Digital Filters design for ECG and EEG noise signals

Yang Du 46884803

In this assignment, I design series of filters to eliminate noises of ECG and EEG signals by using IIR BLT, FIR windowing and FIR Parks-McClelland method respectively in Matlab.

Before designing the filters, we need to know what type of noises in the given signals and the frequencies they lie in and the criterion. In order to calculate and estimate the noise, I use pwelch() function in Matlab to calculate the power spectral density(PSD) for each signal. It is common that noises will appear on peaks in PSD waveform. And from the general background of ECG and EEG signal analysis, we understand that the noises are consist of several noise which are from different sources such as transmission line, electrode contact noise, 60Hz noise,and some low frequency noise that the frequency is around 1 Hz.Therefore we need design a highpass filter and set the corner frequency to 1Hz to eliminate those noises. In addition, the EEG and ECG instrumentation both have 60 Hz in their signal, So we need to design a notch filter to eliminate 60 Hz noise. Moreover, EEG signals consist by different waves including θ wave, α wave, β wave (biomedical instrumentation slides) and they normally lie in frequency below 30Hz but above 5Hz, hence in this case we need a band-pass or a low filter to maintain all those waves and remove signals not from the frequency band.

Analyzing these types of noise in EEG and ECG signal, we can obtain our design methods and steps in Matlab code. Obviously, our design criteria and steps are: calculate PSD, make low-pass, high-pass and band-pass filter by using different methods.

The instruction shows that I should use 3 methods of filtering, Therefore we use butter(), fir1(), and firpm() to replace the 3 method in Matlab.

ECG signal analysis and filter design method

My design criteria are:

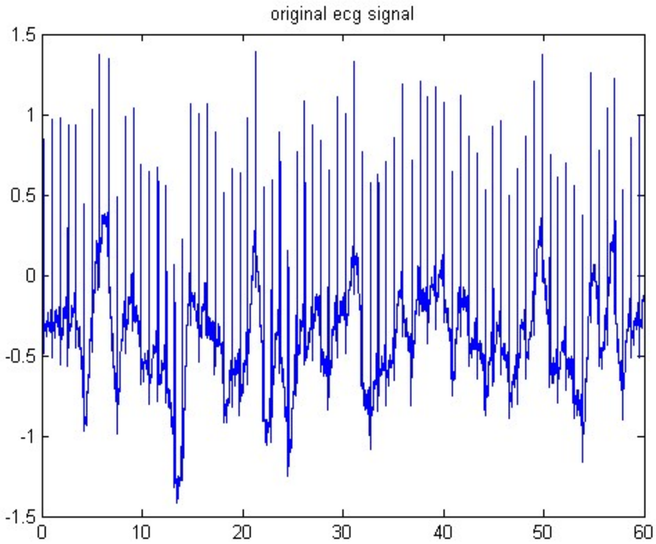
(1) Calculate PSD and use notch filter to eliminate 60 Hz ,120Hz and other frequency noise

(2) Design high pass filter which cutoff frequency is 5Hz.

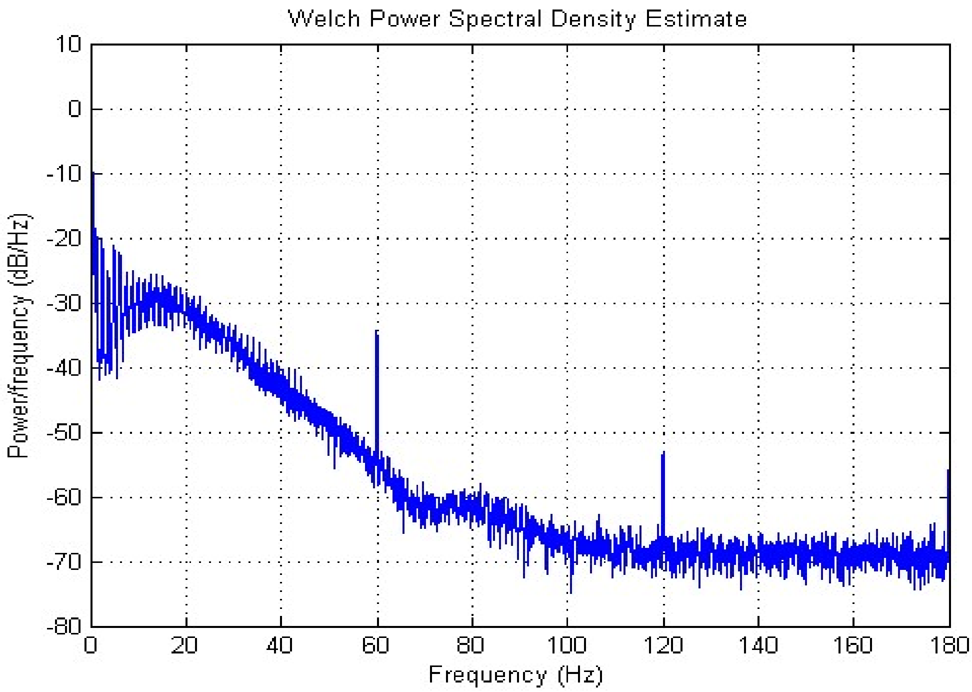
(3) use the three methods to refine noise signal

1. First look of ECG Frequency Noise

(1)Loading the data of original ECG noise signal, since the data has two columns to show sampling time and signal power respectively, I extract the two columns as t and y to show the original ECG signal waveform.



