Filter Design

K Rahul
EE23BTECH11027

1 Introduction

We are supposed to design the equivalent FIR and IIR filter realizations for given filter number. This is a bandpass filter whose specifications are available below.

2 Why are Ideal Filters not realizable in real life?

Let us take the impulse response of an ideal lowpass filter with the following frequenct response:

$$H(f) = \begin{cases} 1, & \text{if } |f| < f_c \\ 0, & \text{elsewhere} \end{cases} \tag{1}$$

Its impulse response h(n) is given by:

$$h(n) = \frac{\sin 2\pi f_c n}{\pi n} \tag{2}$$

Hence, as is very clear, the filter is not a causal filter(as $h(n) \neq 0 \ \ \forall n < 0$). Hence, it cannot exist in real life.

3 Explanation of terms about to be used:

These terms are common to all non-ideal filters, so a simple picture of a non-ideal filters with explanation is given below

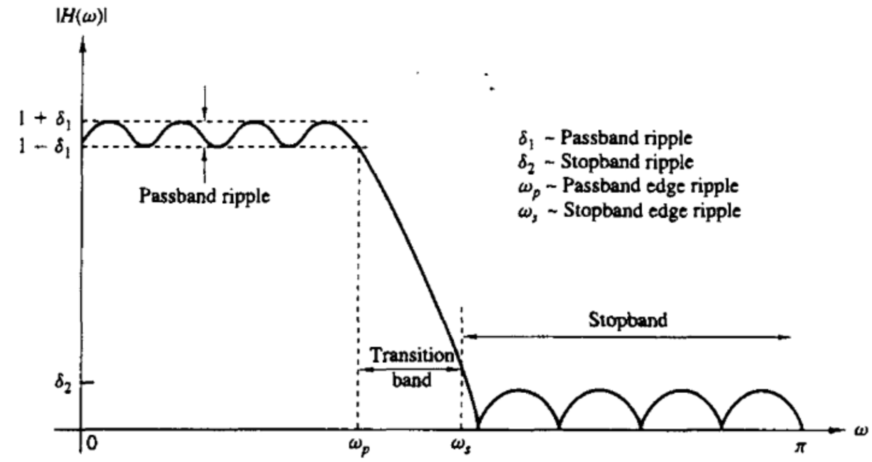

Figure 1: Simple non-ideal filter terms
Digital Signal Processing, Principles, Algorithms and Applications by John G. Proakis and Dimitris G. Manolakis, third edition.

4 Filter Specifications

4.1 The Digital Filter

1. Passband: The passband is from {4 + 0.6(j)}kHz to {4 + 0.6(j+2)}kHz.
   where

$$j = (r - 11000) \mod \sigma \tag{3}$$

   where σ is sum of digits of roll number and r is roll number.

$$r = 11027 \tag{4}$$
$$\sigma = 11 \tag{5}$$
$$j = 5 \tag{6}$$

   substituting $j = 5$ gives the passband range for our bandpass filter as 7 kHz - 8.2 kHz. Hence, the un-normalized discrete time filter passband frequencies are $F_{p1} = 7$ kHz and $F_{p2} = 8.2$ kHz.
   The corresponding normalized digital filter passband frequencies are

$$\omega_{p1} = 2\pi \frac{F_{p1}}{F_s} = 0.2916\pi \tag{7}$$
$$\omega_{p2} = 2\pi \frac{F_{p2}}{F_s} = 0.3417\pi \tag{8}$$

2. Tolerances: The passband (δ_1) and stopband (δ_2) tolerances are given to be equal, so we let $\delta_1 = \delta_2 = \delta = 0.15$.
3. Stopband: The transition band for bandpass filters is $\Delta F = 0.3$ kHz on either side of the passband.

$$F_{s1} = 7 - 0.3 = 6.7 \text{ KHz} \tag{9}$$
$$F_{s2} = 8.2 + 0.3 = 8.5 \text{ KHz} \tag{10}$$

$$\omega_{s1} = 2\pi\frac{F_{s1}}{F_s} = 0.2979\pi \tag{11}$$
$$\omega_{s2} = 2\pi\frac{F_{s2}}{F_s} = 0.3541\pi \tag{12}$$
$$\tag{13}$$

