FILTER DESIGN

Design of a Bandpass FIR Filter

Niruhan Viswarupan. 140422H

This is submitted as a partial fulfillment for the module
EN2570: Digital Signal Processing
Department of Electronic and Telecommunication Engineering
University of Moratuwa

2nd of January, 2016

Design of a Bandpass Finite Impulse Response (FIR) Filter

Niruhan Viswarupan,
Undergraduate, Electronics and Telecommunication Department, University of Moratuwa
Katubedda, Sri Lanka
niruhanv@gmail.com

***Abstract* –** The purpose of this assignment is to design a non-recursive finite impulse response band pass filter in Matlab in using windowing method in conjunction with Kaiser Window. The parameters of the filter is to be selected according to the university admission number.

Keywords – *FIR Filter; bandpass filter; Matlab; Kaiser Window*

I. INTRODUCTION

Filter design is an important part of digital signal processing. Filtering is the removal of unwanted frequency components from a signal. The applications of filters is very diverse including audio processing, image processing, telecommunications etc. Finite impulse response filters are ones' where the length of the impulse response is finite. Here we discuss a non-recursive way of realizing a band pass filter. Band pass filters pass the frequency components within a certain range of frequencies and attenuate greatly the frequency components lower and higher than the range.

II. METHOD

The Matlab code was generated by following the steps below.
Index number 140422H

`Table 1: Filter Specifications

Passband ripple Ap	0.14 dB	Upper passband edge wp2	800 rad/s
Min. stopband attenuation Aa	52dB	Upper stopband edge wa2	950 rad/s
Lower stopband edge wa1	400 rad/s	Sampling frequency ws	2400 rad/s
Lower passband edge wp1	500 rad/s		

Step 01

Selecting the transition width

$$B_t = \min[(\omega_{p1} - \omega_{a1}),\ (\omega_{a2} - \omega_{p2})]$$

B_t = 100 rad/s
Selecting the cutoff points

$$\omega_{c1} = \omega_{p1} - \frac{B_t}{2}, \qquad \omega_{c2} = \omega_{p2} + \frac{B_t}{2}$$

w_{c1} = 450 rad/s w_{c2} = 850 rad/s

Step 02

Choose δ (represented in code as d) such that the actual passband ripple,A_p is equal to or less than specified passband ripple $\tilde{A}_p$, and the actual minimum stopband attenuation,A_a is equal or greater than the specified minimum stopband attenuation, $\tilde{A}_a$

A suitable value is

$$\delta = \min(\delta_p,\ \delta_a)$$

Where $\delta_p = \frac{10^{0.05A_p} - 1}{10^{0.05A_p} + 1}$ and $\delta_p = 10^{-0.05A_p}$

So, d = 0.0025

Step 03

With the required δ defined, the actual stopband attenuation A_a can be calculated as

$$A_a = -20\log(\delta)$$

So, A_a = 52

Step 04

Choose parameter α (represented in code as *alpha*) as

$$\alpha = \begin{cases} 0 & for\ A_a \leq 21 \\ 0.5842(A_a - 21)^{0.4} + 0.07886(A_a - 21) & for\ 21 < A_a \leq 50 \\ 0.07886(A_a - 21) & for\ A_a > 50 \end{cases}$$

So, *alpha* = 4.77166

Step 05

Choose parameter D as

$$D = \begin{cases} 0.9222 & for\ A_a \leq 21 \\ \dfrac{A_a - 7.95}{14.36} & forA_a - 21 \end{cases}$$

So, D = 3.0675

Then select the lowest odd value of N that would satisfy the inequality

N ≥ $\frac{\omega_s D}{B_t}$ + 1 where $B_t = \omega_a - \omega_p$

So, N = 75

Form $w_k(nT)$ using the following equations:

$$w_k(nT) = \begin{cases} \dfrac{I_0(\beta)}{I_0(\alpha)} & for\ |n| \leq (N-1)/2 \\ 0 & Otherwise \end{cases}$$

Where

$$\beta = \alpha\sqrt{1 - \left(\frac{2n}{N-1}\right)^2},\ I_0(\alpha) = 1 + \sum_{k=1}^{\infty}\left[\frac{1}{k!}\left(\frac{x}{2}\right)^k\right]^2$$

A Bessel function of first kind was written in Matlab as a function for the above purpose. The infinite sum was reduced as follows. It was seen that the summation terms converge to zero. So the terms smaller than 10^{-6} were neglected, thus making it a computable sum.

