fontsize',5,'XTick', [0, 1, 1.2069], '
    xticklabel', {'0', '\Omega_{Lp} = 1','\Omega_{Ls} =
    1.2069'});
set(gca, 'YTick', [0, 0.15, 0.85,1], 'yticklabel', {'0'
    , '\delta = 1','1-\delta = 0.85','1'});
figure();
plot(w,angle(h));
title("Equivalent Chebyshew Low Pass Phase Response");
xlabel("\Omega_L");
ylabel("\angle H(j\Omega_L");

%% ANALOG BANDPASS TRANSFORMATION

syms s1;
Omega_0 = 0.5007;
B = 0.31067;
s1 = (B * s)./(s^2 + Omega_0^2);
h = subs(den_f, s, s1);
H_BSF = subs(H_LPF,s,s1);
H = num_f/h;

%% MAGNITUDE AND PHASE PLOT
[num,den] = numden(H);
k = subs(num(1),s,1);
num_coeff = sym2poly(num/k);
den_coeff = sym2poly(den/k);
% disp
    ('********************************************')
    ;
% disp(num_coeff);
```

```
% disp(den_coeff);
disp('************************************************');
w = linspace(0,2,1000);
[g,w] = freqs(num_coeff,den_coeff,w);
%disp(g);
figure();
plot(w,abs(g),'LineWidth',1);
hold on;
yline(0.85,['--','r']);
xline(os1,['--','g']);
xline(os2,['--','g']);
xline(op1,['--','g']);
xline(op2,['--','g']);
yline(1.,['--','r']);
yline(0.15,['--','r']);
ylim([0,1.2]);
xlim([0,1])
title("Chebyshew Bandstop Filter Magnitude Response");
xlabel("\Omega");
ylabel("|H(j\Omega|");
set(gca, 'XTick', [op1, os1, os2, op2], 'xticklabel', {'\Omega_{p1}', '\Omega_{s1}', '\Omega_{s2}', '\Omega_{p2}'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
%% phase response
figure();
plot(w,angle(g),'LineWidth',1);
hold on;
title("Chebyshew Bandstop Filter Phase Response");
xlabel("\Omega");
ylabel('\angle H(j \Omega)');
%% Discrete domain transformation
syms z;
c = subs(H_BSF, s, (z-1) / (z+1));
disp(c);

```

```
[num,den] = numden(c);
num_coeff = sym2poly(num);
den_coeff = sym2poly(den);
% disp('*******************************************');
disp(num_coeff);
disp(den_coeff);
w = linspace(0,2,1024);
[g,w] = freqz(num_coeff,den_coeff,1024*1024,400e3);
figure();
plot(w,abs(g),'Linewidth',1);
hold on;
yline(0.85,['--','r']);
xline(45e3,['--','g']);
xline(48e3,['--','g']);
xline(73e3,['--','g']);
xline(76e3,['--','g']);
yline(1.,['--','r']);
yline(0.15,['--','r']);
ylim([0,1.2]);
xlim([0,140e3])
title("Discrete Time Bandstop Filter Magnitude Response");
xlabel("\Omega");
ylabel("|H(j\Omega|");
set(gca, 'XTick', [45e3, 48e3, 73e3, 76e3], 'xticklabel', {'f_{p1}', 'f_{s1}', 'f_{s2}', 'f_{p2}'});
set(gca, 'YTick', [0.15, 0.85, 1, 1.15], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1', '1 + \delta_1 = 1.15'});
figure();
plot(w,angle(g),'LineWidth',1);
hold on;
title("Discrete Time Bandstop Filter Phase Response");
xlabel("\Omega");
ylabel('\angle H(j \Omega)');
```

10.4 BS-FIR FILTER

```
clc; clear all; close all;
%% BS Filter Specs
omega_s1 = 0.240 * pi;
omega_s2 = 0.365 * pi;
omega_p2 = 0.380 * pi;
omega_p1 = 0.225* pi;
transition_bw = 0.0471;
delta = 0.15;
omega_c1 = (omega_p1 + omega_s1)/2;
omega_c2 = (omega_p2 + omega_s2)/2;
%% Kaiser window parameters
A = -20 * log10(delta);
min_width = ceil( 1 + ((A - 8)/(2.285 * transition_bw)));

alpha = -1;
if A < 21
    alpha = 0;
elseif A >=21 && A <= 50idea
    alpha = 0.5842 * (A - 21) ^ 0.4 + 0.07886 * (A - 21);
elseif A > 50
    alpha = 0.1102 * (A - 8.7);
else

end
beta = alpha /min_width;
%% Magnitude Response
M = min_width + 13;
w = kaiser(M,beta);
BSF_IDEAL = ideal_lpf(pi,M) - ideal_lpf(omega_c2,M) + ideal_lpf(omega_c1,M);
BSF_FIR = BSF_IDEAL .* w';
[H,f] = freqz(BSF_FIR,1,1024, 400e3);
figure();
plot(f, abs(H));
```

