Date: 09.05.2022 8:15-12:00
Author: Sławomir Batruch

EDISP Lab 4

Digital Filters

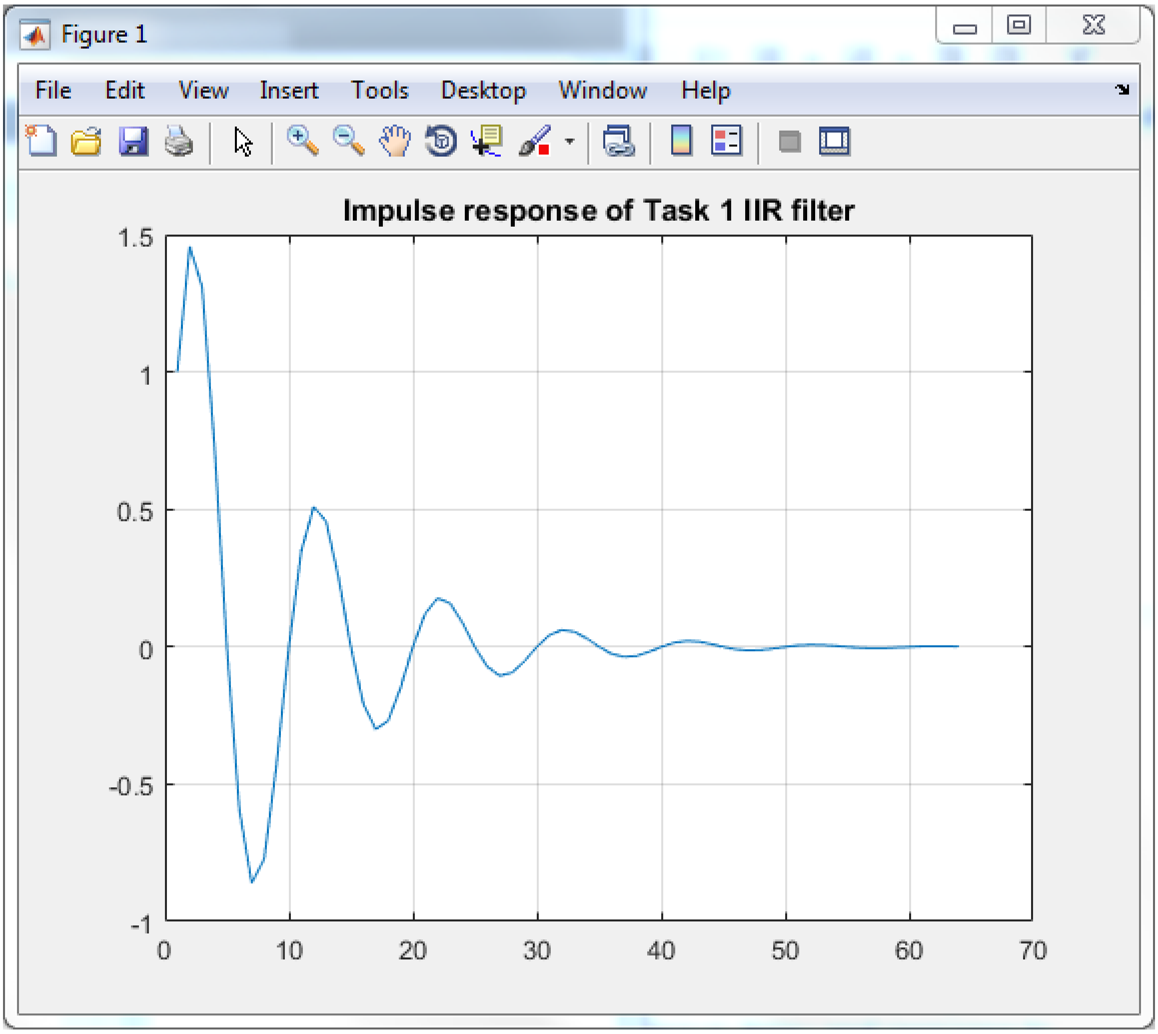
Task 1

2nd order IIR filter analysis

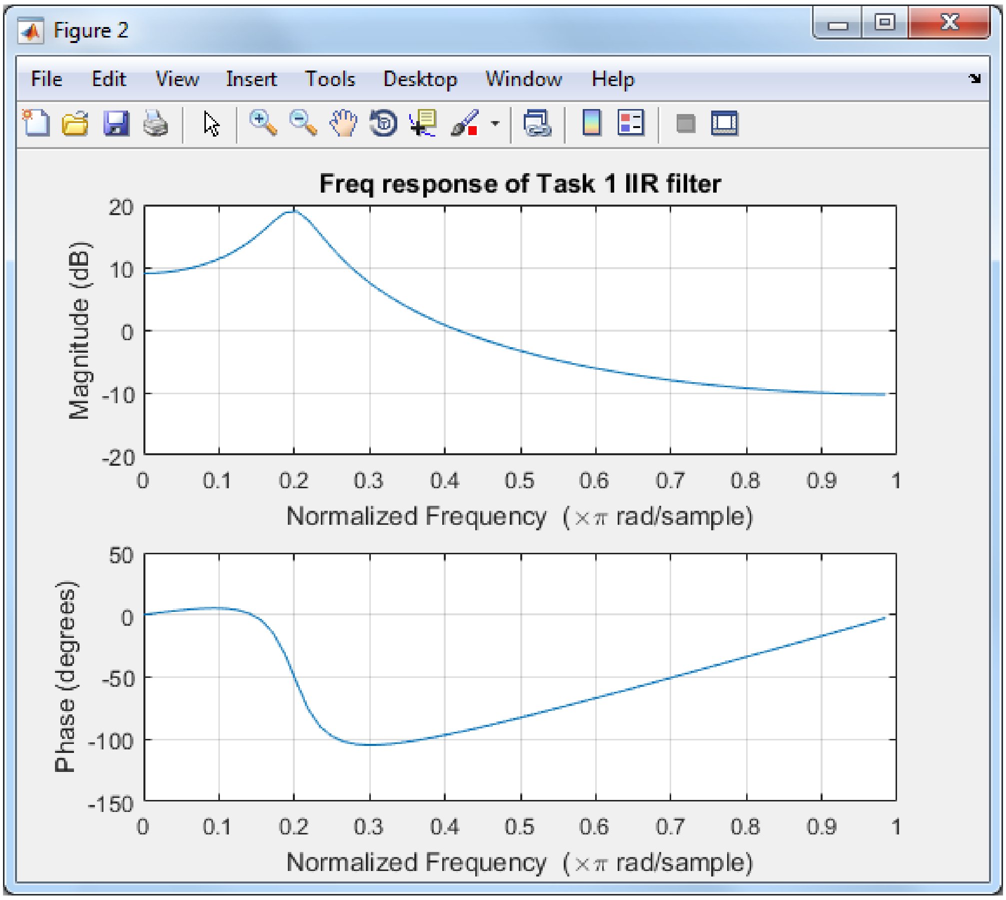
For this task we design a 2nd order IIR filter in MATLAB, and then analyze basic properties of it: impulse and frequency response; group delay and response to exemplary sinusoidal signals.

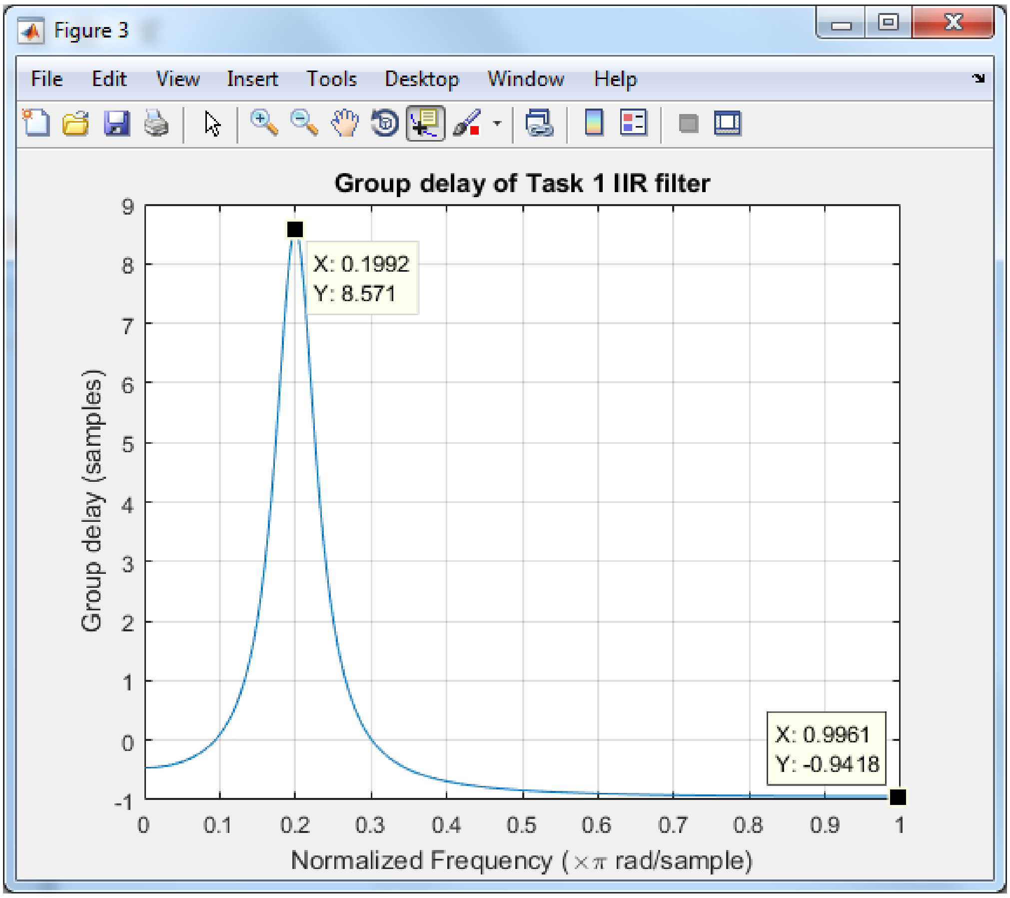
The properties of the filter for this task are as follows:

- no zeros
- poles at `0.9*exp(1j*0.2pi)` and `0.9*exp(-1j*0.2pi)`

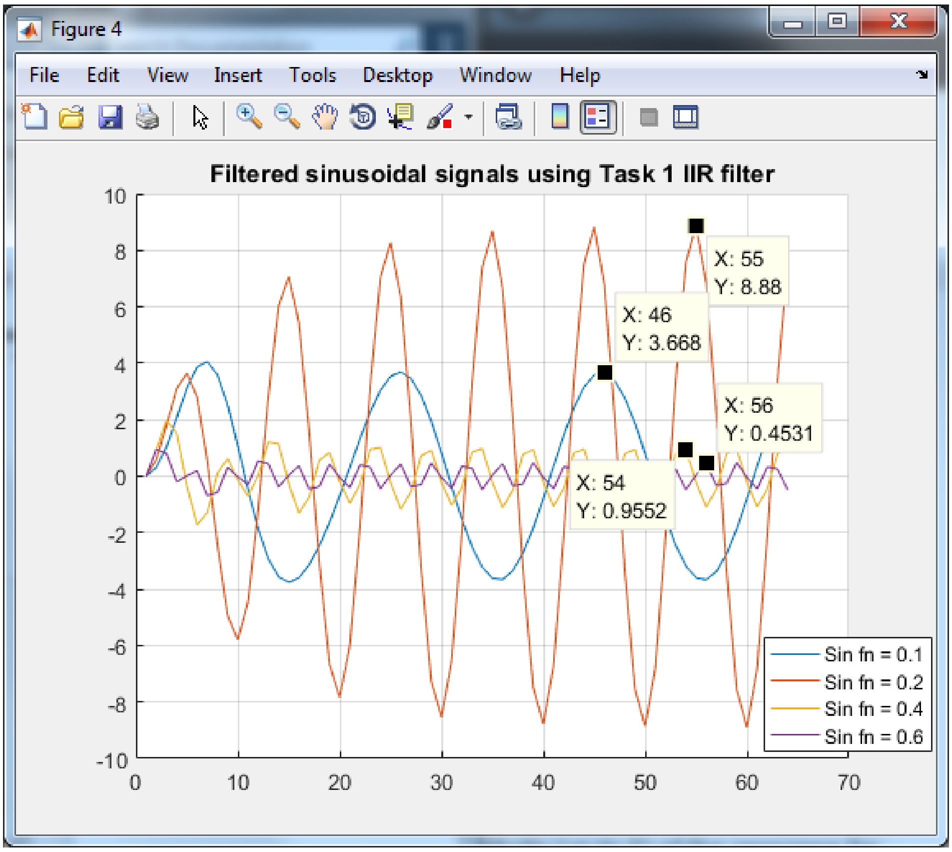


Oscillation period is about 10 samples.

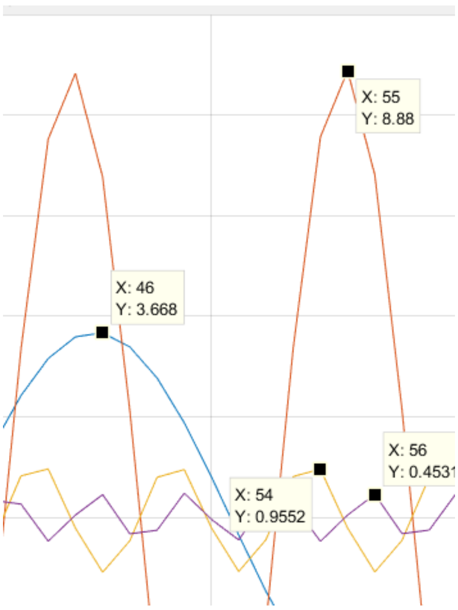


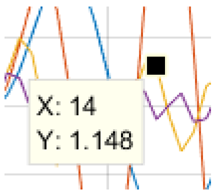


Min and max values of group delay are marked on plot



Here we see that the filter yields the highest magnitude (up to 9) of the response for the sine wave with fn = 0.2. For sine waves with higher fn, the magnitude of the sine is attenuated. This characteristic is similar to a low-pass filter. Frequencies lower than about fn = 0.4 are amplified. Frequencies higher than that are attenuated. This can be seen from the frequency response of the filter and when looking at how the sinusoidal signals are filtered.





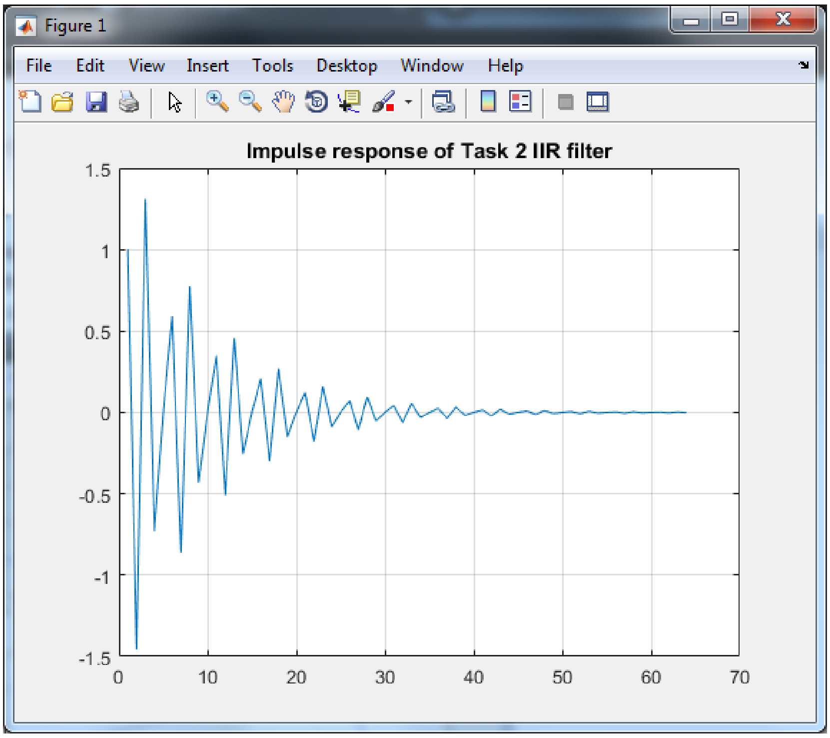
x: 46 is for fn 0.1; x: 55 for fn 0.2; x:54 and x: 14 for fn 0.4 and x:56 for fn 0.6 amplitude of fn 0.4 being about 1 checks out with the frequency response graph (about 0 dB gain in filter for fn = 0.4). We see that for x: 14 it is 1.15

Task 2

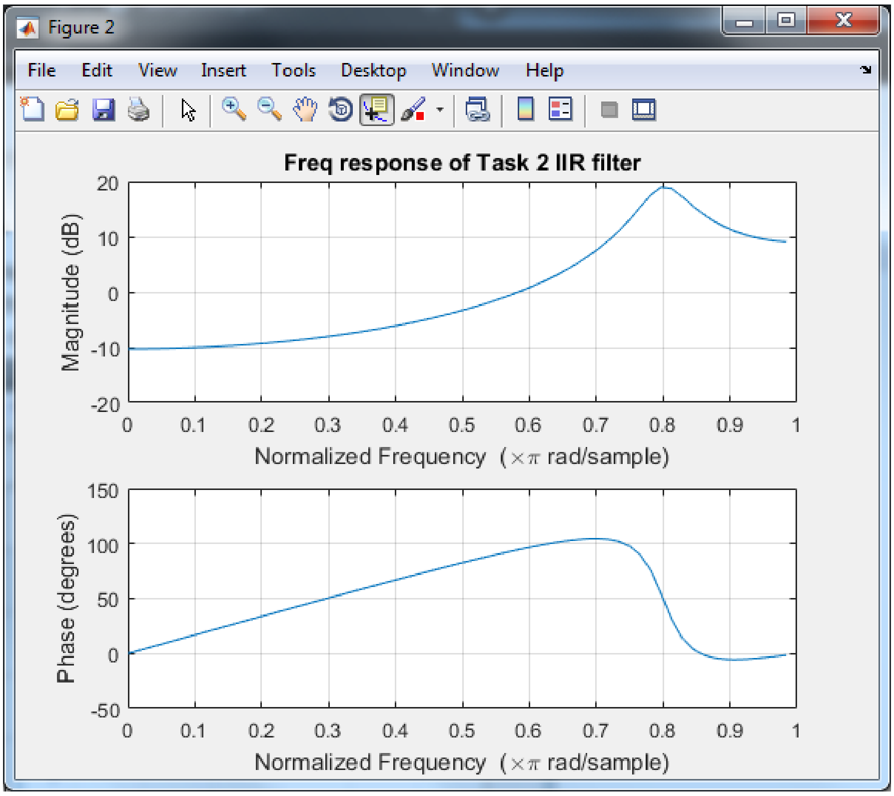
2nd order IIR filter analysis

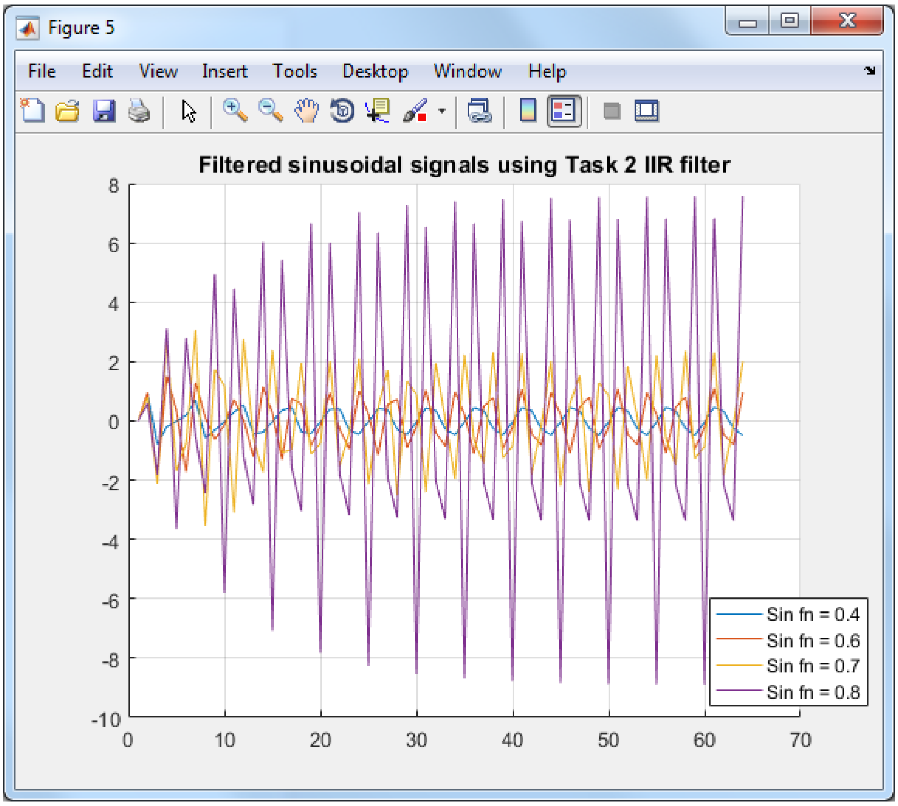
We do the same analysis as in task 1, but with changed configuration. This time, we change the location of poles to `0.9*exp(1j*0.8pi)` and `0.9*exp(-1j*0.8pi)`

