Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

Experiment 4: Design and Analysis of FIR and IIR filters

AIM: To design FIR/IIR filters in MATLAB using various algorithms

Filter specifications and design methods:

```
Max passband ripple: 1dB            Min stopband attenuation: 30dB
Lower passband edge: 0.2pi          Lower stopband edge: 0.19pi
Upper passband edge: 0.3pi          Upper stopband edge: 0.31pi
1. FIR Window (Rectangular, Hanning)
2. FIR LS
3. FIR Park McClellan
4. IIR Butterworth
5. IIR Chebyshev
6. IIR Inverse Chebyshev
7. IIR Elliptic
```

For the purpose of designing minimum order filters, for the sake of exploration, we have used our own code. However, a more accurate way to do this is to use the MATLAB filter design tool, as we have done in the subsequent parts for seeing the effect of bit precision on filter response, passband ripple, stopband attenuation and stability

Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

Code:

Experiment4.m

```
%% 1.a FIR-Rectangular window
N=512;
h1 = fir1(N, [0.199 0.301], 'bandpass', rectwin(N+1));
freqz(h1, 1, 10000);
%figure;
%h1_dash = bitprecision(10,1,N+1,h1);      %used for finite word length
%fvtool(h1,1,h1_dash,1);
%% 1.b.FIR-Hanning window
N=400;
h1 = fir1(N, [0.198 0.302], 'bandpass', hann(N+1));
figure;
freqz(h1,1,10000);
%% 2.FIR-LS
h1 = firls(155,[0 0.19 0.203 0.297 0.31 1],[0 0 1 1 0 0]);
figure;
freqz(h1,1,10000);
%% 3.FIR-Parks-McClellan
N=175;
h1=firpm(N,[0 0.19 0.205 0.295 0.31 1],[0 0 1 1 0 0]);
figure;
freqz(h1,1,10000);
%% 4.Butterworth IIR
[z,p,k] = butter(52,[0.2 0.3],'bandpass');
sos = zp2sos(z,p,k);
fvtool(sos);
xlim([0.18 0.32]);
ylim([-45 5]);
%% 5.Chebyshev IIR
[z,p,k] = cheby1(18,1,[0.2 0.3],'bandpass');
sos = zp2sos(z,p,k);
fvtool(sos);
xlim([0.18 0.32]);
ylim([-45 5]);
%% 6.Inverse Chebyshev IIR
[z,p,k] = cheby2(18,30,[0.19 0.31],'bandpass');
sos = zp2sos(z,p,k);
fvtool(sos);
xlim([0.18 0.32]);
ylim([-45 5]);
%% 7.Elliptic IIR
[z,p,k] = ellip(10,1,30,[0.2 0.3],'bandpass');
sos = zp2sos(z,p,k);
fvtool(sos);
xlim([0.18 0.32]);
ylim([-45 5]);
```

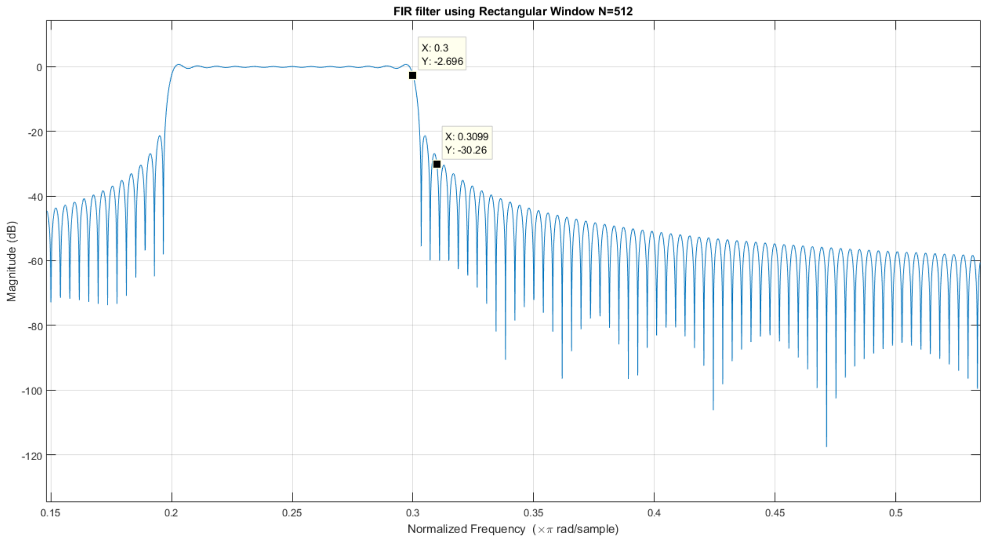
Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

