# Elliptic filter - bandpass and bandstop case

In a sequel, we design an elliptic filter for both bandpass and bandstop cases. In this case a ellipband function is used, it has the following arguments: integer filter order, string or integer filter type ("pass"=0, or "stop"=1), both upper and lower passband edges in Hz, passband ripple, stopband attenuation and sampling rate in Hz. We design a bandpass case first with a order of 6, lower passband edge at 600Hz, upper passband edge at 1200Hz, passband ripple at 1 dB, attenuation in stopband equal to 80dB and a sampling frequency at 3000Hz.

Ellippass := ellipband(6 , "pass" , 600 , 1200 , 1 , 80 , 3000) Filter design

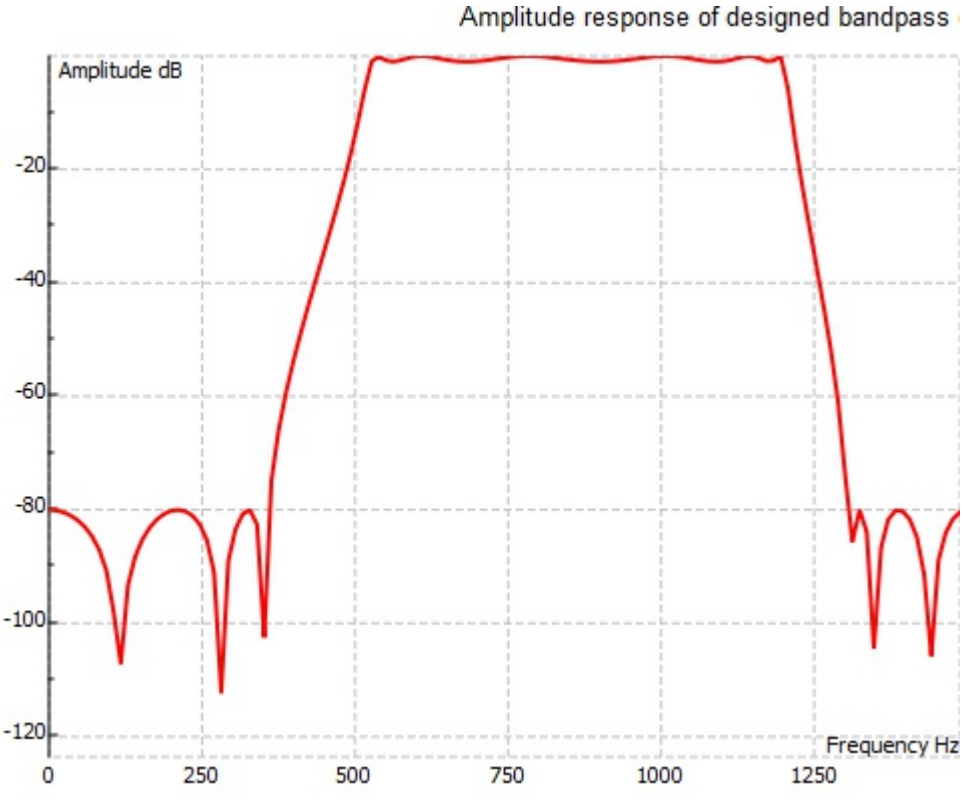
A1 := col2vec(Ellippass , 1) Denominator coefficients

B1 := col2vec(Ellippass , 0) Numerator coefficients

Fpass := iirfreqres(A1 , B1 , 128 , 1) Frequency response of the designed filter

fre := ynodes(z , 0 , 1 - 1/128 , 128) Frequency axis

Fpassg := join mat cols(1500 fre , 20 log10(fabs(Fpass))) Graph of the bandpass filter




## Bandstop case

Next, we design a bandstop filter of order 6, lower passband edge at 600Hz, upper passband edge at 1200Hz, passband ripple at 1dB, attenuation in stopband equal to 80dB and sampling frequency at 3000Hz.

Ellipstop := ellipband(6 , "stop" , 600 , 1200 , 1 , 80 , 3000)  Filter design

A2 := col2vec(Ellipstop , 1)

B2 := col2vec(Ellipstop , 0)

Fstop := iirfreqres(A2 , B2 , 128 , 1)

Fstopg := join mat cols(1500 fre , 20 log10(fabs(Fstop)))  Amplitude response of the designed elliptic bandstop filter