Step 07

Now we have $w_k(nT)$. We need *h(nT)*

$$h(nT) = \begin{cases} \dfrac{2}{\omega_s}(\omega_{c1} - \omega_{c2}) & for\ n = 0 \\ \dfrac{1}{n\pi}(\sin\omega_{c2}nT - \sin\omega_{c1}nT) & Otherwise \end{cases}$$

$$H'_w(z) = z^{-(N-1)/2}H_w(z)\ \ where\ H_w(z) = \mathcal{Z}[w_k(nT)\mathrm{h(nT)}]$$

$w_k(nT$ and *h(nT)* generated were element wise multipled to generate the filter response.

Step 08

Input signal given below was generated by addition of three sin waves.

$$x(nT) = \sum_{i=1}^{3} \sin(\omega_i nT)$$

To find the output fft's of the input and the filter response were multipled and ifft was taken.

III. RESULTS

The filter characteristic plots and input output analysis is as follows,

Figure 1.

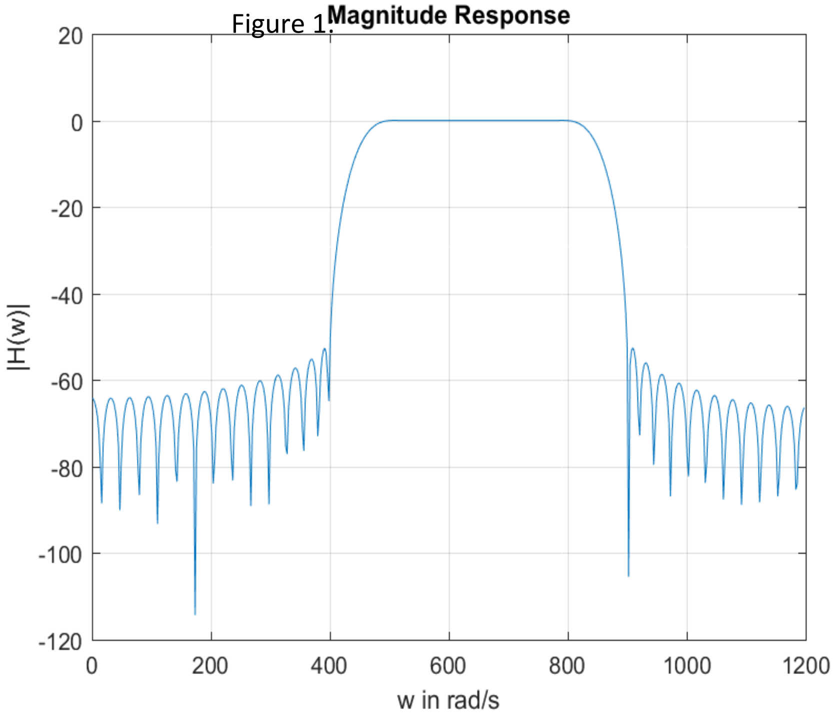


Figure 2.