bitprecision.m

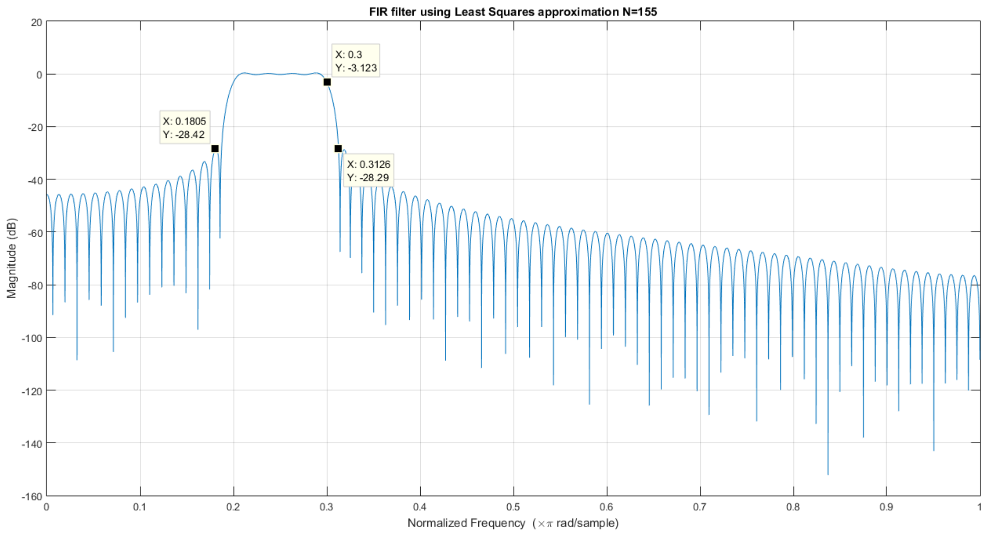
```
function a_dash = bitprecision(m,n,N,a)
d2b=zeros(floor(N),floor(m+n));

for i=1:N
    d2b(i,:) = fix(rem(a(i)*pow2(-(n-1):m),2));     %de2bi with precision
of n,m bits in integral and fractional parts
end

a_dash = d2b*pow2(n-1:-1:-m).';
a_dash = a_dash';
end
```

Here are the obtained filter responses, along with order for the 8 specified designs:

1. FIR – Rectangular Window

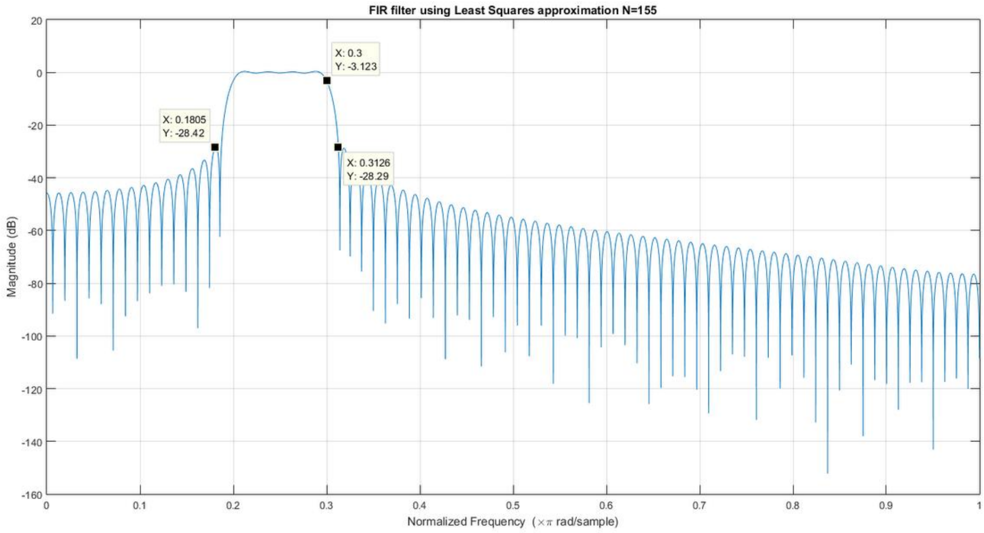


2. FIR – Hanning Window

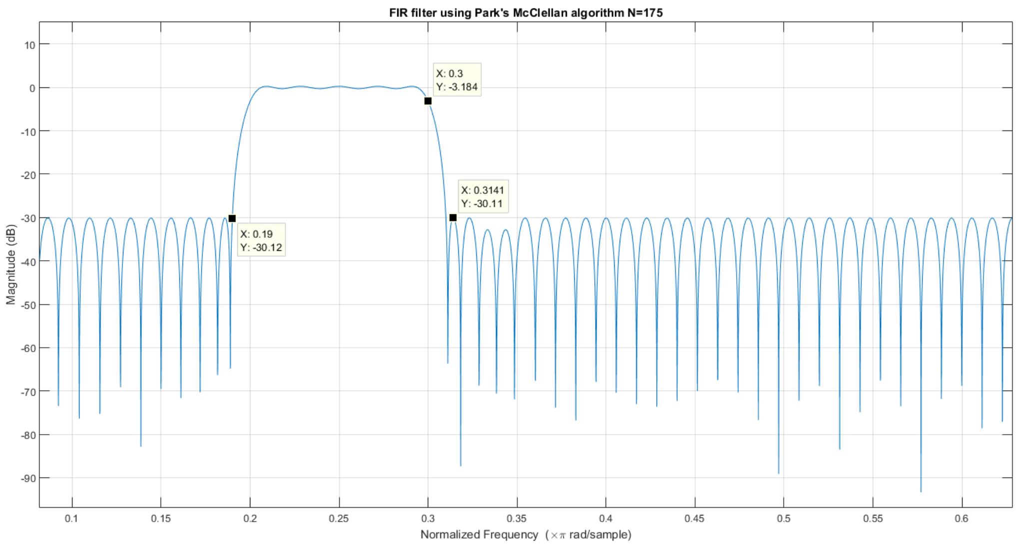


Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

3. FIR – Least Squares approximation

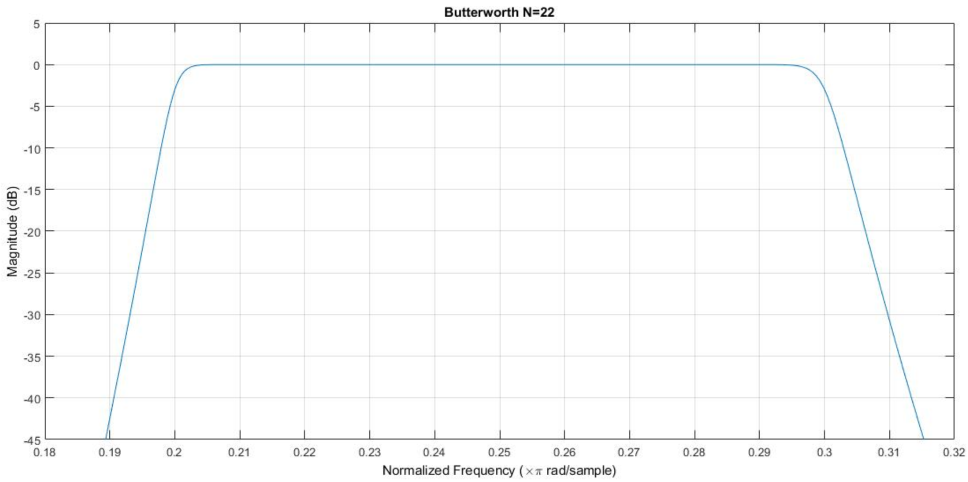


4. FIR – Parks McClellan algorithm

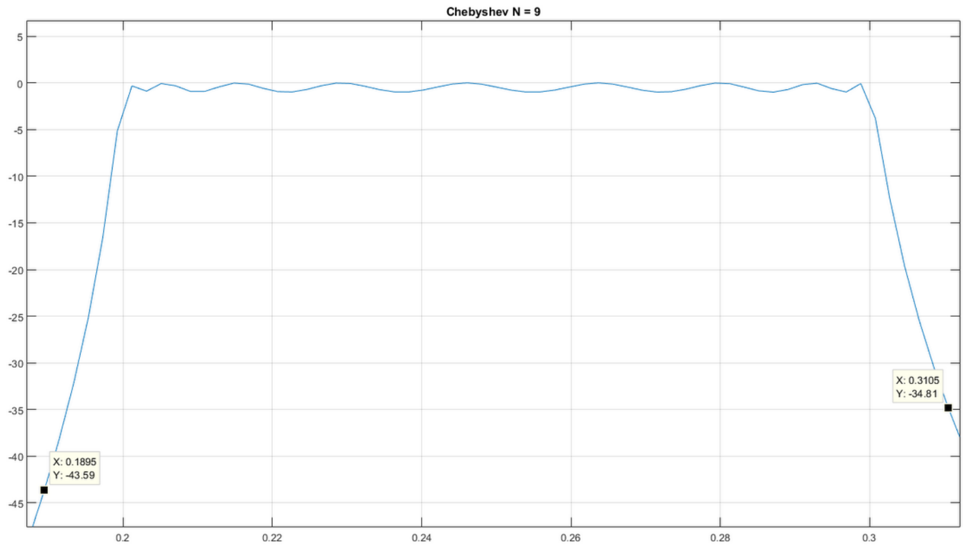


Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

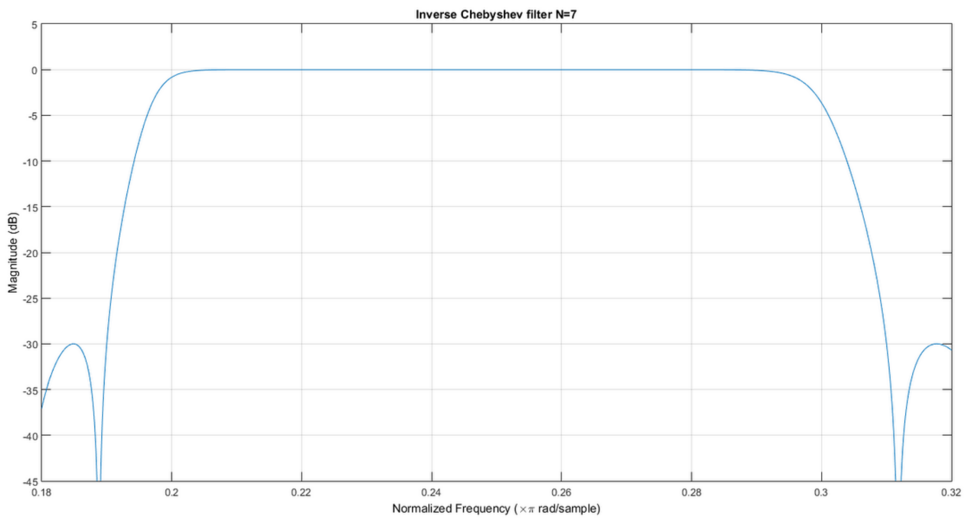
5. <u>IIR – Butterworth</u>



6. <u>IIR – Chebyshev (Chebyshev Type I)</u>

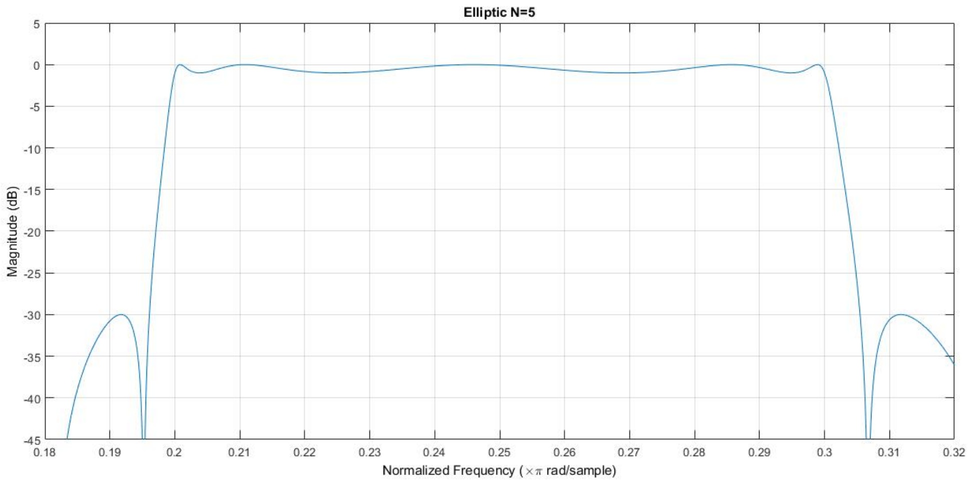


7. <u>IIR – Inverse Chebyshev (Chebyshev Type II)</u>

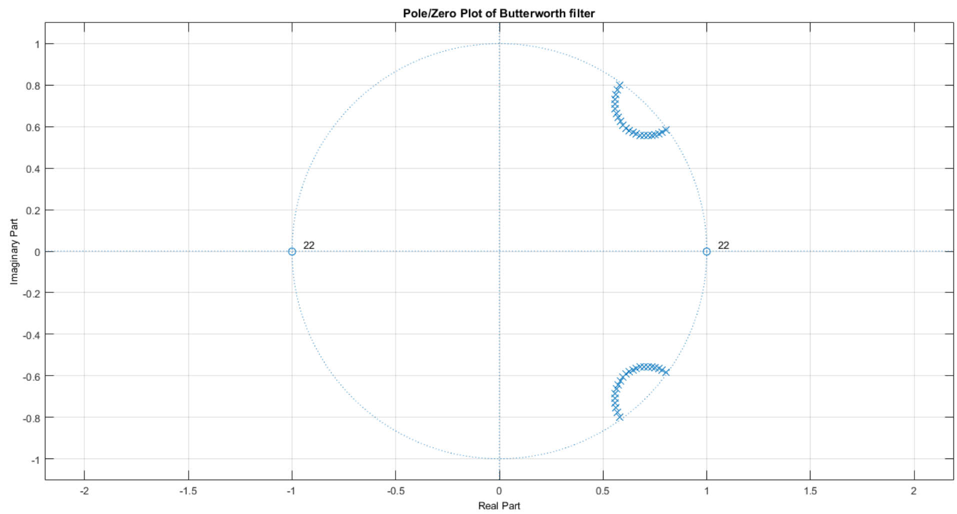


Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

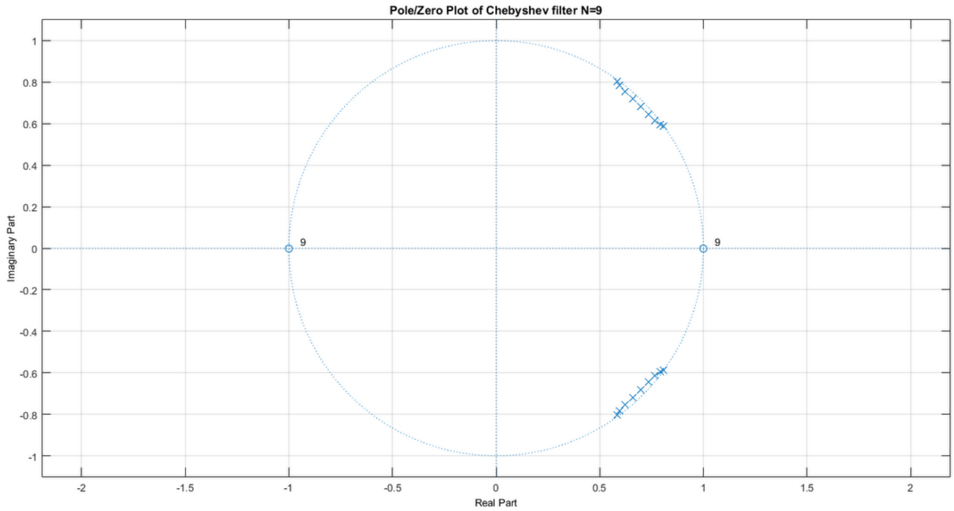
8. IIR – Elliptic

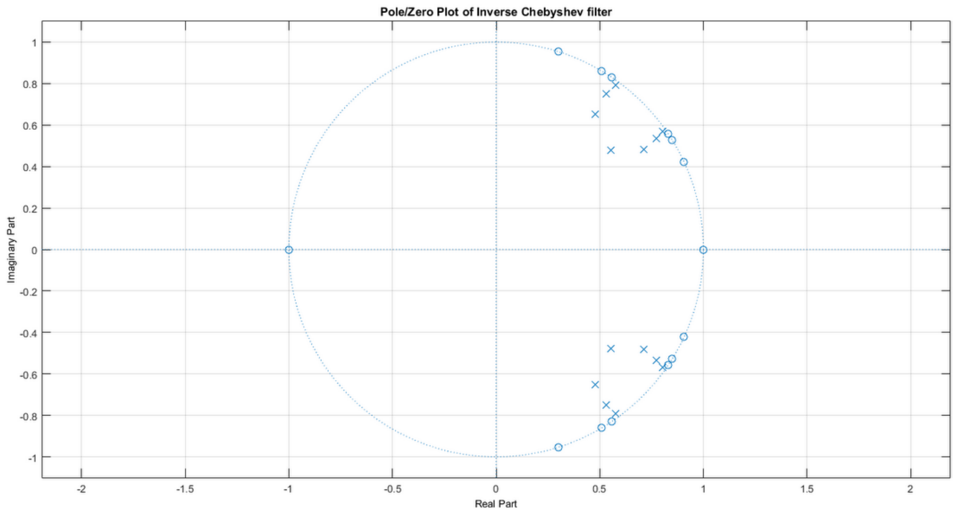


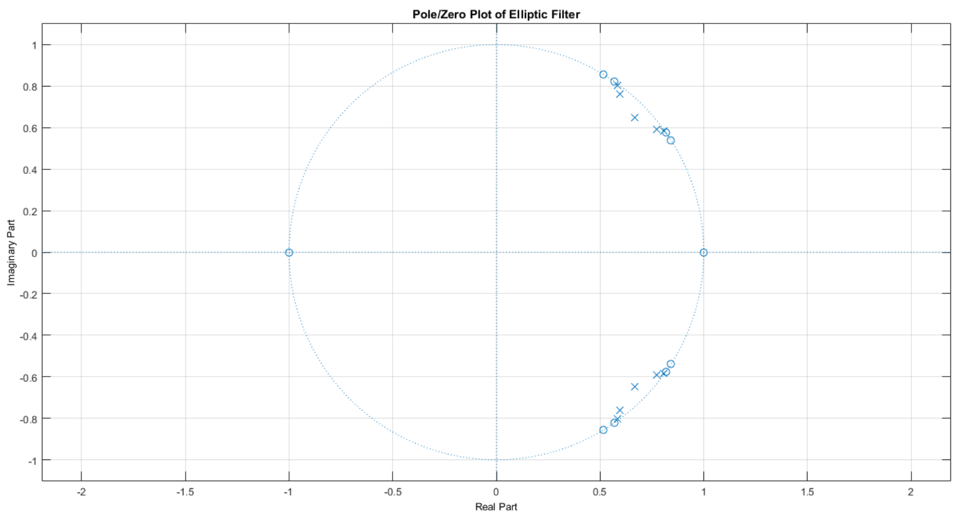
We can use the fvtool filter visualizer window to view the pole zero plots for different filters. FIR filters don't have stability issues, as there is no feedback. However, for IIR filters, we need to ensure that all the poles are inside the unit circle, for the filter to be stable. As we can see in the following plots, this is indeed true.



Aditya Sinha 14EC10002
Saurabh Dash 14EC10050







Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

There are a few things worth noting from these plots. Firstly, IIR filter orders are much lesser than FIR filter orders. This is purely due to the fact that IIR filters are acausal. Secondly, in FIR filters, the required specifications are met at the least order for the Parks – McClellan algorithm (N=135), followed by Least Squares method and Hanning window, least effective being Rectangular window.

Next, we try to see the effect of bit precision on the filter response and stability. Note that we'll only be doing this for IIR filters, as they are the only ones who have stability issues. Also, the values of passband ripple and stop band attenuation have minimal variation in the FIR case.

