

Lab

IIR Filter Design

Dr. Aly El-Osery

October 2, 2017

1 Introduction

There are specifications regarding passband, and stopband edges, passband ripple and stopband attenuation when designing filters. There are also several common filter types such as Butterworth, Chebychev and Elliptic filters each having its advantages and disadvantages.

In this lab you will design a high order filter based on the given specification.

2 Lab

2.1 Filter Design

1. On the Figure 1 label
 - passband edge,
 - stopband edge,
 - passband ripple, and
 - stopband attenuation.
2. Design a Butterworth IIR filter with the following specifications:
 - $f_{pass} = 4\text{kHz}$,
 - $A_{pass} = 0.1\text{dB}$,
 - $f_{stop} = 4.5\text{kHz}$,
 - $A_{stop} = 50\text{dB}$, and
 - $f_s = 48\text{kHz}$.

once by having the `butter` function return a transfer function numerator and denominator and a second time by having the function return poles and zeros.

3. What is the order of the filter?
4. Plot the pole-zeros of both approaches.

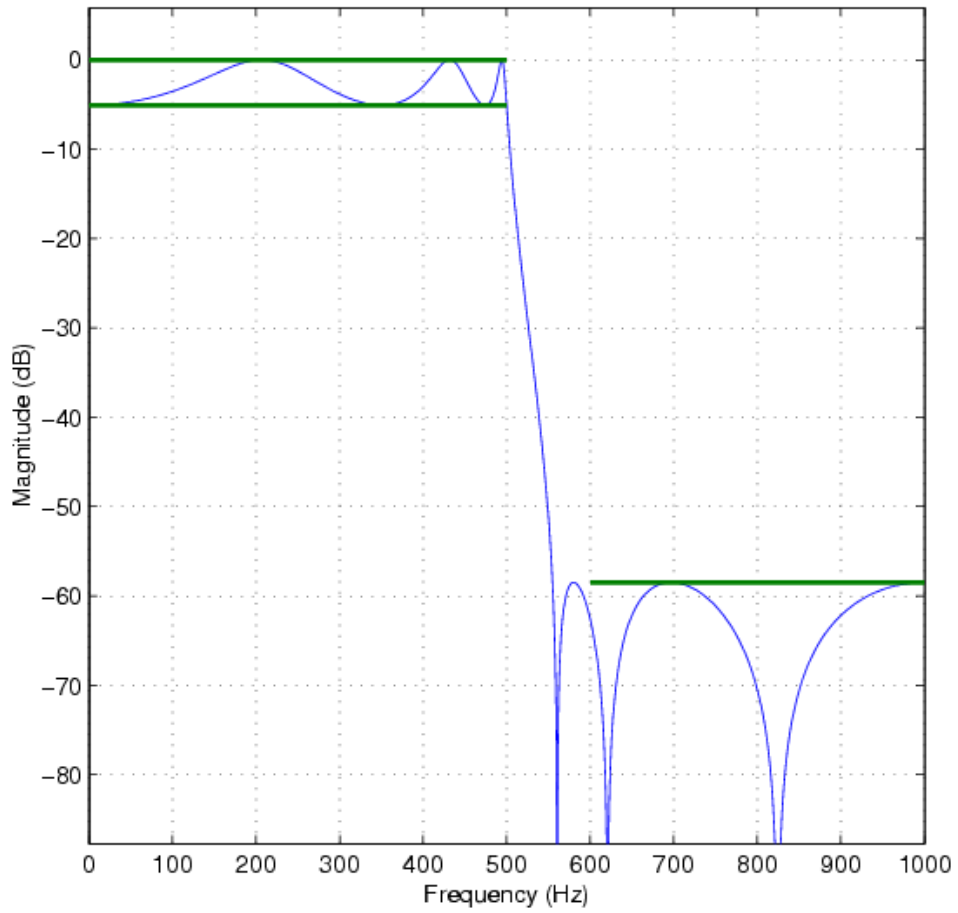


Figure 1: Elliptic filter design specifications

2.2 Filter Implementation

1. Design an elliptic filter with the above specifications listed in Part 1.
2. Write a MATLAB code to store the filter coefficients in a header file to be later used in your c code.
3. Design a generic c-function to implement an N order filter as cascade of second order systems.
4. Implement the filter on the DSP board and verify the magnitude and phase response.

2.3 Fixed-point vs floating-point

Floating-point arithmetic requires more clock cycles than fixed-point. Depending on your application, using fixed-point might be desirable. To design a filter function that uses fixed-point arithmetic, you will need to use `int` (signed 32-bits) or `short` (signed 16-bits) for all your data types. But, the filter coefficients are of type `float` usually between -1 and 1. Therefore, you need to scale your filter coefficients by 2^{n-1} where n is the desired number of bits including a sign bit, then round the numbers to an integer. Further scaling is required if the coefficients are not between -1 and 1.

1. Scale the coefficients to obtain them in 16-bit format and store them.
2. Design a filter that uses fixed-point operation.
3. There are several ways to group the poles and zeros of the filters. Implement the filter twice: once by pairing the poles closest to the unit circle and go inward, and the other by pairing the poles furthest from the unit circle and go outward. Each time pair the poles with the zeros closest to it.