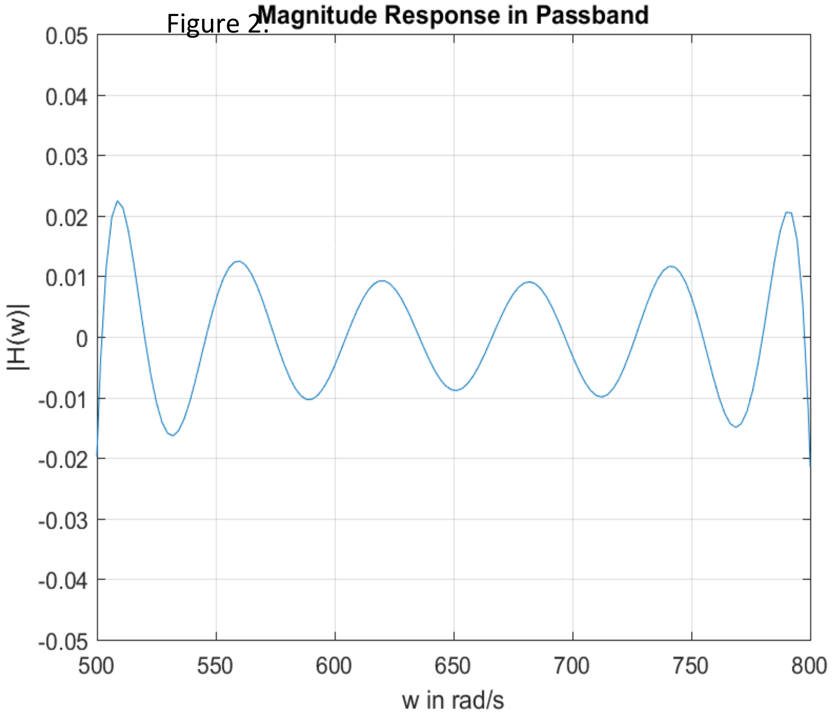


Figure 3:

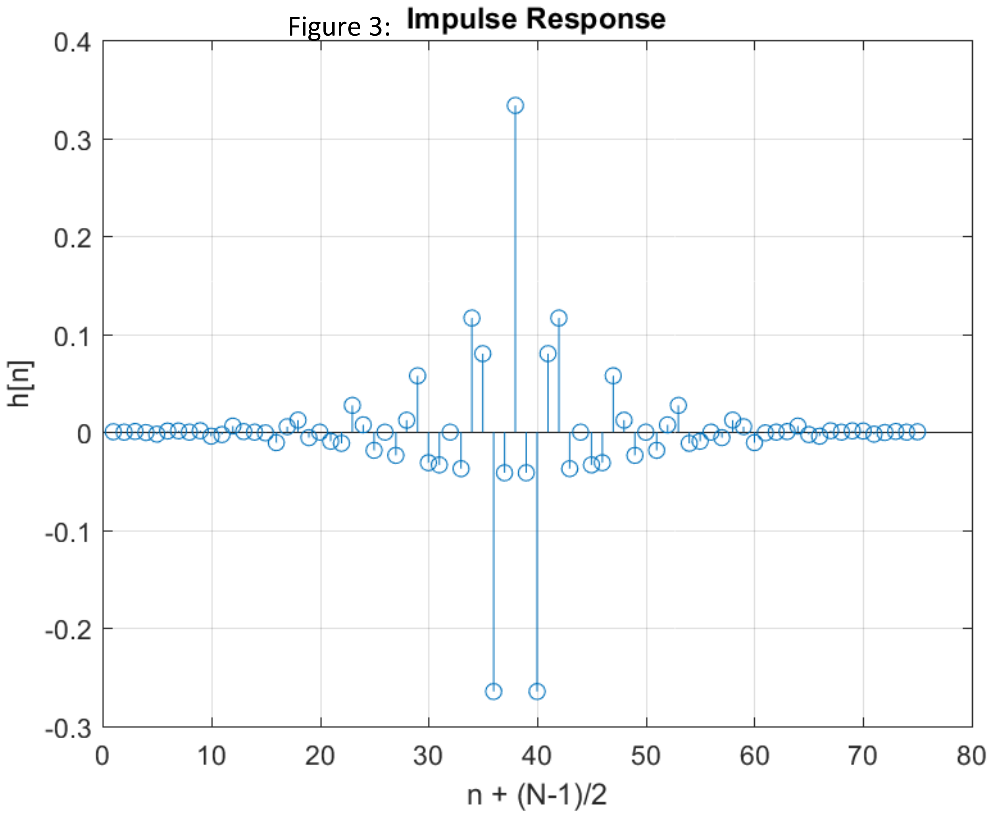
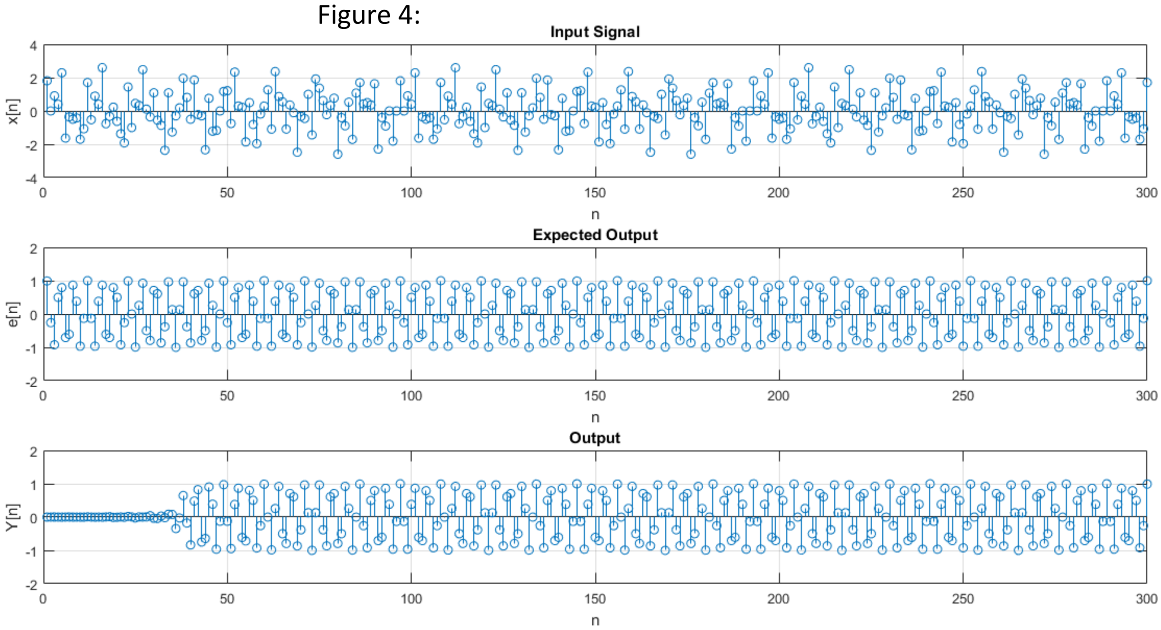