Now we obtain the following results:

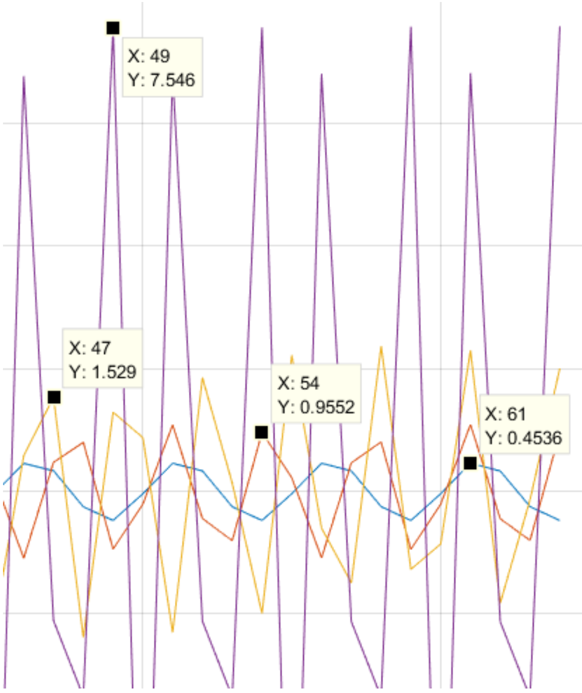


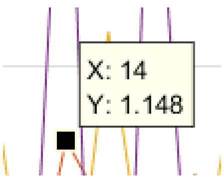
Period is 2 or 3 samples depending on the place we check.





For task 2 IIR filter, we see that the reverse is true. Lower frequencies get attenuated, whereas higher frequencies get amplified. The response of the filter to the sinusoidal signals with provided frequencies matches with the frequency response of the filter provided earlier. We can clearly notice the highest amplification is for the sinusoidal wave of fn = 0.8, as indicated with about 20 dB magnitude in the frequency response for fn = 0.8. Amplitude of fn 0.6 sine is about 1, which also checks out with the frequency response shown earlier.



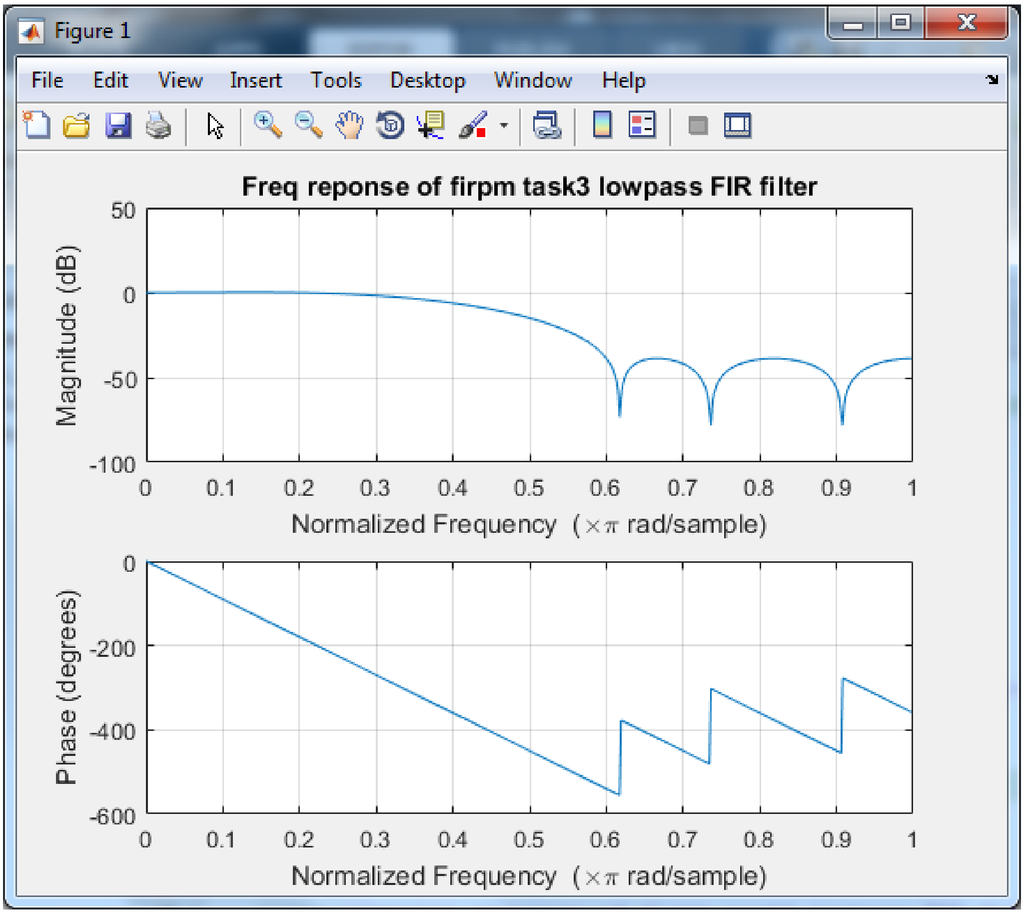


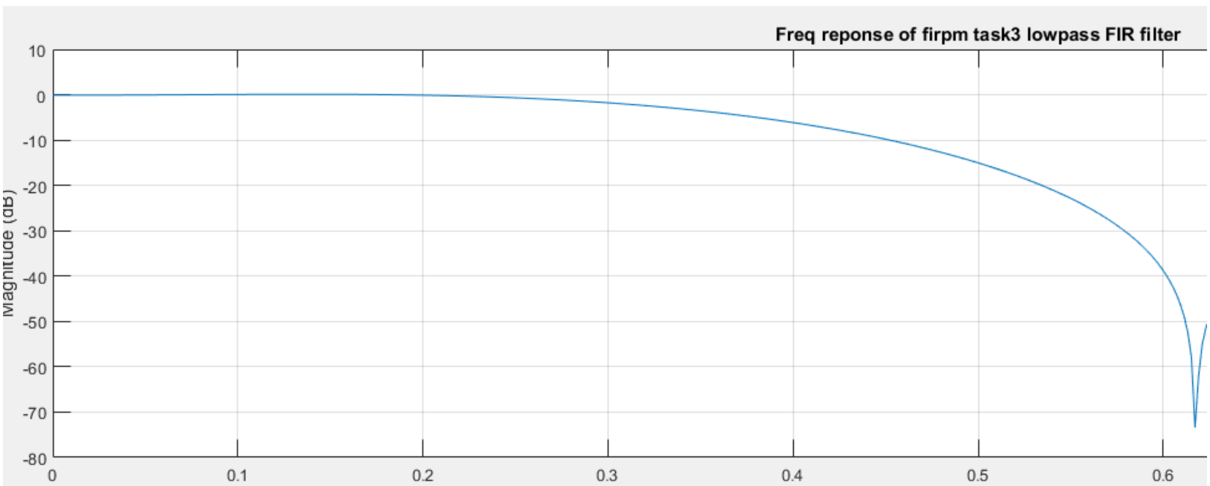
x:61 for fn 0.4; x:54 and x:14 for fn 0.6; x:47 for fn 0.7 and x:48 for fn 0.8

Task 3

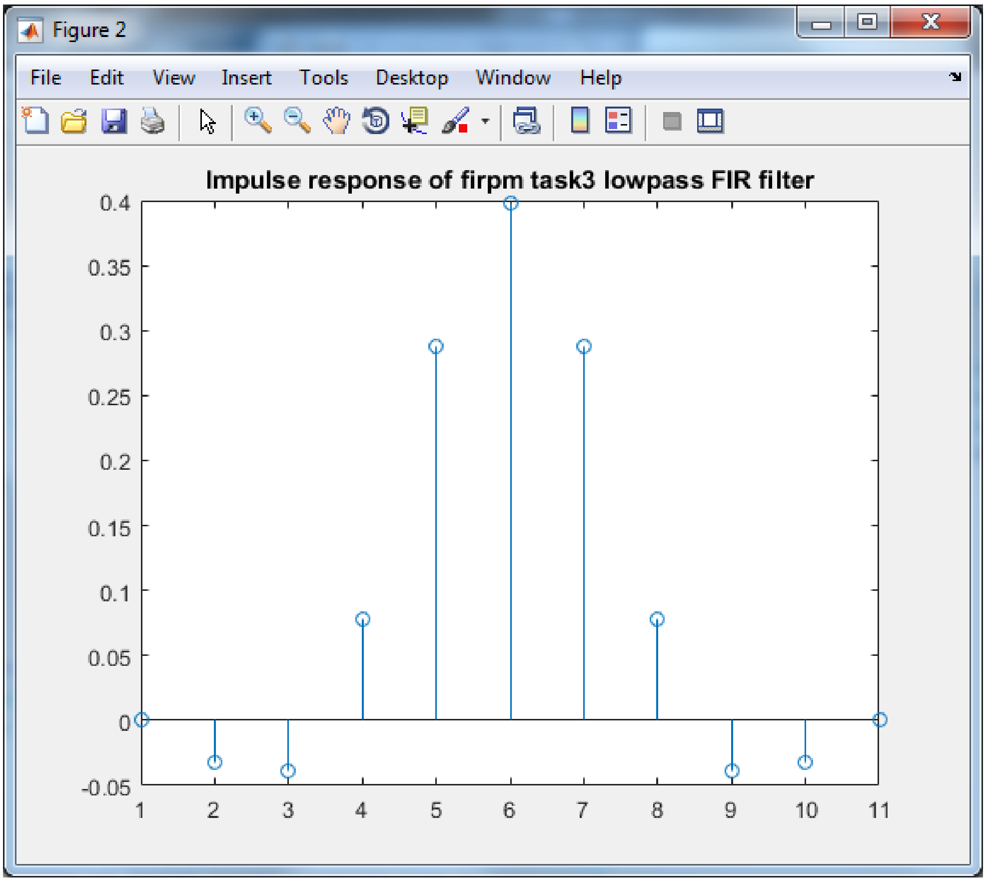
Lowpass FIR filter design using `firpm`

In this task we design a lowpass FIR filter with passband from 0.2pi, stopband from 0.6pi, and order = 10. We will use `firpm` matlab function.