4.2 The Analog filter

Since an analog filter equivalent is required, we need to map from the z-plane to the s-plane. A mapping from s-plane to z-plane is called the bilinaer transform. Using $z = re^{j\omega}$ and $s = \sigma + j\Omega$ and using bilinear

$$s = \frac{z-1}{z+1} \tag{14}$$
$$\sigma + j\Omega = \frac{re^{j\omega} - 1}{re^{j\omega} + 1} \tag{15}$$
$$= \frac{r^2 - 1}{1 + r^2 + 2r\cos\omega} + j\frac{2r\sin\omega}{1 + r^2 + 2r\cos\omega} \tag{16}$$

When $r = 1$, we get

$$\Omega = \tan\frac{\omega}{2} \tag{17}$$

Using this relation, we obtain the analog passband and stopband frequencies as: $\Omega_{p1} = 0.4930$, $\Omega_{p2} = 0.5950$ and $\Omega_{s1} = 0.4690$, $\Omega_{s2} = 0.6217$.

5 The IIR Filter Design

We are supposed to design filters whose stopband is monotonic and passband equiripple. Hence, we use the Chebyschev approximation to design our bandpass IIR filter.

5.1 The Chebyshev Filter and Polynomials

Since an ideal filter is not possible as shown before, a good approximation can be sought after such that the ripple is as less as possible. The transfer function of such a minimally flat approximation is of the form

$$|H(j\omega)|^2 = \frac{1}{1 + |K(j\omega)|^2} \tag{18}$$

Where $K(j\omega)$ is a polynomial function of $j\omega$. Transfer function value must be between 1 and $\frac{1}{1+\epsilon^2}$, where ϵ is some small number so that ripple is minimum.

$$\implies -\epsilon \le |K(j\omega)| \le \epsilon \implies -1 \le \epsilon^{-1}|K(j\omega)| \le 1 \tag{19}$$

If $y = \epsilon^{-1}|K(j\omega)|$, then it is an oscillatory function lying between -1 and +1 always. Since we can choose any function, for ease, we use a sinusoidal function. Now bear in mind that ω is the normalized frequency, hence $|\omega| \le 1$

$$y = cos(ncos^{-1}(x)) = c_N(\omega)\ \forall |\omega| < 1 \tag{20}$$

Thus,

$$\epsilon^{-1}|K(j\omega)| = cN(\omega) \implies |H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 cN^2(\omega)} \tag{21}$$

Where ω is the normalized wave frequency. Using the expansion of $cos(n\theta)$ in terms of $cos(\theta)$, and putting $\theta = cos^{-1}(\omega)$, we get

$$c_N(\omega) = 2^{N-1}\omega^N - \frac{N}{1!}2^{N-3}\omega^{N-2} + \frac{N(N-3)}{2!}2^{N-5}\omega^{N-4} - \frac{N(N-3)(N-5)}{3!}2^{N-7}\omega^{N-6} + \dots \tag{22}$$

It can be obtained using a recursive algorithm too

$$c_N(x) = 2c_{N-1}(x) - c_{N-2}(x) \tag{23}$$

5.2 The Analog Filter

1. Low Pass Filter Specifications: Let $H_{a,BP}(j\Omega)$ be the desired analog bandpass filter, with the specifications provided in Section 2.2, and $H_{a,LP}(j\Omega_L)$ be the equivalent low pass filter, then

$$\Omega_L = \frac{\Omega^2 - \Omega_0^2}{B\Omega} \tag{24}$$

where $\Omega_0 = \sqrt{\Omega_{p1}\Omega_{p2}} = 0.5416$ and $B = \Omega_{p2} - \Omega_{p1} = 0.1020$.