```
hold on;
xline(48e3, 'm--', 'LineWidth', 1.5);
hold on;
xline(73e3, 'm--', 'LineWidth', 1.5);
hold on;
xline(45e3, 'm--', 'LineWidth', 1.5);
hold on;
xline(76e3, 'm--', 'LineWidth', 1.5);
hold on;
yline(1.15, 'r--', 'LineWidth', 1.5);
hold on;
yline(0.85, 'r--', 'LineWidth', 1.5);
hold on;
yline(0.15, 'r--', 'LineWidth', 1.5);
xlabel('f in 10^4 Hz');
ylabel('|H(e^{j2 \pi f})');
title('Magnitude Response of Discrete Time FIR Bandstop Filter');
set(gca, 'XTick', [45e3, 48e3, 73e3, 76e3], 'xticklabel', {'f_{p1}', 'f_{s1}', 'f_{s2}', 'f_{p2}'});
set(gca, 'YTick', [0.15, 0.85, 1, 1.15], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1', '1 + \delta_1 = 1.15'});
%% phase and impulse response
disp(BSF_FIR);
fvtool(BSF_FIR, 'Analysis', 'Phase');
fvtool(BSF_FIR, 'Analysis', 'Impulse');
%% ideal lpf code
function lpass = ideal_lpf(wc,M);

alpha = (M-1)/2;
n = [0:1:(M-1)];
m = n - alpha + eps;
lpass = sin(wc*m) ./ (pi*m);
end
```

10.5 BP-ELLIPTICAL FILTER

```
% Please execute section wise
 clc; clear ; close all;
%% elliptical filter parameter specifications
   calculations
ws =1.1226;
wp=1;
Gp = 0.85;
D1 = 0.384;
D2 = 43.44;
Ap = 1.4116;
As = 16.478;
ep = sqrt(10^(Ap/10) - 1); % ripple factors
es = sqrt(10^(As/10) - 1); % ripple factors
k = wp/ws;
k1 = ep/es;
kf = sqrt(1-k^2);
t = (k1)^2;
kf1 = sqrt(1-t);
%% JACOBIAN INTEGRAL CALCULATION
syms x;
f = 1/sqrt(1-k^2*sin(x)^2);
g = 1/sqrt(1-kf^2*sin(x)^2);
h = 1/sqrt(1-k1^2*sin(x)^2);
i = 1/sqrt(1-kf1^2*sin(x)^2);
K = double(int(f,0,pi/2));
Kf = double(int(g,0,pi/2));
K1 = double(int(h,0,pi/2));
Kf1 = double(int(i,0,pi/2));
disp(K);   %Jacobian integral output
disp(Kf);
disp(K1);
disp(Kf1);
N = ceil((Kf1*K)/(K1*Kf));  %filter order
disp(N);
%% updation of k value
u1 = 1/N;
```

```
u2= Kf1/4;
m = (Kf1*3)/4;
%kj = (((kf1)^N)*(sne(u1,kf1)^4)*(sne(u2,kf1)^4)); % ellipdeg uses this eqn
kj = ellipdeg(N,k1);
%% pole zero calculation for the above specifications
L = floor(N/2); r = mod(N,2); % L is the number of second-order sections
i = (1:L)'; u = (2*i-1)/N; zeta_i = cde(u,kj);
z = wp*1j./(kj*zeta_i); % zeros of elliptic rational function
ZEROES = z;
v0 = -1j*asne(1j/ep, k1)/N;
p = wp*1j*cde((u-1j*v0), kj); % filter poles
poles = p;
p0 = wp*1j*sne(1j*v0, kj); % first-order pole, needed when N is odd
B = [ones(L,1), -2*real(1./z), abs(1./z).^2]; % second-order numerator coefficients
A = [ones(L,1), -2*real(1./p), abs(1./p).^2]; % second-order denominator coefficients
if r==0 % prepend first-order sections
    B = [Gp, 0, 0; B]; A = [1, 0, 0; A];
else
    B = [1, 0, 0; B]; A = [1, -real(1/p0), 0; A];
end
ZEROES = cplxpair([ZEROES; conj(ZEROES)]); % append conjugate zeros
p = cplxpair([p; conj(p)]); % append conjugate poles
if r==1, p = [p; p0]; end % append first-order pole when N is odd
H0 = Gp^(1-r); % dc gain
%% Elliptic Lowpass filter transfer function and plots
syms sl Omega_L;
H_LPF = ((1 + 0.8835 * sl^2)*(1 + 0.3196 * sl^2)*0.85) / ((1 + 0.0619 * sl + sl^2) * (1 + 1.1314 * sl + 1.5998 * sl^2));
H_LPF_freq = subs(H_LPF, sl, 1i * Omega_L);
```