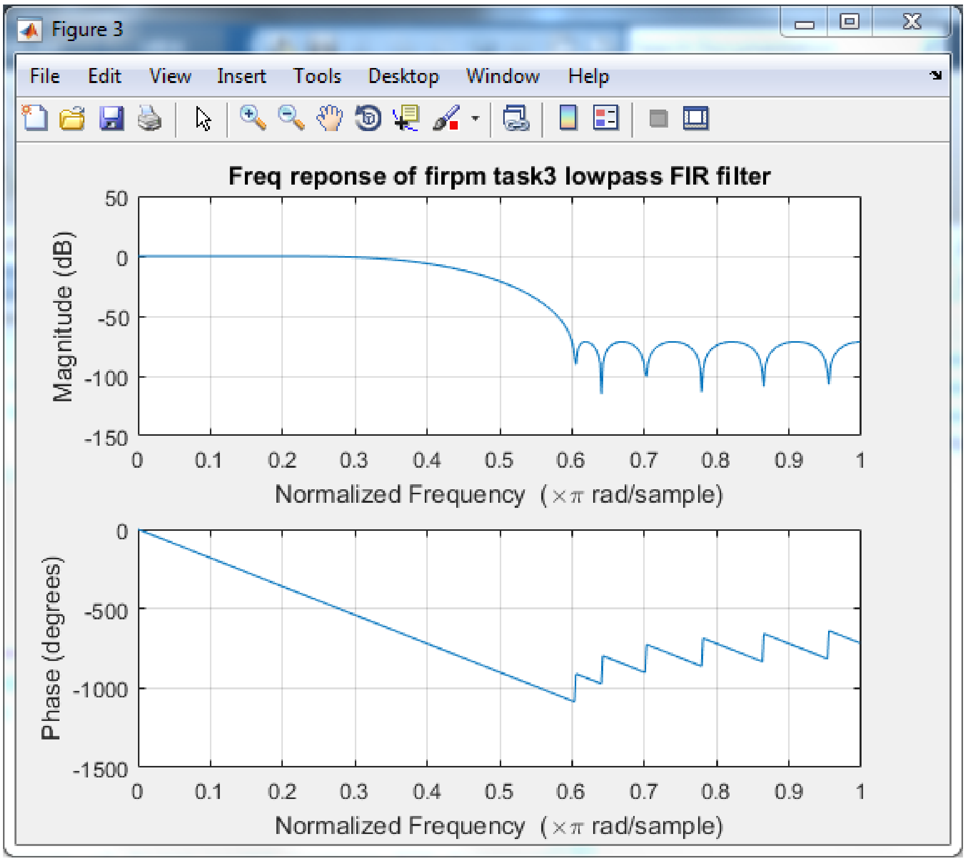


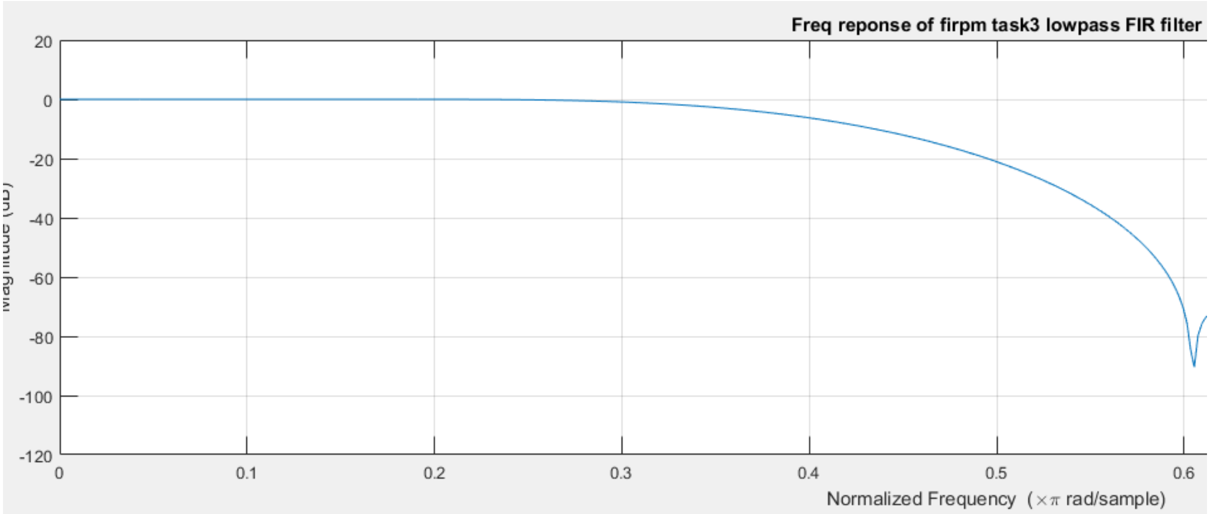


Frequency response shows us the design is good, but of course not perfect. At fn = 0.2, the magnitude of freq response starts to drop slowly at the start. The higher the fn, the bigger the drops in magnitude, reaching the lowest magnitude at around the stopband frequency of fn = 0.6. Perfect passband to fn = 0.2 should have a more significant drop in freq response magnitude at this fn. The signal is significantly attenuated (-10 dB) for fn > 0.45

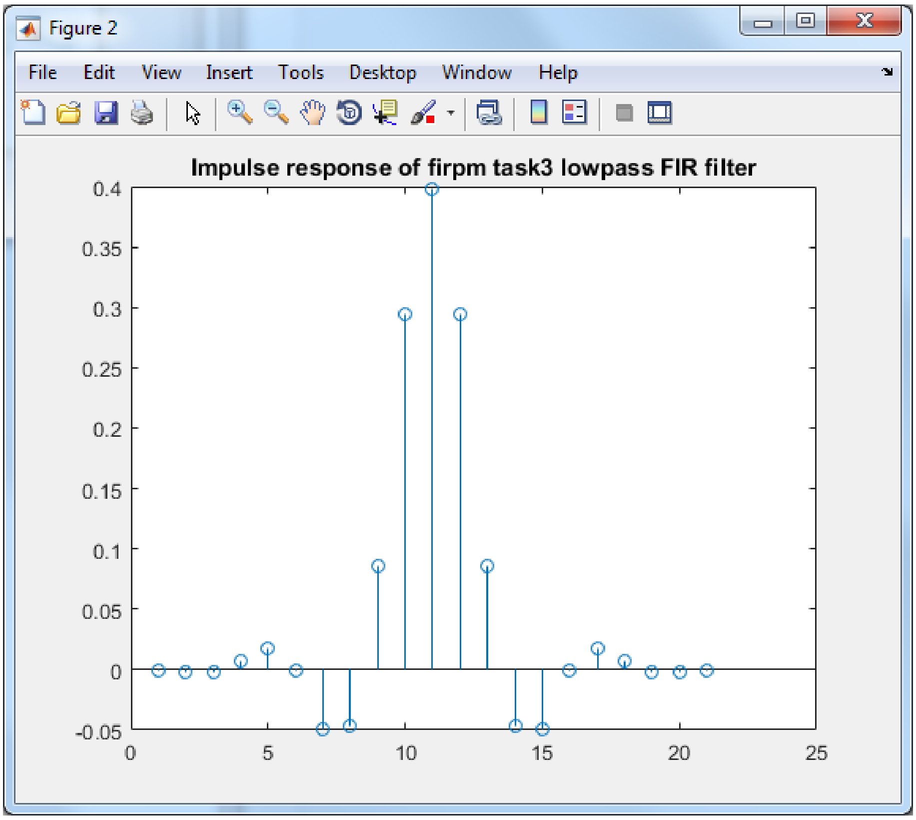


Below is a freq response for the same filter, but with order N = 20:





The difference between N = 10 is that the magnitude starts to drop for higher fn, but it has a bigger drop of magnitude overall, especially at fn = 0.6 (around 15 dB difference). Proper attenuation (<= -10 dB) occurs for roughly fn > 0.42

