**RTDSP LAB 4 REPORT**

**Real-time Implementation of FIR Filters**

**HAO DING, JIABO ZHOU**

**Content**

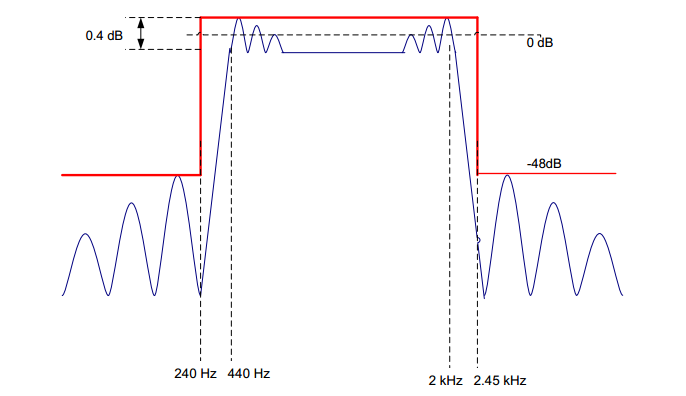
MATLAB filter design

Non-circular FIR filter

Circular FIR design

Frequency response analysis

**MATLAB filter design**

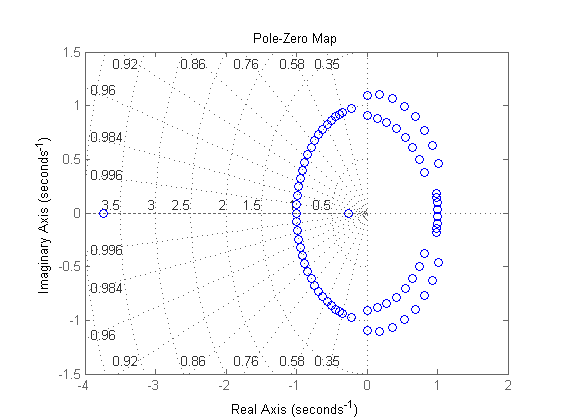
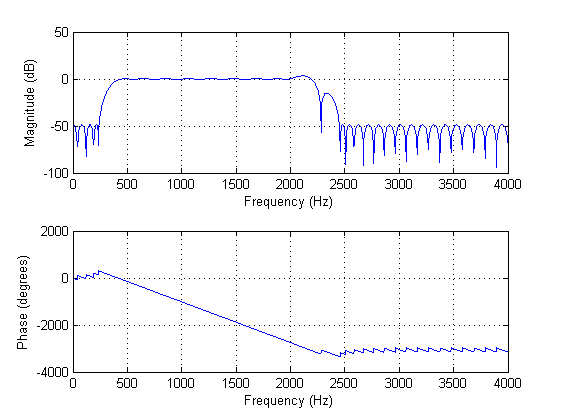
****The FIR filter required to build in MATLAB has specifications shown in Figure 1.

**Figure 1.** FIR specifications

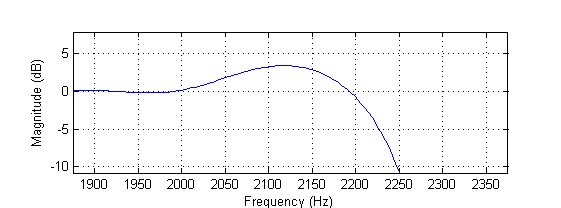
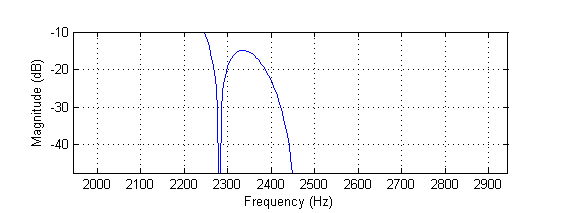
In MATLAB, the Parks-McClelland algorithm is suggested to approximate required filters using *firpmord* and *firpm* functions. Code shown as the following:

**Figure 2.** MATLAB filter design code

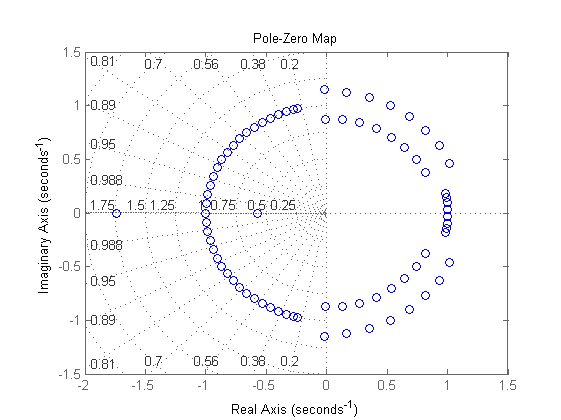
|  |
| --- |
| clear all;    f1=240;f2=440;f3=2000;f4=2450;% frequency boundary  f=[f1,f2,f3,f4];%make array of frequencies    rp=0.4;% ripple in dB  sa=48;% minimum stop band attenuation in dB  dev1=(10^(rp/20)-1)/(10^(rp/20)+1);%calculate pass band deviation  dev2=10^(-sa/20);%calculate stop band deviation  dev=[dev2,dev1,dev2];%make array of deviations    Fs=8000;%specify sampling rate  a=[0,1,0];%specify amplitude    [N,Fo,Ao,W] = firpmord(f,a,dev,Fs);%function to approximate filter  coefs = firpm(N,Fo,Ao,W);%function to calculate frequency coefficients.    freqz(coefs,1,1024,8000)%Plot frequency and phase response  H = tf(coefs,1);%derive transfer function  figure;  pzmap(H);%plot pole and zero map of filter  grid on;    %The following code store filter coefficients in format readable for c.  fileID = fopen('fir\_coef.txt','w');  fprintf(fileID,'double coefs[]={');  for i = 1:length(coefs)  fprintf(fileID,'%f,',coefs(1,i));  end  fprintf(fileID,'};');  fclose(fileID); |

Frequency and phase response as well as pole map are included in Figure 3, from which we can see that specifications are described are satisfied well. However, at the end of pass band and middle of transition band exist two bumps, which may bring in potential risk, shown in Figure 4.

**Figure 3.** (Left)Frequency and phase response. (Right) Pole and zero map

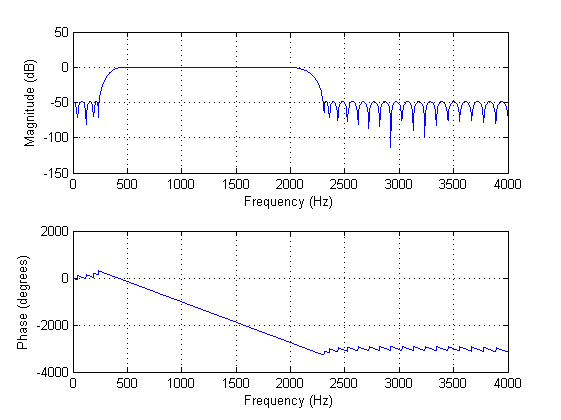


**Figure 4.** (Left)Bump at the end of pass band. (Right) Bump in the middle of transition band

Accordingly, we slightly narrow the transition band range by change the boundary from 2.45 kHz to 2.3 kHz. In z-domain, it is equivalent to a small shift of zeroes. Frequency response and pole map of modified filter is shown in Figure 5, with pole shift marked. At position **A**, poles are closer to each other, which avoids a single zero to pull down amplitude too early so that the bump in middle of transition band is removed. At position **B**, a pole pair is moving to opposite direction, which reduces gain at that frequency and removes the bump at the end of pass band.

**A**

**B**



**Figure 5.** (Left)Frequency and phase response. (Right) Pole and zero map