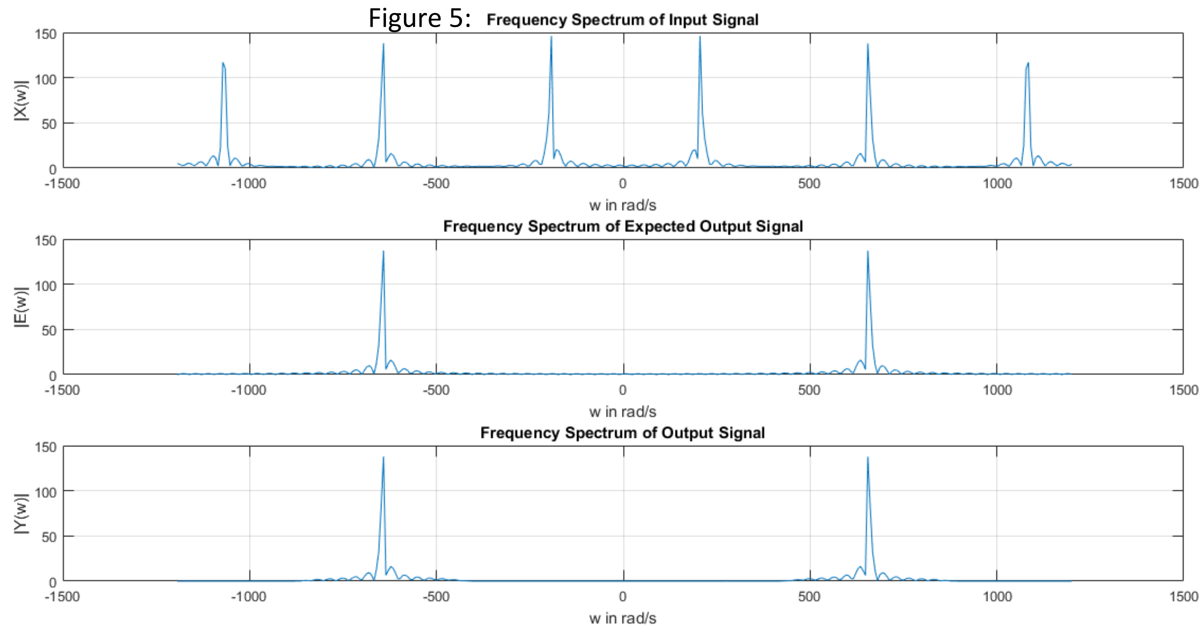