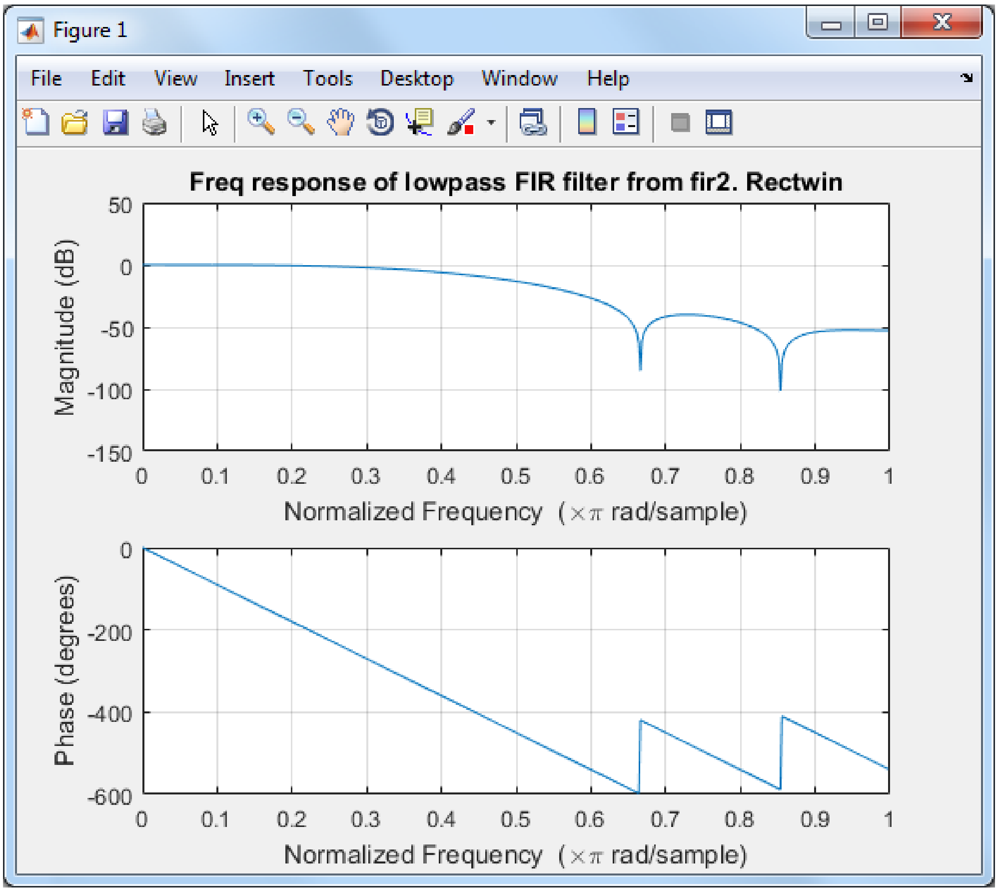


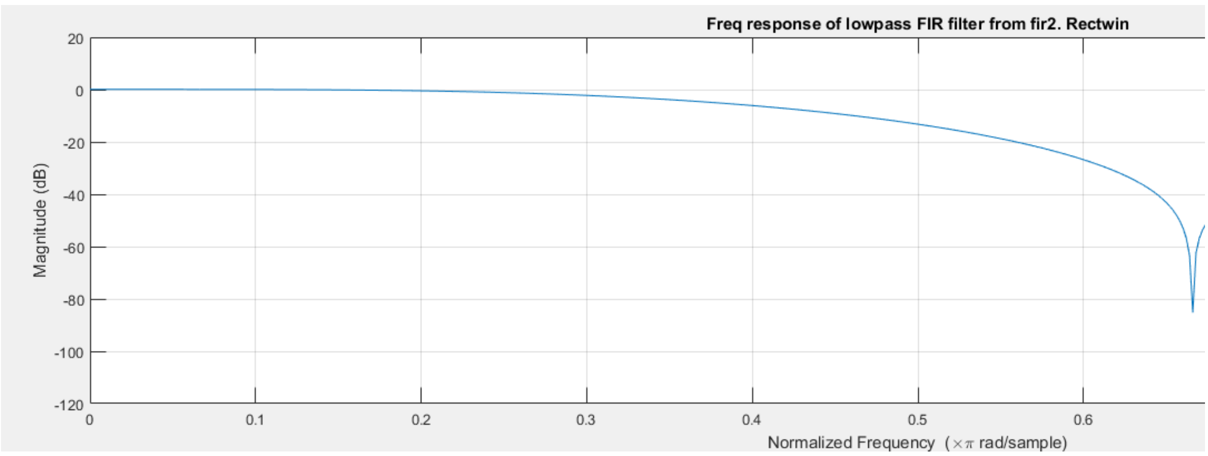
The sidelobes in the stopband have a similar shape to a frequency spectrum sidelobes of e.g. boxcar window of 8 samples, with the difference that for stopband sidelobes, the magnitude of sidelobes is constant, whereas for boxcar freq spectrum the sidelobes have lower magnitude the higher the normalized freq.

Task 4

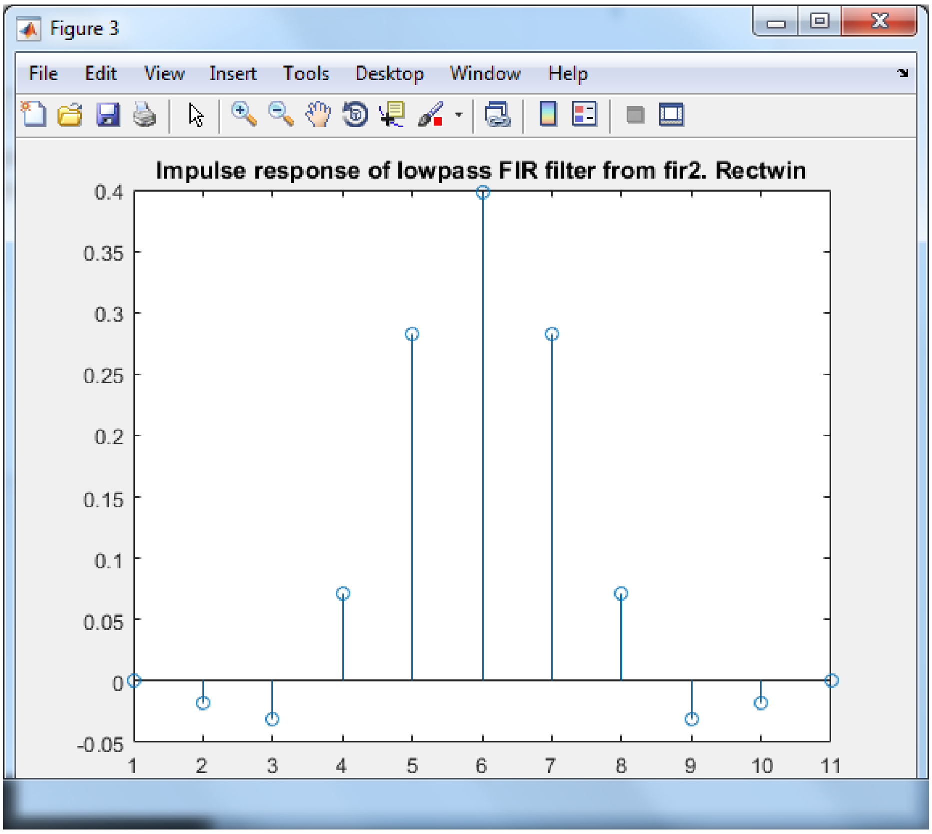
Lowpass FIR filter design using window methods with `fir2`

In this task we design the filters with the same settings as in **task 3**, but this time we use `fir2` matlab function, with the addition of window methods. We will design two filters - one with a boxcar window, one with hamming. Obtained results are:

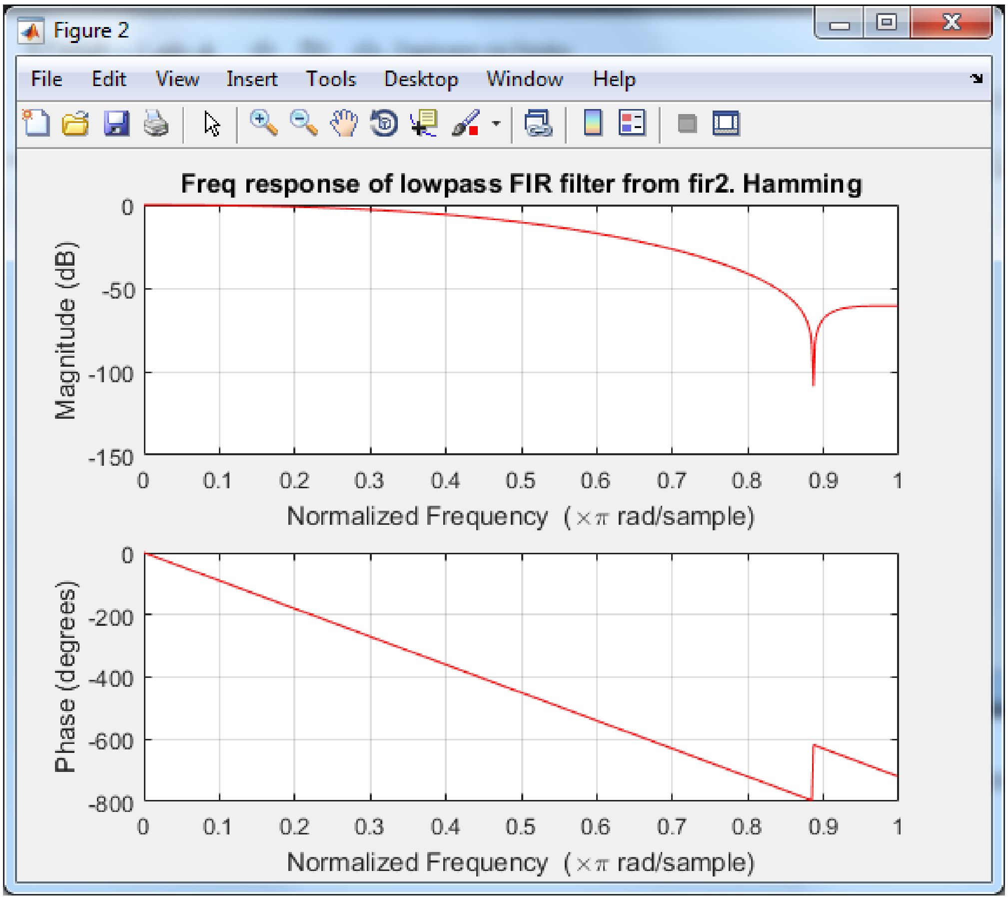


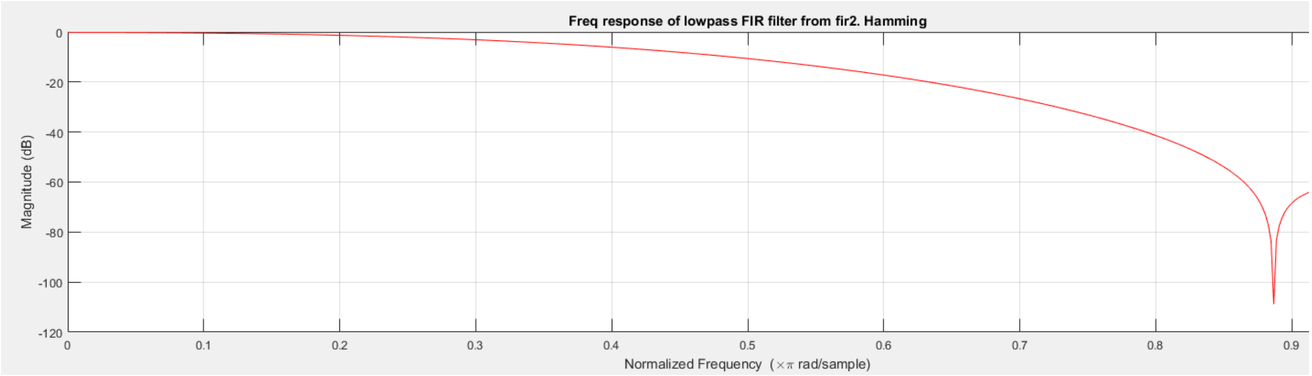


A small downside here is that the maximum magnitude drop occurs around fn = 0.65 rather than 0.6. This shouldn't have much of an effect, since the attenuation is high enough already at 0.6 fn. <-10dB is for fn around 0.45