```
[ns, ds] = numden(H_LPF);
nsl = sym2poly(ns);
dsl = sym2poly(ds);
kn = ds(1);
nsl = nsl / kn;
ds = ds / kn;
disp(nsl);
disp(dsl);
figure();
fplot(abs(H_LPF_freq),'LineWidth',1);
hold on;
fplot(sl - sl - 0.15 + 1, 'g--', 'Markersize', 10);
hold on;
fplot(sl-sl+1, 'g--', 'Markersize', 10);
hold on;
fplot(sl - sl + 0.15, 'g--', 'Markersize', 10);
xline(1, 'm--','LineWidth',1);
hold on;
xline(1.1226,'m--','LineWidth',1);
axis([0 2 0 1.2]);
set(gca,'FontSize',5, 'XTick', [0, 1, 1.1226], 'xticklabel', {'0', '\Omega_{Lp} = 1','\Omega_{Ls} = 1.1226'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
daspect([1 1 1]);
title('Equivalent Elliptic Lowpass filter mangitude response');
xlabel('\Omega_L');
ylabel('|H(j \Omega_L)|');
figure();
fplot(angle(H_LPF_freq));
xlabel('\Omega_L');
ylabel('\angle H(j \Omega_L)');
title('Equivalent Butterworth Lowpass filter phase response');

%% Analog lowpass to bandpass frequency transformation
```

```matlab
 syms s Omega;
 Omega_0 = 0.592;
 B1 = 0.362;
 H_BPF = subs(H_LPF, sl, (s^2 + Omega_0^2) / (B1 * s));
[ns, ds] = numden(H_BPF);
ns = sym2poly(ns);
ds = sym2poly(ds);
kn = ds(1);
ns = ns ./ kn;
ds = ds ./ kn;
disp(ns);
disp(ds);
H_BPF_freq = subs(H_BPF, s, 1i * Omega);
figure();
fplot(abs(H_BPF_freq),'LineWidth',1);
set(gca, 'XTick', [0.4175, 0.4381, 0.8001, 0.8291], 'xticklabel', {'\Omega_{s1}', '\Omega_{p1}', '\Omega_{p2}', '\Omega_{s2}'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
hold on;
xline(0.4175, 'r--');hold on;
xline(0.4381, 'r--');hold on;
xline(0.8001, 'r--');hold on;
xline( 0.8291, 'r--');
hold on;
fplot(s - s - 0.15 + 1, 'g--', 'Markersize', 10);
hold on;
fplot(s-s+1, 'g--', 'Markersize', 10);
hold on;
fplot(s - s + 0.15, 'g--', 'Markersize', 10);
daspect([1 1 1]);
title('Elliptic Bandpass filter mangitude response');
xlabel('\Omega');
ylabel('|H(j \Omega)|');
axis([0 1.2 0 1.2]);
%
```