Substituting Ω_{s1} and Ω_{s2} in (24) we obtain the stopband edges of lowpass filter

$$\Omega_{Ls1} = \frac{\Omega_{s1}^2 - \Omega_0^2}{B\Omega_{s1}} = -1.5337 \tag{25}$$

$$\Omega_{Ls2} = \frac{\Omega_{s2}^2 - \Omega_0^2}{B\Omega_{s2}} = 1.4694 \tag{26}$$

And we choose the minimum of these two stopband edges

$$\Omega_{Ls} = \min(|\Omega_{Ls_1}|, |\Omega_{Ls_2}|) = 1.4694. \tag{27}$$

2. The Low Pass Chebyschev Filter Paramters: The magnitude of frequency response of the low pass filter is given by

$$|H_{a,LP}(j\Omega_L)|^2 = \frac{1}{1+\epsilon^2 c_N^2(\Omega_L/\Omega_{Lp})} \tag{28}$$

The passband edge of the low pass filter is chosen as $\Omega_{Lp} = 1$. Therfore ,

$$|H_{a,LP}(j\Omega_L)|^2 = \frac{1}{1+\epsilon^2 c_N^2(\Omega_L)} \tag{29}$$

Imposing the band restrictions on (28)

$$|H_{a,LP}(j\Omega_L)|^2 < \delta_2 \text{ for } \Omega_L = \Omega_{Ls} \tag{30}$$

$$1-\delta_1 < |H_{a,LP}(j\Omega_L)|^2 < 1 \text{ for } \Omega_L = \Omega_{Lp} \tag{31}$$

$$\tag{32}$$

we obtain :

$$\frac{\sqrt{D_2}}{c_N(\Omega_{Ls})} \leq \epsilon \leq \sqrt{D_1},$$

$$N \geq \left\lceil \frac{\cosh^{-1}\sqrt{D_2/D_1}}{\cosh^{-1}\Omega_{Ls}} \right\rceil, \tag{33}$$

where $D_1 = \frac{1}{(1-\delta)^2} - 1$ and $D_2 = \frac{1}{\delta^2} - 1$ and $\lceil . \rceil$ is known as the ceiling operator .
we get $N \geq 4$ and $0.3084 \leq \epsilon \leq 0.6197$
The below code plots (28) for different values of ϵ .

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/plot1.py

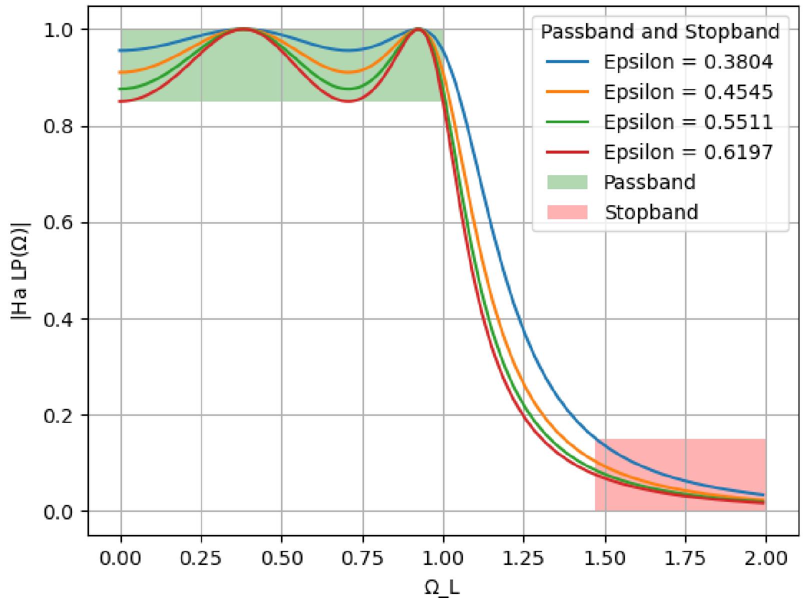

Figure 2: The Analog Low-Pass Frequency Response for $0.3084 \leq \epsilon \leq 0.6197$

In Fig. 2 we can observe the equiripple behaviour in passband and monotonic behaviour in stopband. As the value of ϵ increases the value of $|H_{a,LP}(j\Omega_L)|$ decreases.

3. The Low Pass Chebyschev Filter: The next step in design is to find an expression for magnitude response in s domain.

   Using $s = j\Omega$ or in this case $s_L = j\Omega_L$ we obtain:

$$|H_{a,LP}(j\Omega_L)|^2 = \frac{1}{1 + \epsilon^2 c_N^2(\frac{s_L}{j})} \tag{34}$$

   To find poles equate the denominator to zero:

$$1 + \epsilon^2 c_N^2 \left(\frac{s_L}{j}\right) = 0 \quad \text{where} \quad c_N(x) = cos\left(Ncos^{-1}(x)\right) \tag{35}$$

   On solving (35) we obtain poles :

$$s_k = -\Omega_{Lp} \sin(A_k) \sinh(B_k) - j\Omega_{Lp} \cos(A_k) \cosh(B_k) \tag{36}$$

where k is the index of the pole and

$$A_k = (2k+1)\frac{\pi}{2N} \tag{37}$$

$$B_k = \frac{1}{N}\sinh^{-1}\left(\frac{1}{\epsilon}\right) \tag{38}$$

The below code computes the values of s_k and stores it in a text file.