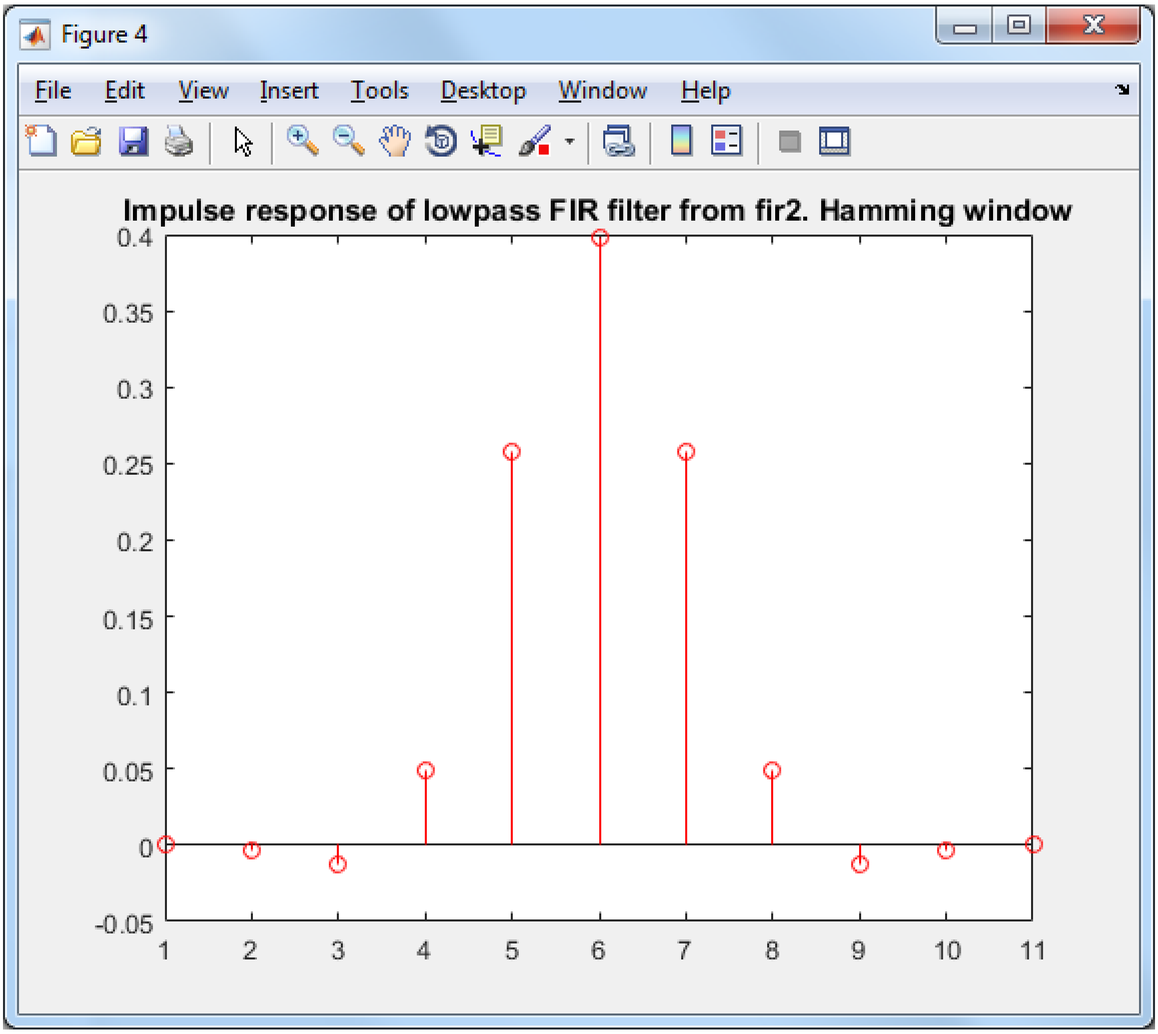
Now the hamming window case:





'Fir2' vs 'Firpm':

Overall comparing the frequency response of `fir2` filter vs `firpm`, we see that freq response in `fir2` starts to drop in magnitude for smaller fn, but the slope at which the drop occurs is smaller than for firpm. But, since -10 dB or less occurs from the similar fn as in the previous cases (which is around 0.45 fn), we can say that the transition band is better for the `fir2` filter designer, but the stopband might be worse in some extreme cases, as the magnitude for fn > 0.6 for `fir2` is higher than for firpm, which is worse because magnitude should be as close to -inf as possible for the stopband.