```matlab
%% Analog to z bilinear transformation
syms z;
Hz = subs(H_BPF, s, (z-1)/(z+1));
[Nz, Dz] = numden(Hz);
Nz = sym2poly(Nz);
Dz = sym2poly(Dz);
kn = Dz(1);
Nz = Nz / kn;
Dz = Dz / kn;
disp(Nz);
disp(Dz);
[H,f] = freqz(Nz,Dz,1024*1024, 540e3);
figure();
plot(f,abs(H),'LineWidth',1);
axis([0 170e3 0 1.3]);
hold on;
yline(1.00, 'g--', 'LineWidth', 1);
hold on;
yline(0.85, 'g--', 'LineWidth', 1);
hold on;
yline(0.15, 'g--', 'LineWidth', 1);
hold on;
xline(71e3, 'r--', 'LineWidth', 1);
hold on;
xline(116e3, 'r--', 'LineWidth', 1);
hold on;
xline(69e3, 'r--', 'LineWidth', 1);
hold on;
xline(119e3, 'r--', 'LineWidth', 1);
xlabel('f in 10^4 Hz');
ylabel('|H(e^{j 2\pi f})|');
title('Magnitude Response of the Discrete Time Bandpass Filter');
set(gca, 'XTick', [68e3, 71e3, 116e3, 119e3], 'xticklabel', {'f_{s1}', 'f_{p1}', 'f_{p2}', 'f_{s2}'});
set(gca, 'YTick', [0.15, 0.85, 1, 1.15], 'yticklabel',
```

```
{'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1', '1 + \delta_1 = 1.15'});
fvtool(Nz, Dz, 'Analysis', 'Phase');
```

10.6 BS-ELLIPTICAL FILTER

```
% Please execute section wise
clc; clear ; close all;
%% elliptical filter parameter specifications calculations
ws =1.2069;
wp=1;
Gp = 0.85;
D1 = 0.384;
D2 = 43.44;
Ap = 1.4116;
As = 16.478;
ep = sqrt(10^(Ap/10) - 1); % ripple factors
es = sqrt(10^(As/10) - 1);
k = wp/ws;
k1 = ep/es;
kf = sqrt(1-k^2);
t = (k1)^2;
kf1 = sqrt(1-t);
%% JACOBIAN INTEGRAL CALCULATION
syms x;
f = 1/sqrt(1-k^2*sin(x)^2);
g = 1/sqrt(1-kf^2*sin(x)^2);
h = 1/sqrt(1-k1^2*sin(x)^2);
i = 1/sqrt(1-kf1^2*sin(x)^2);
K = double(int(f,0,pi/2));
Kf = double(int(g,0,pi/2));
K1 = double(int(h,0,pi/2));
Kf1 = double(int(i,0,pi/2));
disp(K);   %Jacobian integral output
disp(Kf);
disp(K1);
disp(Kf1);
N = ceil((Kf1*K)/(K1*Kf));  %filter order
disp(N);
%% updation of k value
u1 = 1/N;
```

```
u2= Kf1/4;
m = (Kf1*3)/4;
%kj = (((kf1)^N)*(sne(u1,kf1)^4)*(sne(u2,kf1)^4));
kj = ellipdeg(N,k1);
%% pole zero calculation for the above specifications
L = floor(N/2); r = mod(N,2); % L is the number of second-order sections
i = (1:L)'; u = (2*i-1)/N; zeta_i = cde(u,kj);
z = wp*1j./(kj*zeta_i); % zeros of elliptic rational function
ZEROES = z;
v0 = -1j*asne(1j/ep, k1)/N;
p = wp*1j*cde((u-1j*v0), kj); % filter poles
poles = p;
p0 = wp*1j*sne(1j*v0, kj); % first-order pole, needed when N is odd
B = [ones(L,1), -2*real(1./z), abs(1./z).^2]; % second-order numerator coefficients
A = [ones(L,1), -2*real(1./p), abs(1./p).^2]; % second-order denominator coefficients
if r==0 % prepend first-order sections
    B = [Gp, 0, 0; B]; A = [1, 0, 0; A];
else
    B = [1, 0, 0; B]; A = [1, -real(1/p0), 0; A];
end
ZEROES = cplxpair([ZEROES; conj(ZEROES)]); % append conjugate zeros
p = cplxpair([p; conj(p)]); % append conjugate poles
if r==1, p = [p; p0]; end % append first-order pole when N is odd
H0 = Gp^(1-r); % dc gain
%% Elliptic Lowpass filter transfer function and plots
syms sl Omega_L;
H_LPF = (1 + 0.6295 * sl^2)/ ((1 + 1.6047 * sl) * (1 + 0.2305 * sl + 0.9994 * sl^2));
H_LPF_freq = subs(H_LPF, sl, 1i * Omega_L);
[ns, ds] = numden(H_LPF);
nsl = sym2poly(ns);
```