(2) I set the sampling frequency at 360 samples/sec, so Fs=360Hz. First of all, we use pwelch() function to calculate signal power which is important to filter design.

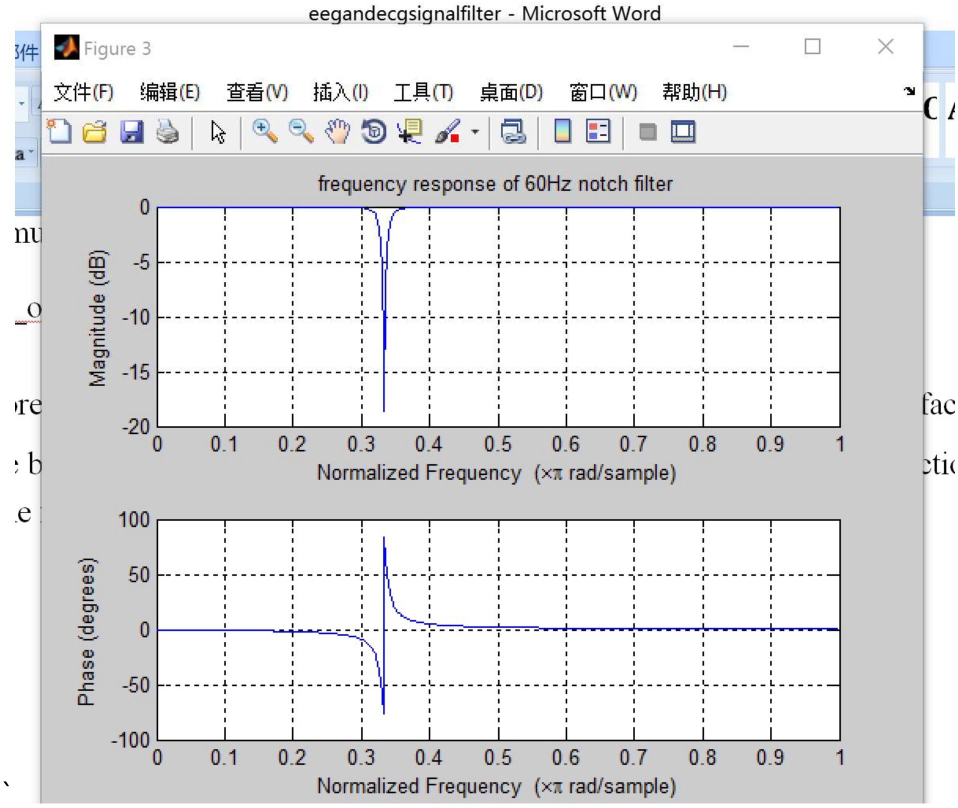


As the waveform above, we can observe that the signal power is higher in 60 Hz, 120Hz and 180Hz. So, next step, we should design a notch filter to eliminate these abnormal signal.

3). Use notch filter function iirnotch() to remove 60Hz noise in the ECG signal. Here is a formula that is

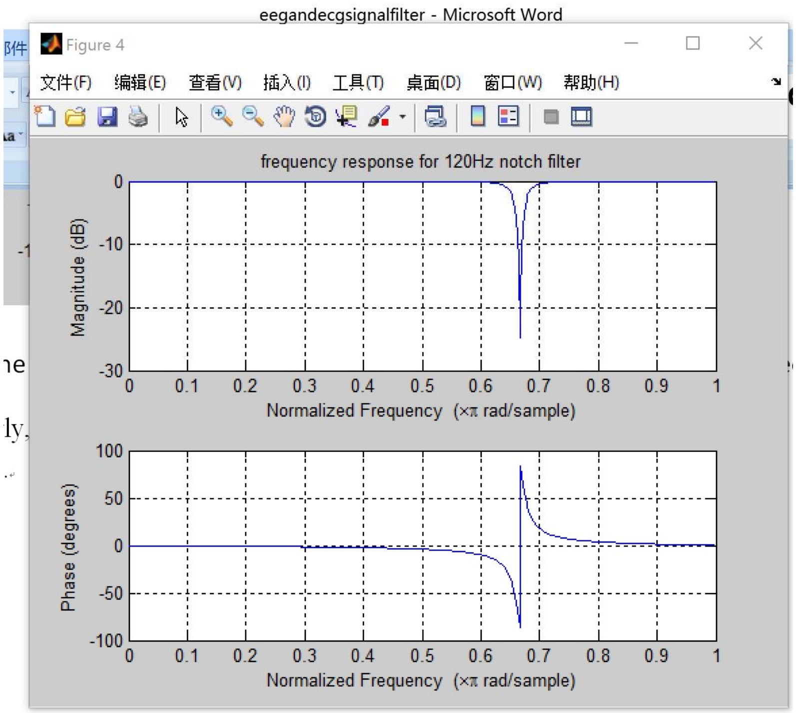
Omega_o = cut-off freq/(sampling freq/2)

Therefore, $w_0 = 60/(Fs/2) = 1/3$, we choose the filter to 30 orders and then 30 as Q factor, then the bandwidth at -3dB amplitude is Bw=Wo/Q. Then apply iirnotch() function to show the frequency response of the notch filter.