Figure 4:





IV. DISCUSSION

In figure 1 it can be seen that the frequency spectrum of the filter resembles a band pass filter. Figure 2 shows the ripples in passband even though they are invisible in figure 1. In figure 5 the first image shows the frequency spectrum of the input signal which is a combination of 3 sinusoids. The second image shows the frequency spectrum of the expected output where only the signal within the passband gets passed to the output. Image 3 shows the actual filtered output which closely resembles the expected output.

V. SUMMARY AND CONCLUSION

The filter designed for the purpose of this assignment truly acts as a band pass filter of the stipulated parameters. It is evident from the results shown in figure 5 that the filter blocks the undesired signals and passed the signals within the passband. There is a slight difference between ideally expected output and the obtained one due to the following reasons.

1. Passband ripples causes undesirable modifications to the required part of the spectrum.
2. Non-zero magnitude response in the stop band for the designed filter.
3. Stop band ripples distort the signals that need attenuation.

REFERENCES

[1] "DESIGN OF NONRECURSIVE (FIR FILTERS)" at http://www.d-filter.ece.uvic.ca/SupMaterials/Slides/DSP-Ch09-S3,4.pdf.

[2] "Bessel function" at https://en.wikipedia.org/wiki/Bessel_function

VI. APPENDIX

```
%Design of FIR Digital Filter
clc;
close all;
%parameters
Ap = 0.14; %dB maximum passband ripple
Aadesirable = 52; % dB minimum stopband attenuation
wp1 = 500; %rad/s lower passband edge
wp2 = 800; %rad/s upper passband edge
wa1 = 400; % rad/s lower stopband edge
wa2 = 950; %rad/s upper stopband edge
ws = 2400; %rad/s sampling frequency
T = 2*pi/ws;

Bt = min(wp1-wa1, wa2-wp2);
wc1 = wp1 - Bt/2;
wc2 = wp2 + Bt/2;

dp = ((10^(0.05*Ap))-1)/((10^(0.05*Ap))+1);
da = 10^(-0.05*Aadesirable);
d = min(dp, da);

Aa = -20*log10(d); %actual Aa of the filter

%kaiser window
if Aa<=21
    alpha = 0;
elseif Aa<=50
        alpha = (0.5842*((Aa-21)^0.4)) + (0.07886*(Aa-21));
else
    alpha = 0.1102*(Aa - 8.7);
end

if Aa<=21
    D = 0.9222;
else
    D = (Aa - 7.95)/14.36;
end

N = ceil((ws*D/Bt)+1);

if mod(N,2) == 0
    N=N+1;
end

wk = zeros(N,1);

for n = -(N-1)/2:(N-1)/2
   beta = alpha * (1 - (2*n/(N-1))^2)^0.5;
   numerator = myBessel(beta);
   denominator = myBessel(alpha);
   wk(n+(N-1)/2+1) = numerator/denominator;
end

stem(wk);

h = zeros(N,1);
h(38) = (2/ws)*(wc2-wc1);
for n = -(N-1)/2:(N-1)/2
```

```matlab
    if n==0
        h(n+(N-1)/2+1) = (2/ws)*(wc2-wc1);
    else
        h(n+(N-1)/2+1) = (1/(n*pi)) * (sin(wc2*n*T) - sin(wc1*n*T));
    end
end

figure;
stem(h);

fil = h.*wk;
filim =figure;
stem(fil);
title('Impulse Response');
xlabel('n + (N-1)/2');
ylabel('h[n]');
grid on;
saveas(filim, 'q4.png');

[amp, digiFreq] = freqz(fil);

analogFreq = digiFreq*ws/(2*pi);
ampdb = 20*log10(abs(amp));

fr = figure;
plot(analogFreq, ampdb);
axis([wp1 wp2 -0.05 0.05]);
title('Magnitude Response in Passband');
xlabel('w in rad/s');
ylabel('|H(w)|');
grid on;
saveas(fr, 'q3.png');

x = zeros(300,1);
w1 = wa1/2;
w2 = (wp2+wp1)/2;
w3 = (ws/2+wa2)/2;

for n = 1:300
    x(n) = sin (w1*n*T) + sin (w2*n*T) + sin (w3*n*T);
end

[amp, digiFreq] = freqz(x);

analogFreq = digiFreq*ws/(2*pi);
ampdb = 20*log10(abs(amp));

fr = figure;
plot(analogFreq, ampdb);
title('Input Signal in Frequency Domain');
xlabel('w in rad/s');
ylabel('|X(w)|');
grid on;
saveas(fr, 'q6.png');

lenin = length(x);
lenh = length(fil);
lenfft = lenin+lenh-1;
```

```
IN = fft(x, lenfft);
H = fft(fil, lenfft);
OUT = H.*IN;
out = ifft(OUT, lenfft);

threeplots = figure;
subplot(3,1,1);
stem(x);
title('Input Signal');
xlabel('n');
ylabel('x[n]');
grid on;


e = zeros(300,1);

for n = 1:300
    e(n) = sin (w2*n*T);
end
subplot(3,1,2);
stem(e);
axis([0 300 -2 2]);
title('Expected Output');
xlabel('n');
ylabel('e[n]');
grid on;

subplot(3,1,3);
stem(out);
axis([0 300 -2 2]);
title('Output');
xlabel('n');
ylabel('Y[n]');
grid on;

figure;

[amp, digiFreq] = freqz(out);

analogFreq = digiFreq*ws/(2*pi);
ampdb = 20*log10(abs(amp));



w = ws*(1-lenfft/2:lenfft/2)/lenfft;
IN1 = abs(fftshift(IN));
OUT1 = abs(fftshift(OUT));
E = fft(e, lenfft);
E1 = abs(fftshift(E));
figure;
subplot(3,1,1);
plot(w, IN1);
title('Frequency Spectrum of Input Signal');
xlabel('w in rad/s');
ylabel('|X(w)|');
grid on;

subplot(3,1,2);
plot(w, E1);
```

```
title('Frequency Spectrum of Expected Output Signal');
xlabel('w in rad/s');
ylabel('|E(w)|');
grid on;

subplot(3,1,3);
plot(w, OUT1);
title('Frequency Spectrum of Output Signal');
xlabel('w in rad/s');
ylabel('|Y(w)|');
grid on;
```