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/sk_gen.c

The poles obtained are formulated in the table below.

Pole	*Value*
s_1	-0.1621 + j1.0033
s_2	-0.3913 - j0.4156
s_3	-0.3913 - j0.4156
s_4	-0.1621 - j1.0033
s_5	-0.1621 + j1.0033
s_6	-0.3913 + j0.4156
s_7	-0.3916 - j0.4156
s_8	-0.1621 - j1.0033

Table 1: Values of s_k

The below code plots the pole-zero plot.

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/pole_zero_plot.py

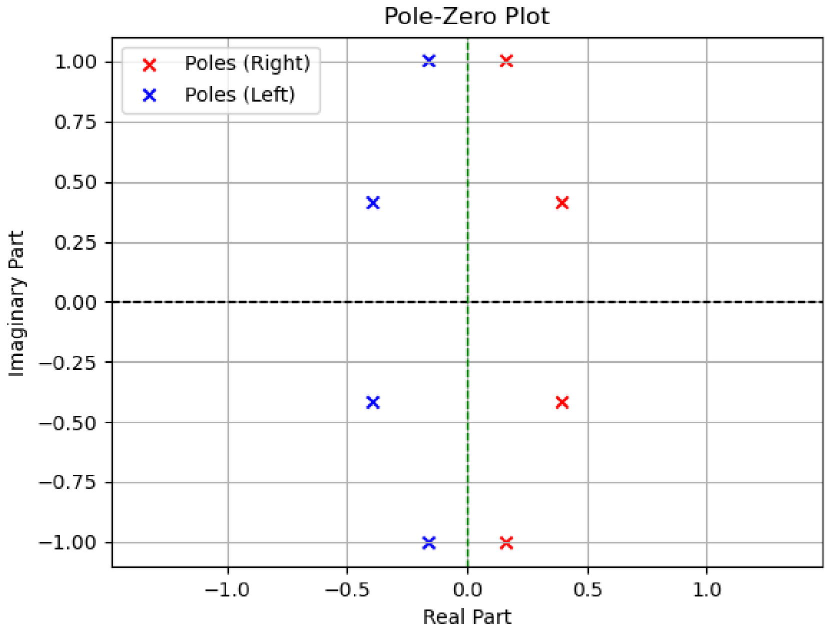

Figure 3: The Pole zero plot and all the poles lie on an ellipse. The left and right poles have been identified as shown.

The poles in the left half of the plane are considered in the design as we intend to design a stable system.
Therefore the magnitude response is written as :-

$$H_{a,LP}(s_L) = \frac{G_{LP}}{(s_L - s_5)(s_L - s_6)(s_L - s_7)(s_L - s_8)} \tag{39}$$

where G_{LP} is the gain of the Low pass filter. Refer to Table 1 for s_k values.

We know that from (28):-

$$\left|H_{a,LP}(s_L)\right| = \frac{1}{\sqrt{1+\epsilon^2}} \text{at } \Omega_L = 1 \implies s_L = j \tag{40}$$

The value of G_{LP} can be found using the following code after appropriate substitution

```
https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/
    gain_LP.py
```

Substituting respective values in (40) we get $G_{LP} = 0.3126$. Simplifications for equation (42) and (50) are done using the following code https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/HaLP_denom.py

$$H_{a,LP}(s_L) = \frac{0.3126}{(s_L - s_5)(s_L - s_6)(s_L - s_7)(s_L - s_8)} \tag{41}$$

$$= \frac{0.3126}{s_L^4 + 1.071s_L^3 + 1.6127s_L^2 + 0.9143s_L + 0.3367} \tag{42}$$

The figure to compare design and specification is produced using the following code