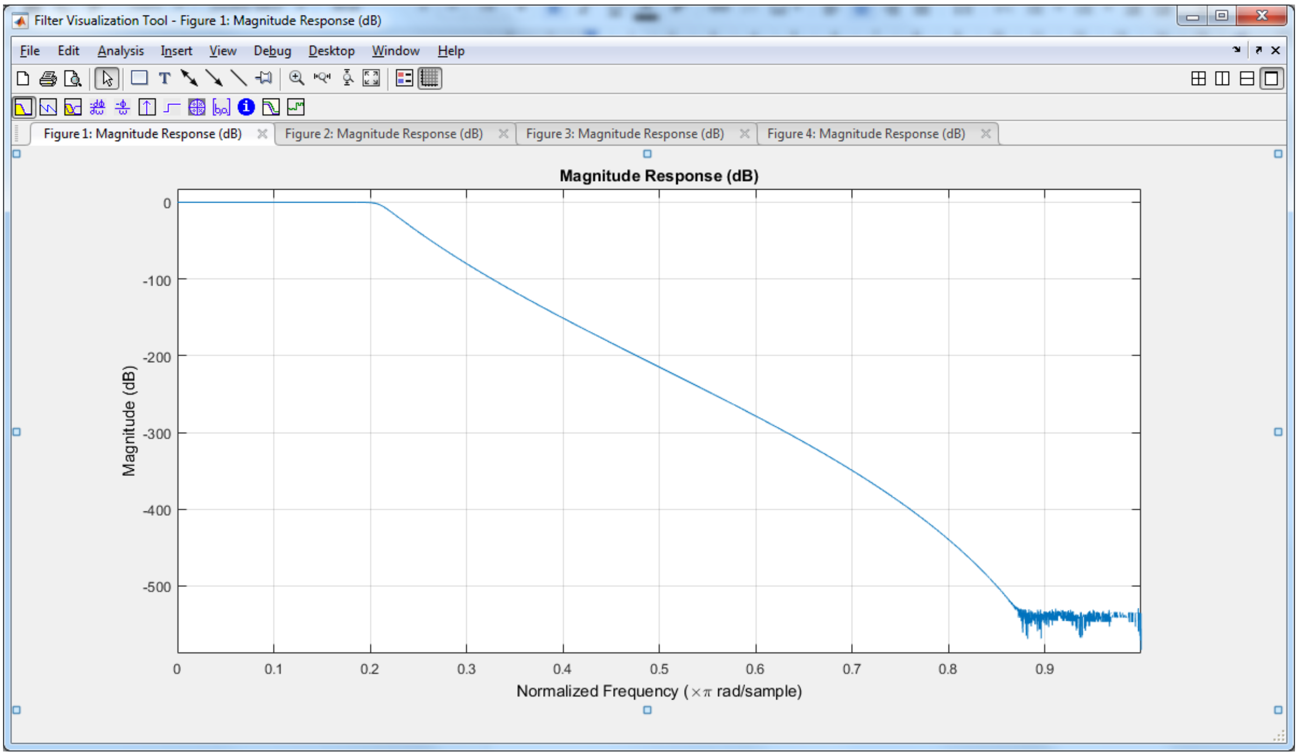


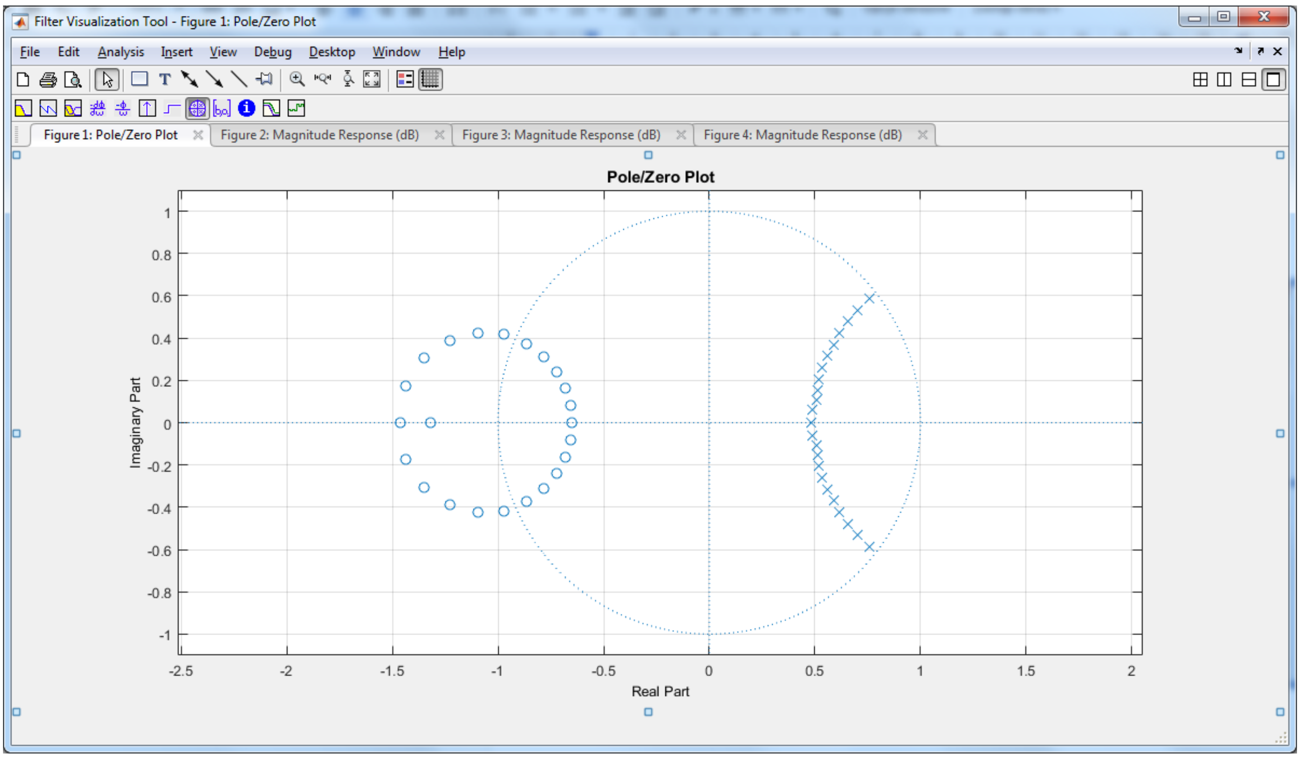
Task 4

IIR filter design by bilinear transformation. Comparison

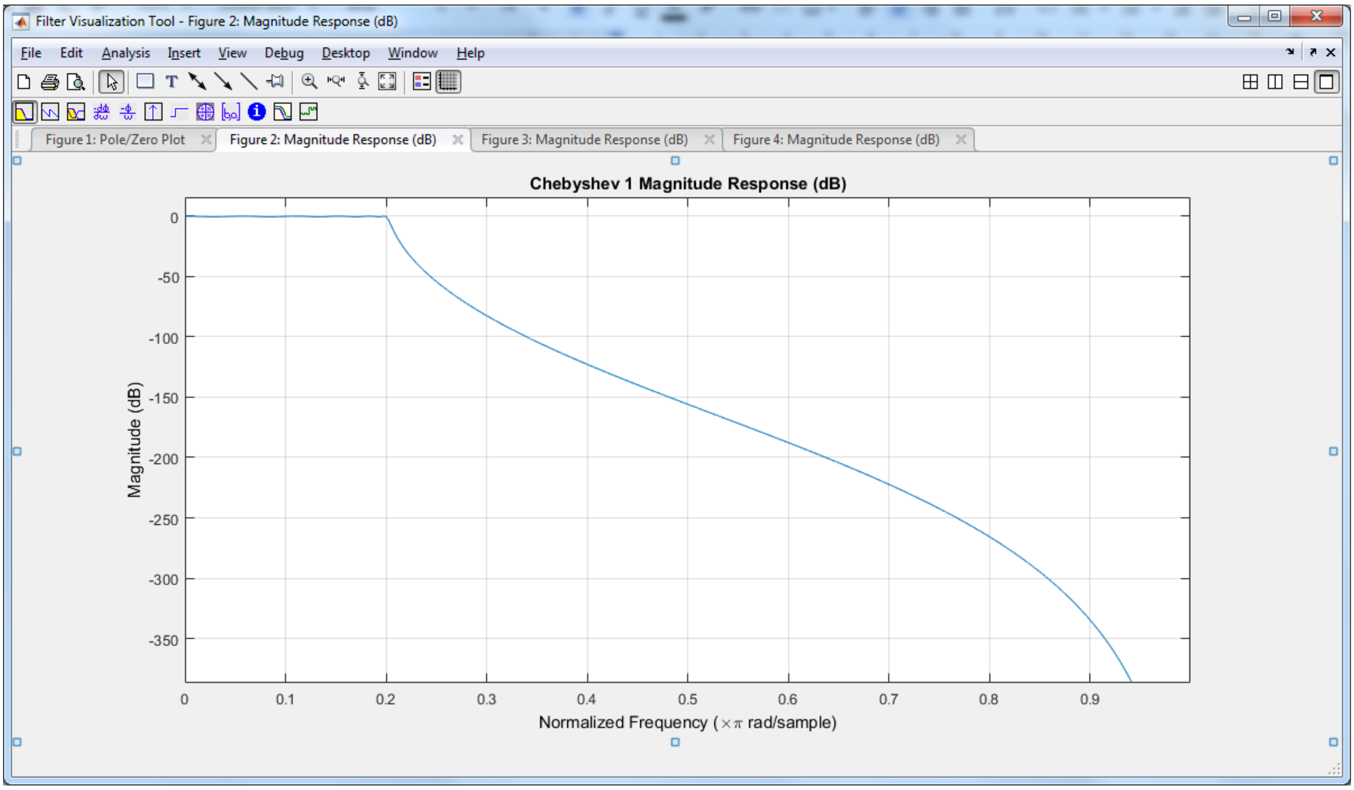
In this task we design an IIR filter with a special method. The given parameters are: passband 0.2pi, stopband from 0.3pi, passband ripple 0.5 dB, stopband 80 dB down.

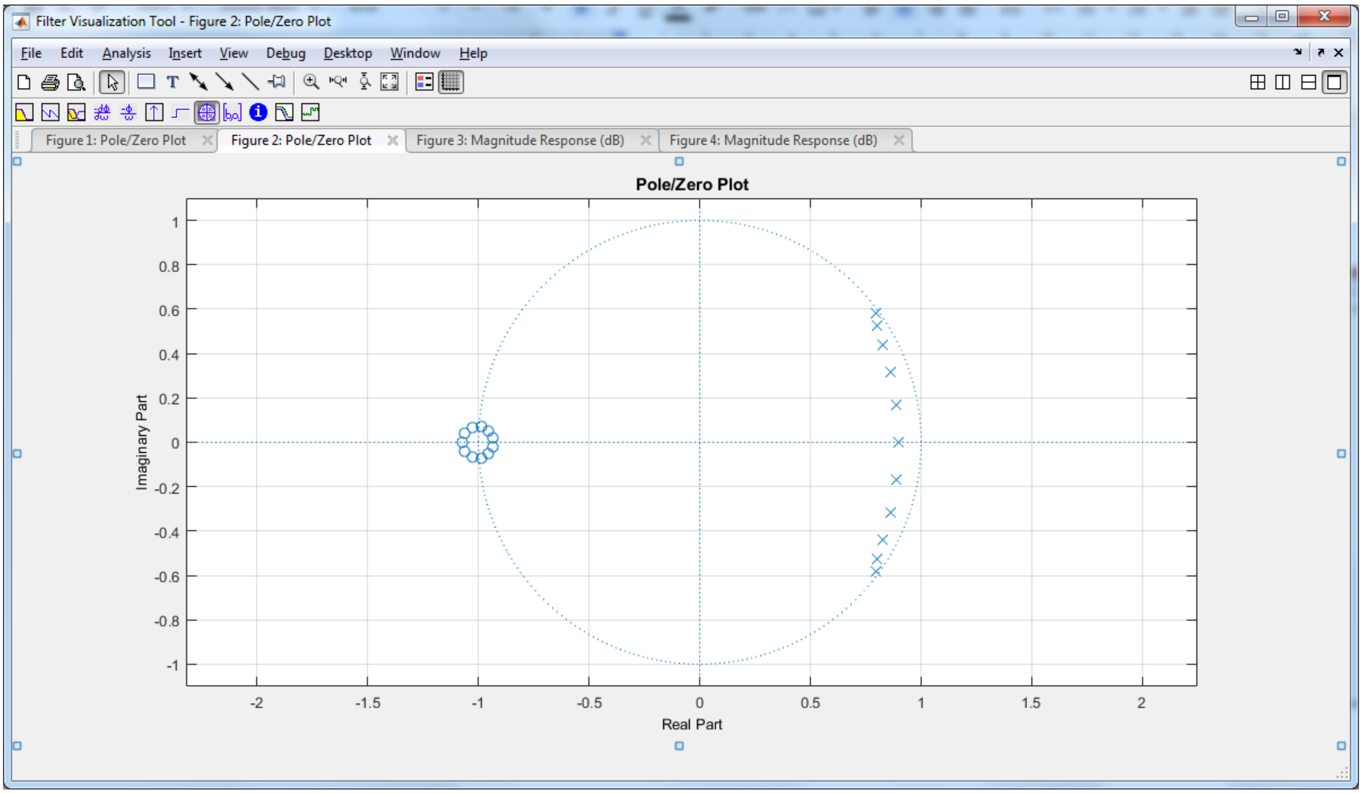
Butterworth. Order needed = 23:



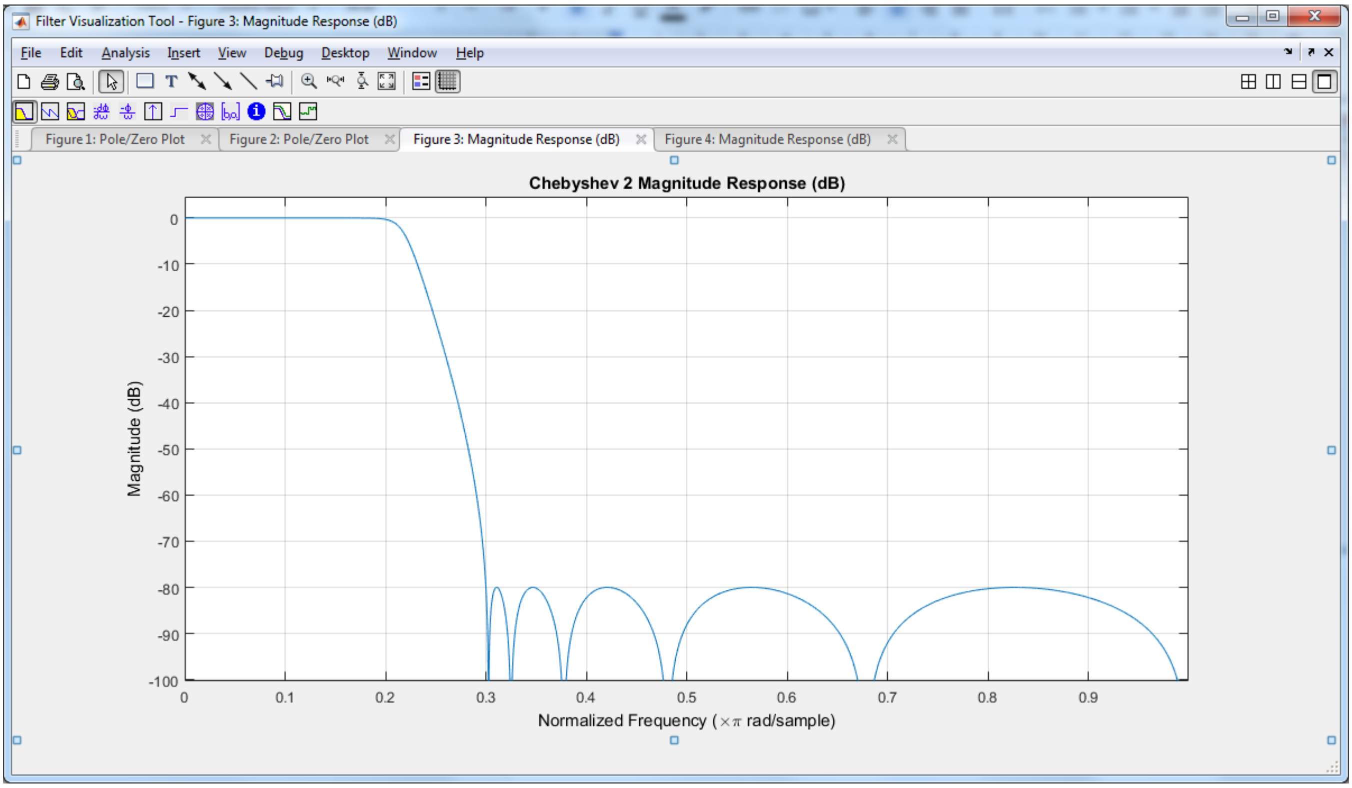


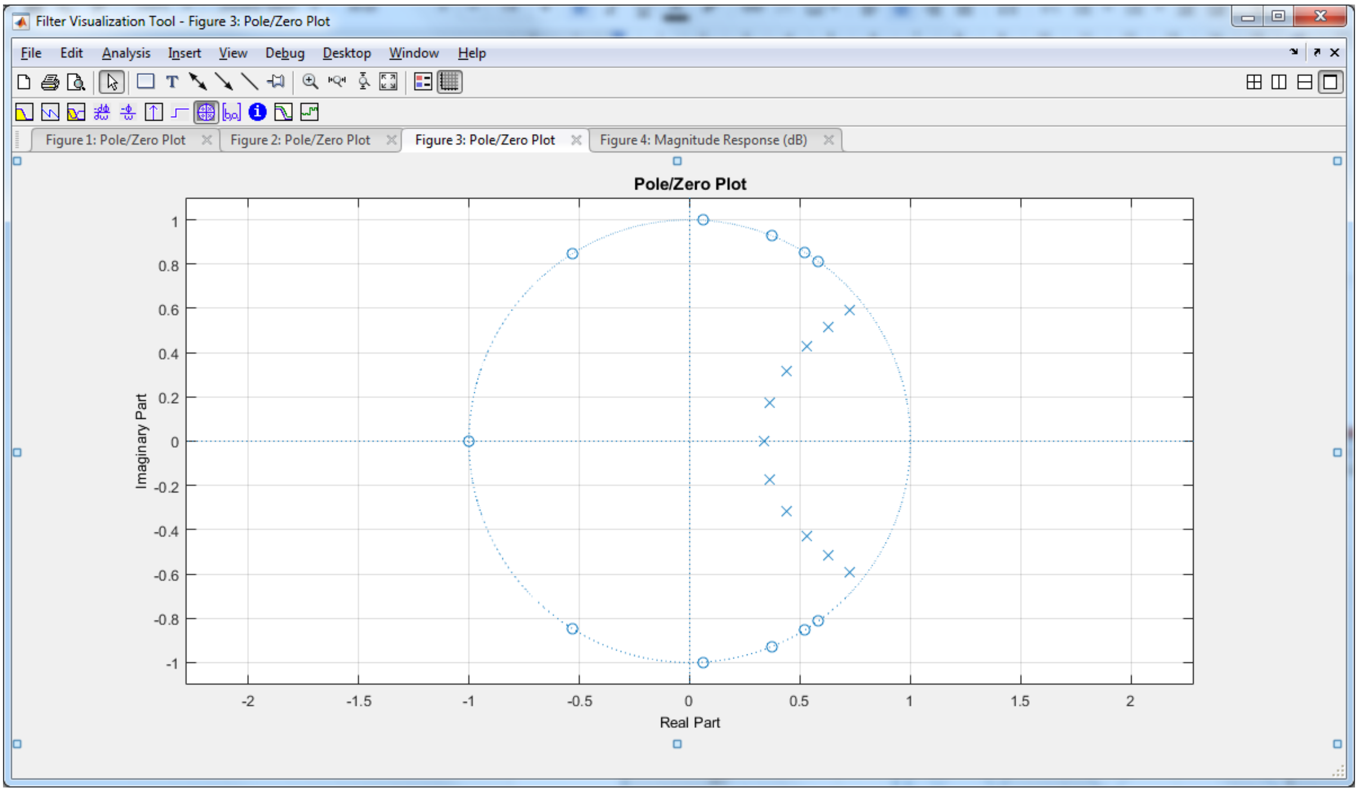
Chebyshev type 1. Order needed = 11:



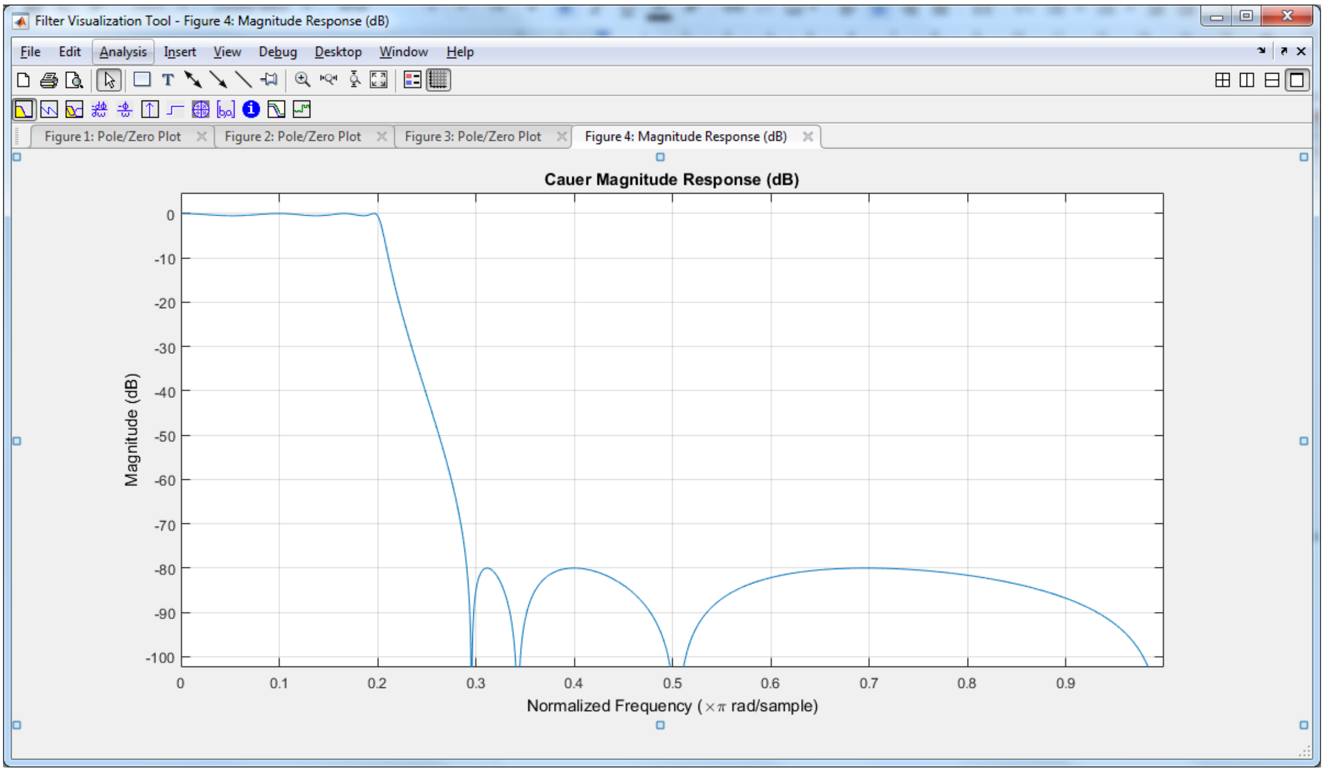


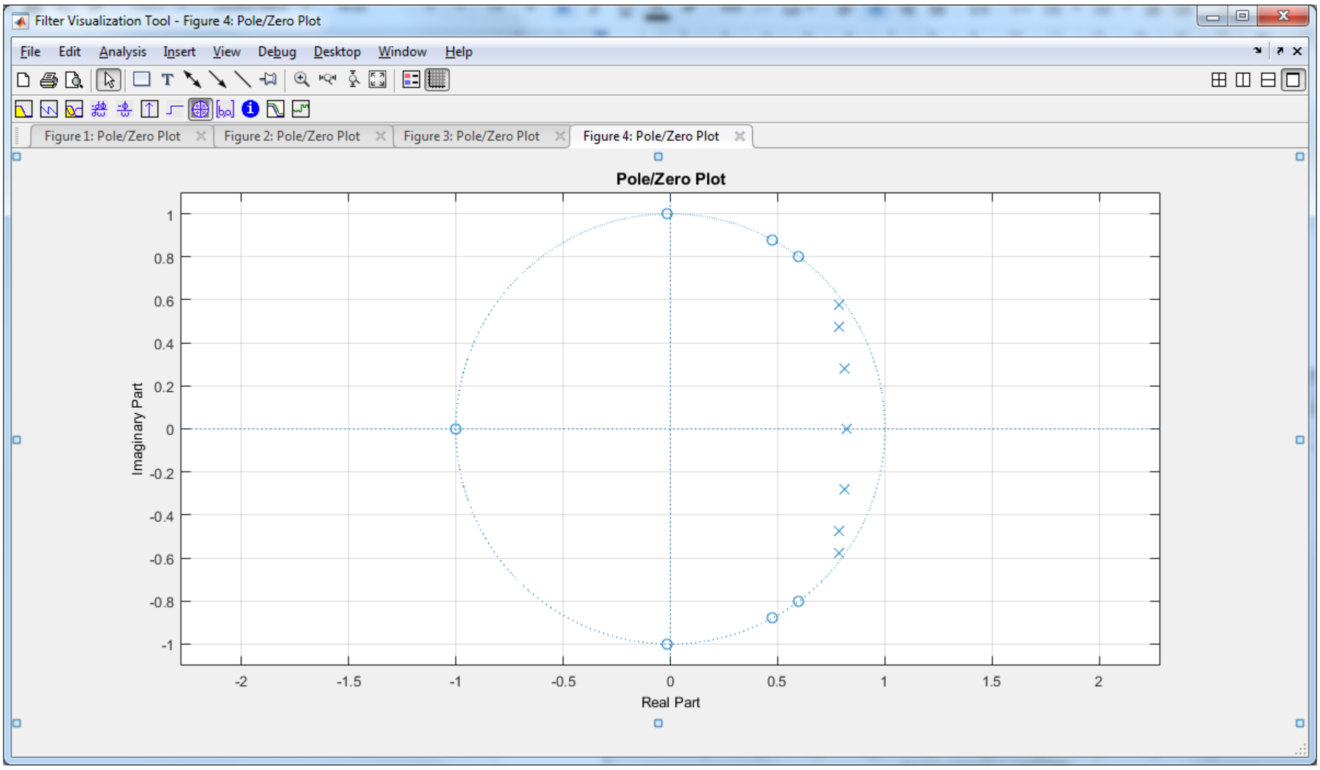
Chebyshev 2. Order needed = 11:





Cauer. Order needed = 7:





For all filters we can see the passband to stop at 0.2 fn. For Chebyshev 2 and Cauer we can additionally notice stopband 80 dB down, stopband starting from 0.3 fn. Passband ripple can be seen well in Cauer filter design, and very slightly for in Chebyshev 1 case. For Chebyshev 1 and Butterworth we see that the freq response goes to very low dB levels. It starts to drop at 0.2 fn like all other filters, but then it looks completely different from Cauer and Chebyshev 2