```
dsl = sym2poly(ds);
k = ds(1);
nsl = nsl / k;
ds = ds / k;
disp(nsl);
disp(dsl);
figure();
fplot(abs(H_LPF_freq),'LineWidth',1);
hold on;
fplot(sl - sl - 0.15 + 1, 'g--', 'Markersize', 10);
hold on;
fplot(sl-sl+1, 'g--', 'Markersize', 10);
hold on;
fplot(sl - sl + 0.15, 'g--', 'Markersize', 10);
xline(1, 'magenta--','LineWidth',1);
hold on;
xline(1.2069,'magenta--','LineWidth',1);
axis([0 2 0 1.2]);
set(gca,'FontSize',5, 'XTick', [0, 1, 1.2069], 'xticklabel', {'0', '\Omega_{Lp} = 1','\Omega_{Ls} = 1.2069'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
daspect([1 1 1]);
title('Equivalent Elliptic Lowpass filter mangitude response');
xlabel('\Omega_L');
ylabel('|H(j \Omega_L)|');
figure();
fplot(angle(H_LPF_freq));
xlabel('\Omega_L');
ylabel('\angle H(j \Omega_L)');
title('Equivalent elliptic Lowpass filter phase response');

%% Analog lowpass to bandpass frequency transformation
syms s Omega;
Omega_0 = 0.5007;
```

```
 B1 = 0.31067;
 H_BSF = subs(H_LPF, sl,(B1 * s)/(s^2 + Omega_0^2));
[ns, ds] = numden(H_BSF);
ns = sym2poly(ns);
ds = sym2poly(ds);
k = ds(1);
ns = ns ./ k;
ds = ds ./ k;
disp(ns);
disp(ds);
sys = tf(ns,ds);   % Analog BSF transfer function
H_BSF_freq = subs(H_BSF, s, 1i * Omega);
figure();
fplot(abs(H_BSF_freq),'LineWidth',1);
set(gca, 'XTick', [0.3689, 0.3959, 0.6457, 0.6795], 'xticklabel', {'\Omega_{p1}', '\Omega_{s1}', '\Omega_{s2}', '\Omega_{p2}'});
set(gca, 'YTick', [0.15, 0.85, 1], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1'});
hold on;
xline(0.3689, 'r--');hold on;
xline(0.3959, 'r--');hold on;
xline(0.6457, 'r--');hold on;
xline( 0.6795, 'r--');
hold on;
fplot(s - s - 0.15 + 1, 'g--', 'Markersize', 10);
hold on;
fplot(s-s+1, 'g--', 'Markersize', 10);
hold on;
fplot(s - s + 0.15, 'g--', 'Markersize', 10);
daspect([1 1 1]);
title('Elliptic Bandstop filter mangitude response');
xlabel('\Omega');
ylabel('|H(j \Omega)|');
axis([0 1.5 0 1.2]);
%----------------------------------------------------------------------
```

```
%% Analog to z bilinear transformation
syms z;
Hz = subs(H_BSF, s, (z-1)/(z+1));
[Nz, Dz] = numden(Hz);
Nz = sym2poly(Nz);
Dz = sym2poly(Dz);
k = Dz(1);
Nz = Nz / k;
Dz = Dz / k;
disp(Nz);
disp(Dz);
[H,f] = freqz(Nz,Dz,1024*1024, 400e3);
SYS = tf(Nz,Dz);
figure();
plot(f,abs(H),'LineWidth',1);
axis([20e3 100e3 0 1.3]);
hold on;
yline(1.00, 'g--', 'LineWidth', 1);
hold on;
yline(0.85, 'g--', 'LineWidth', 1);
hold on;
yline(0.15, 'g--', 'LineWidth', 1);
hold on;
xline(45e3, 'r--', 'LineWidth', 1);
hold on;
xline(48e3, 'r--', 'LineWidth', 1);
hold on;
xline(73e3, 'r--', 'LineWidth', 1);
hold on;
xline(76e3, 'r--', 'LineWidth', 1);
xlabel('f in 10^4 Hz');
ylabel('|H(e^{j 2\pi f})|');
title('Magnitude Response of the Discrete Time Bandstop Filter');
set(gca, 'XTick', [45e3, 48e3, 73e3, 76e3], 'xticklabel', {'f_{p1}', 'f_{s1}', 'f_{s2}', 'f_{p2}'});
set(gca, 'YTick', [0.15, 0.85, 1, 1.15], 'yticklabel', {'\delta_2 = 0.15', '1 - \delta_1 = 0.85', '1', '1 +
```

```
    \delta_1 = 1.15'});
fvtool(Nz, Dz, 'Analysis', 'Phase');
```