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/plot2.py

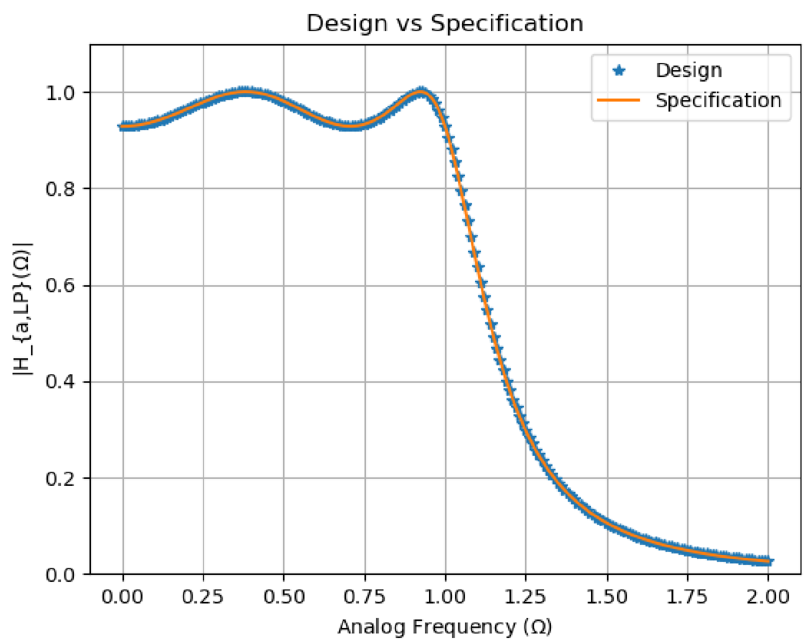

Figure 4: Design vs Specification corresponding to (42) and (29)

4. The Band Pass Chebyschev Filter: After verifying design with the required specifications the next step in design is to jump to required type of filter using fre-

quency transformation.

$$s_L = \frac{s^2 + \Omega_0^2}{Bs} \tag{43}$$

$$H_{a,BP}(s) = G_{BP} H_{a,LP}(s_L)\Big|_{s_L = \frac{s^2+\Omega_0^2}{Bs}}, \tag{44}$$

As there is one to one correspondence between the filters so $\Omega = \Omega_{p1}$ should correspond to Ω_{Lp}

$$s = j\Omega_{p1} \tag{45}$$

$$s_L = \frac{(j\Omega_{p1})^2 + \Omega_0^2}{B(j\Omega_{p1})} \tag{46}$$

$$\left|H_{a,BP}(j\Omega_{p1})\right| = 1 \tag{47}$$

$$G_{BP} \left|H_{a,LP}(s_L)\right| = 1 \tag{48}$$

Substituting (46) in (48) we obtain Gain of required bass pass filter:

$$G_{BP} = 1.077 \tag{49}$$

Thus the response in s domain

$$H_{a,BP}(s) = \frac{3.644 \times 10^{-5} s^4}{s^8 + 0.113s^7 + 1.190s^6 + 0.100s^5 + 0.526s^4 + 0.029s^3 + 0.102s^2 + 0.002s + 0.007} \tag{50}$$

The expressions in the s-domain and gain factors are computed by writing the following Python code.

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/plot3.py

In Figure 5, we plot $|H_{a,BP}(j\Omega)|$ as a function of Ω for both positve as well as negative frequencies. We find that the passband and stopband frequencies in the figure match well with those obtained analytically through the bilinear transformation.