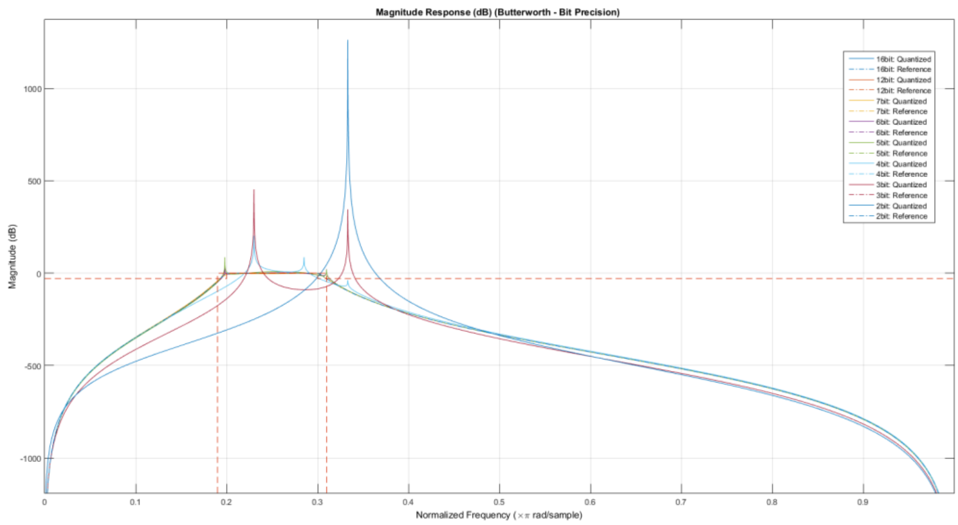
Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

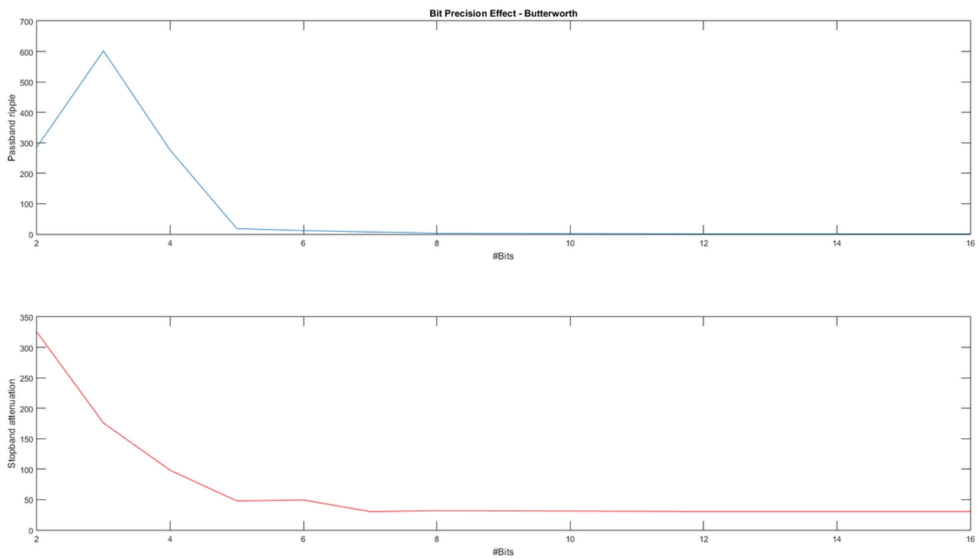
Effect of Bit Precision:

Changing the bit precision of the coefficients from the earlier double precision floating point to the now fixed point leads to a degradation in filter response, and after a certain point makes the filter response unstable, leading to drastic change in passband ripple and stop band attenuation.

Here are the plots of the filter response and the variation of passband ripple and stopband attenuation:

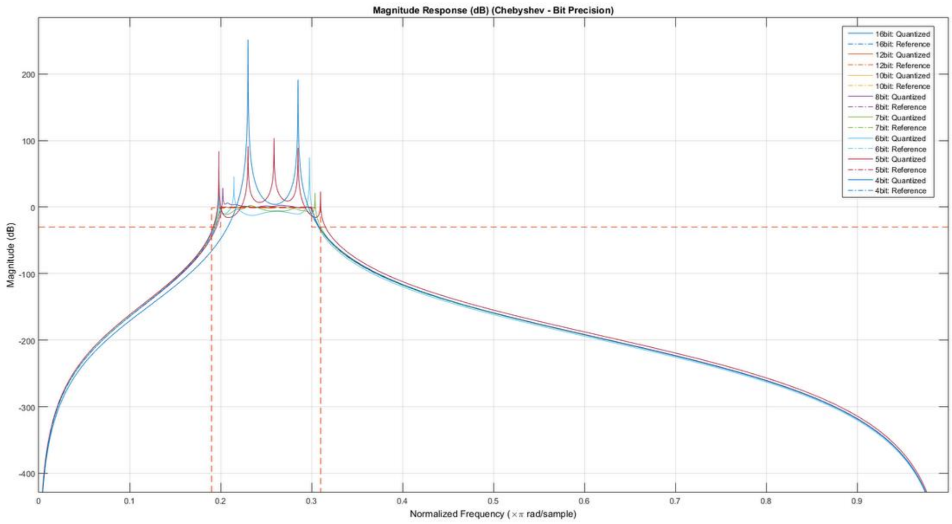
1. <u>IIR – Butterworth</u>

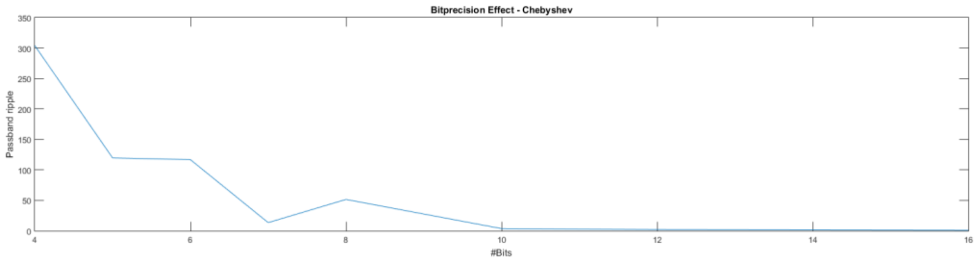


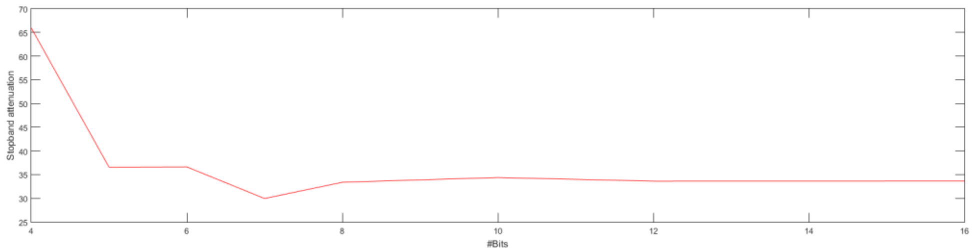


Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

2. IIR – Chebyshev (Chebyshev Type I)

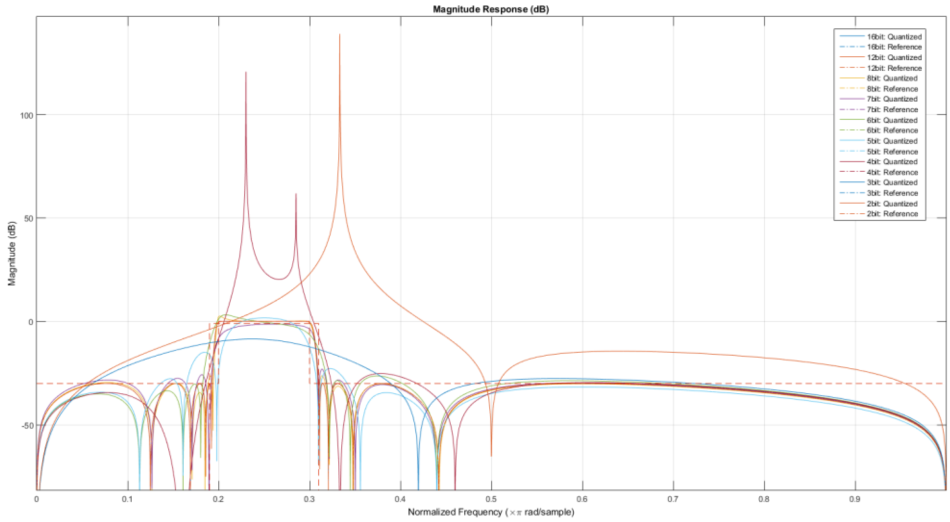


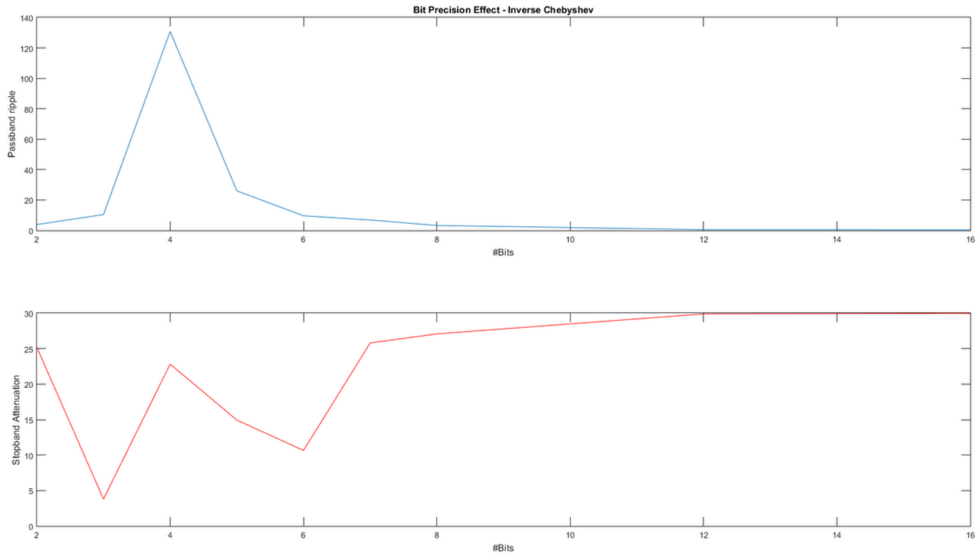




Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

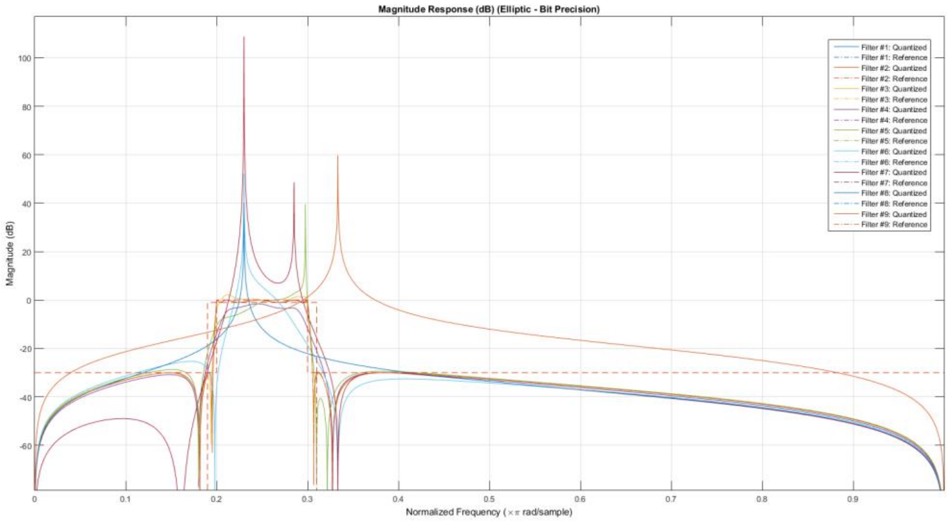
3. IIR – Inverse Chebyshev (Chebyshev Type II)

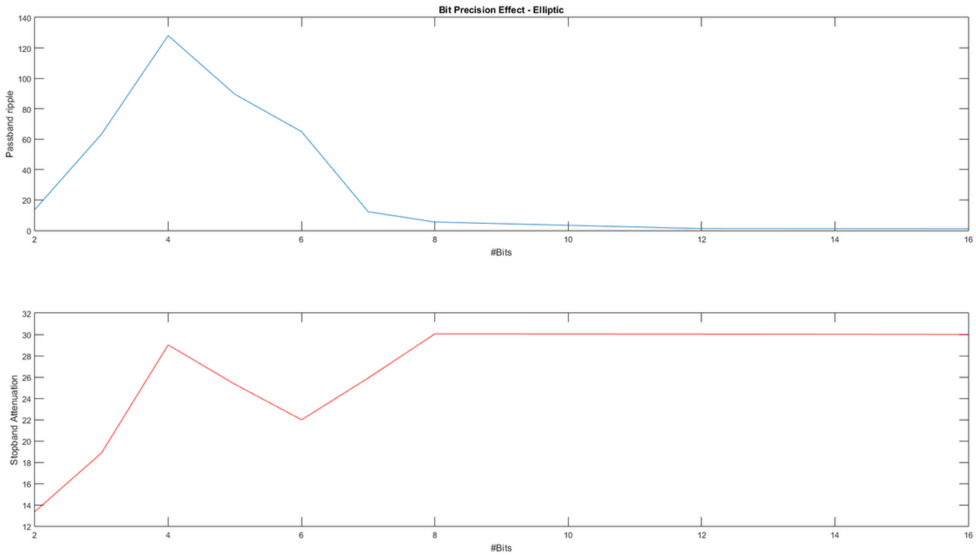




Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

4. <u>IIR – Elliptic</u>





<u>Butterworth:</u> Unstable when #Bits <= 5
<u>Chebyshev-I:</u> Unstable when #Bits <= 7
<u>Chebyshev-II:</u> Unstable when #Bits <= 4
<u>Elliptic:</u> Unstable when #Bits <= 8

Aditya Sinha 14EC10002
Saurabh Dash 14EC10050

Discussions:

Aditya Sinha (14EC10002)

1. Although IIR filters have lesser order than FIR filters, they are more difficult to implement, having more complex hardware (need for feedback). Also, there is the added fact that they are acausal, making them unfit for real time applications.
2. Elliptic IIR filter has a sharp cut-off and meets the requirements at the least order. This is the reason why it is so widely used in spacecraft applications for sensing low voltages.
3. While seeing the bit precision effect, we saw that below a certain number of bits, the filter becomes unstable and poles move outside the unit circle. This happens at a higher number of bits for elliptic filter. So, although elliptic filters are the sharpest, they are also the most vulnerable to stability issues and their passband ripple and stop band attenuation quickly degrade with truncation of bits. So, to implement the elliptic filter, we need more complex hardware.
4. In rectangular window, on increasing the filter order, we see very little change in the stop band attenuation. This is as rectangular windowing method is designed to have a sharp cutoff and the stop band attenuation starts saturating after a order of about 100.

Saurabh Dash (14EC10050)

1. The filter design tool gives us a better result, as it internally uses functions like ellipord() to compute the minimum order filter. Plus, it is indispensable for the purpose of seeing bit precision effect.
2. We see that as the filter response starts becoming unstable, there is a sharp change in the passband ripple and stopband attenuation.
3. Elliptic filter has the most stability concern among the IIR filters. This is a tradeoff that we have to deal with for obtaining a sharp filter. Since elliptic filters normally have a higher ripple in the passband as well as stop band when compared to Butterworth or Chebyshev, so their response also degrades more quickly with change in bit precision.