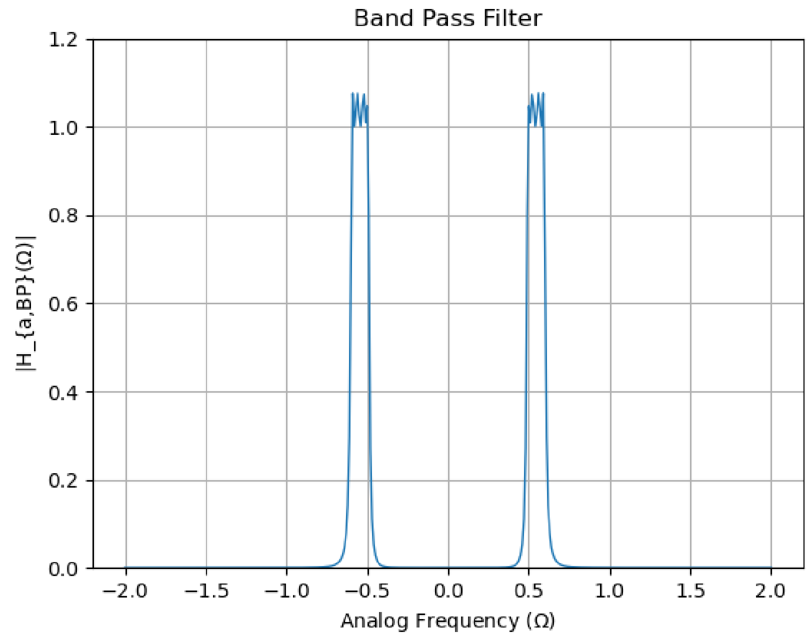

Figure 5: The Analog Bandpass Magnitude Response from (50).The filter design specifications are satisfied

5.3 The Digital Filter

From the bilinear transformation, we obtain the digital bandpass filter from the corresponding analog filter as

$$H_{d,BP}(z) = GH_{a,BP}(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}} \tag{51}$$

Substituting $s = \frac{1-z^{-1}}{1+z^{-1}}$ in (50) and calculating expression using the following python code we get :

$$H_{d,BP}(z) = \frac{G\left(1 - 4z^{-2} + 6z^{-4} - 4z^{-6} + z^{-8}\right)}{3.069 - 13.104z^{-1} + 32.612z^{-2} - 52.384z^{-3} + 60.726z^{-4} - 50.128z^{-5} + 29.908z^{-6} - 11.488^{-7} + 2.581z^{-8}} \tag{52}$$

where $G = 3.644 \times 10^{-5}$

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/filter_in_z.py

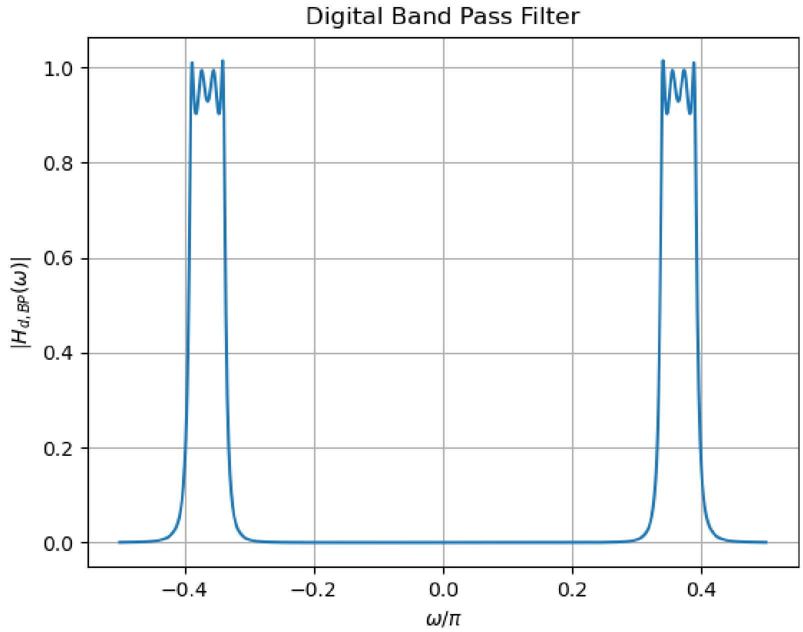

Figure 6: Digital Specifications are met. Passband and stopband frequencies are same

6 The FIR Filter

We design the FIR filter by first obtaining the (non-causal) lowpass equivalent using the Kaiser window and then converting it to a causal bandpass filter.

6.1 The Equivalent Lowpass Filter

The lowpass filter has a passband frequency ω_l and transition band $\Delta\omega = 2\pi\frac{\Delta F}{F_s} = 0.05\pi$. The stopband tolerance is $\delta = 0.15$.The cutoff-frequency is given by :

$$\omega_l = \frac{B}{2} \tag{53}$$

$$= 0.102\pi \tag{54}$$

The following python code is used to plot the frequency and the impulse response of an ideal Low Pass Filter.

```
https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/
FIR_ideal.py
```

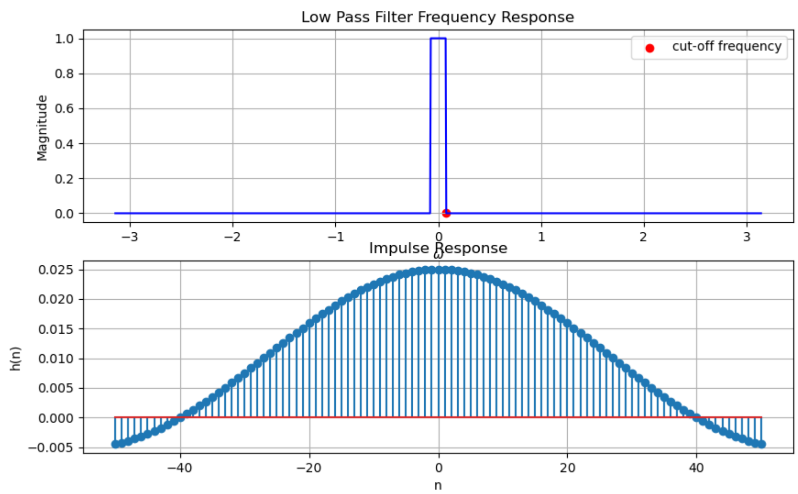

Figure 7: Frequency response and impulse response of an ideal Low Pass Filter

The impulse response of ideal Low Pass Filter is given by :

$$h(n) = \begin{cases} \frac{w_l}{\pi}, & \text{if } n = 0 \\ \frac{\sin(w_l n)}{n\pi}, & \text{if } n \neq 0 \end{cases} \tag{55}$$

From (55) we conclude that $h(n)$ for an ideal Low Pass Filter is not causal and can neither be made causal by introducing a finite delay. And $h(n)$ does not converge and hence the system is unstable.

6.2 The Kaiser Window

Therefore we move on windowing the impulse response.A window function is chosen and multiplied. The Kaiser window is defined as

$$w(n) = \begin{cases} \frac{I_0\left[\beta N\sqrt{1-\left(\frac{n}{N}\right)^2}\right]}{I_0(\beta N)}, & -N \leq n \leq N, \quad \beta > 0 \\ 0 & \text{otherwise,} \end{cases}$$

1. N is chosen according to

$$N \geq \frac{A-8}{4.57\Delta\omega}, \tag{56}$$

where $A = -20\log_{10}\delta$. Substituting the appropriate values from the design specifications, we obtain $A = 16.4782$ and $N \geq 48$.

2. β is chosen according to

$$\beta N = \begin{cases} 0.1102(A - 8.7) & A > 50 \\ 0.5849(A - 21)^{0.4} + 0.07886(A - 21) & 21 \le A \le 50 \\ 0 & A < 21 \end{cases} \quad (57)$$

The window function is defined as :

$$w(n) = \begin{cases} 1, & \text{for } -48 \le n \le 48 \\ 0, & \text{otherwise} \end{cases} \quad (58)$$

Therefore the desired impulse response is :

$$h_{lp} = h_n w_n \quad (59)$$

$$h(n) = \begin{cases} \frac{\sin(w_l n)}{n\pi}, & \text{for } -48 \le n \le 48 \\ 0 & \text{otherwise} \end{cases} \quad (60)$$

The code to plot the magnitude response of the FIR Low Pass Filter is as follows

https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/plot4.py

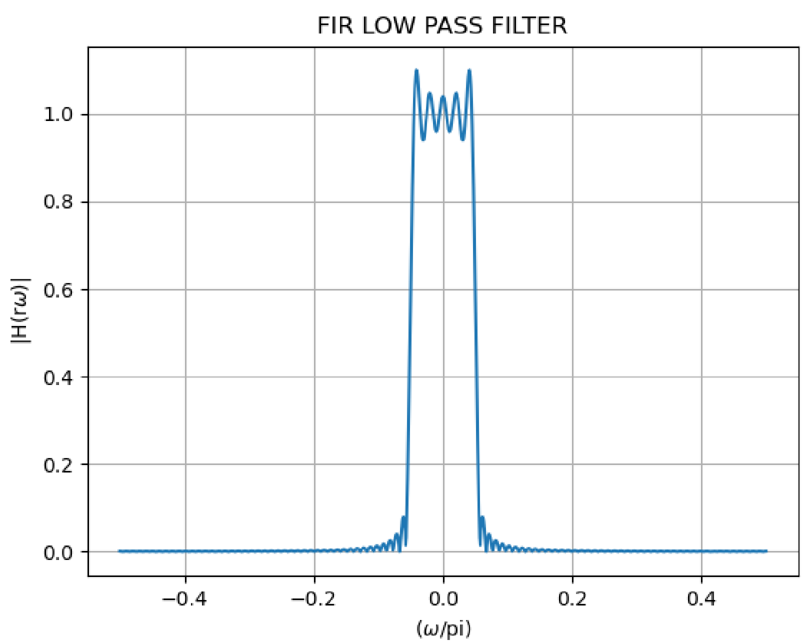

Figure 8: Magnitude Response of Low Pass Filter after using Kaiser Window

6.3 The Equivalent Band Pass Filter

A Band-Pass Filter (BPF) can be obtained by subtracting the magnitude response of a Low-Pass Filter (LPF) with cutoff frequency ω_{p1} from another LPF magnitude response with cutoff frequency ω_{p2}.

$$h_{BP}(n) = \begin{cases} \frac{\sin(w_{p2}n)}{n\pi} - \frac{\sin(\omega_{p1}n)}{n\pi}, & \text{for } n \neq 0 \\ \frac{\omega_{p2}-\omega_{p1}}{\pi} & \text{for} n = 0 \end{cases} \tag{61}$$

$$\frac{\sin(\omega_{p2}n)}{n\pi} - \frac{\sin\left(\omega_{p1}n\right)}{n\pi} = 2\cos\left(\frac{\omega_{p2}n + \omega_{p1}n}{2}\right)\sin\left(\frac{\omega_{p2}n - \omega_{p1}n}{2}\right) \tag{62}$$

$$= \frac{2\cos(0.3166n\pi)\sin(0.025n\pi)}{n\pi} \tag{63}$$

Multipying by window function we get :

$$h_{BP}(n) = \begin{cases} \frac{2\cos(0.3166n\pi)\sin(0.025n\pi)}{n\pi}, & \text{for } -48 \leq n \leq 48 \\ 0 & \text{otherwise} \end{cases} \tag{64}$$

The code to plot the same is as follows

```
https://github.com/HarryNyquist/EE1205/blob/main/Filter_Design/codes/plot5.py
```

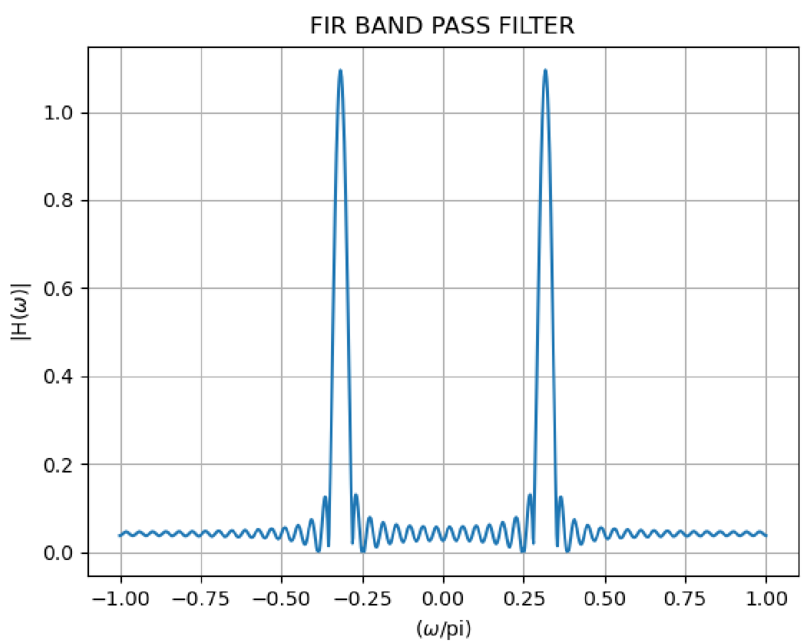

Figure 9: Magnitude Response of Band Pass Filter after using Kaiser Window