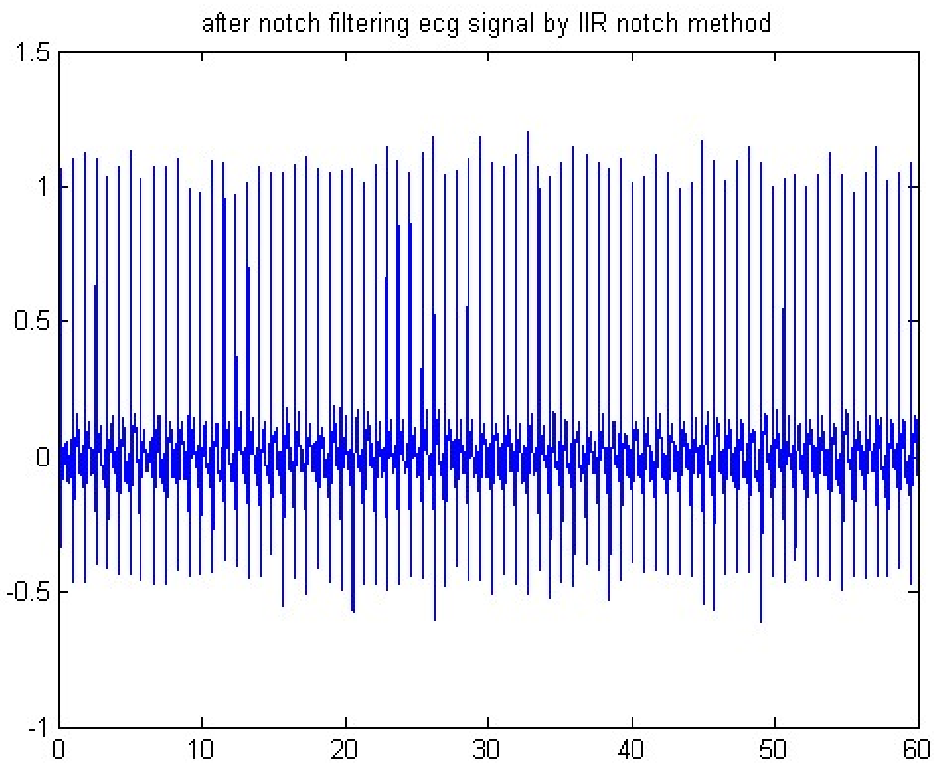
From the waveforms, we can observe that the amplitude in 60 Hz has been reduced.

Similarly, we use the same method to design a notch filter to eliminate 120 Hz and 180 Hz.

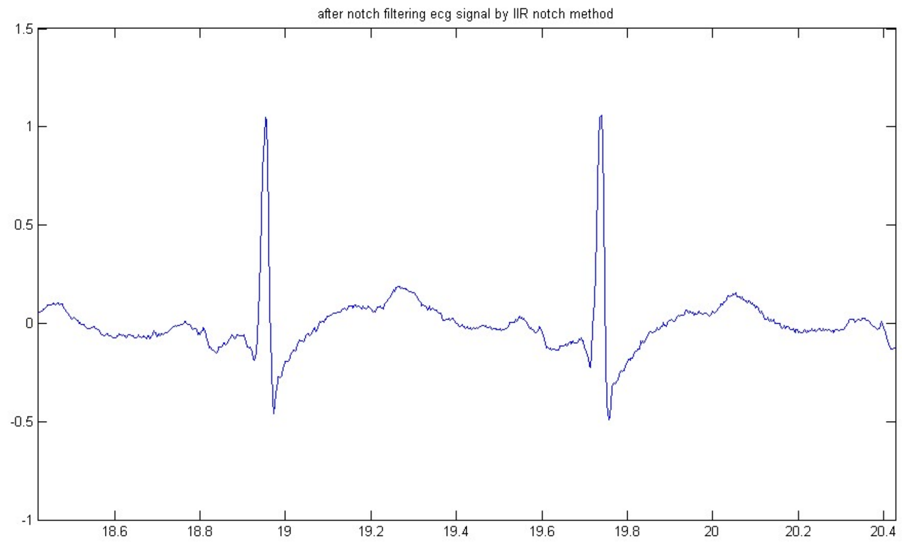


However, when I am going to design a notch filter to eliminate 180 Hz, Matlab shows the error. So, I just design notch filter for 60 Hz noise and 120 Hz noise.

(4) Let us see the signal waveform after notch filtering.



We can enlarge the resolution of x axis, and we can see the signal clearly.

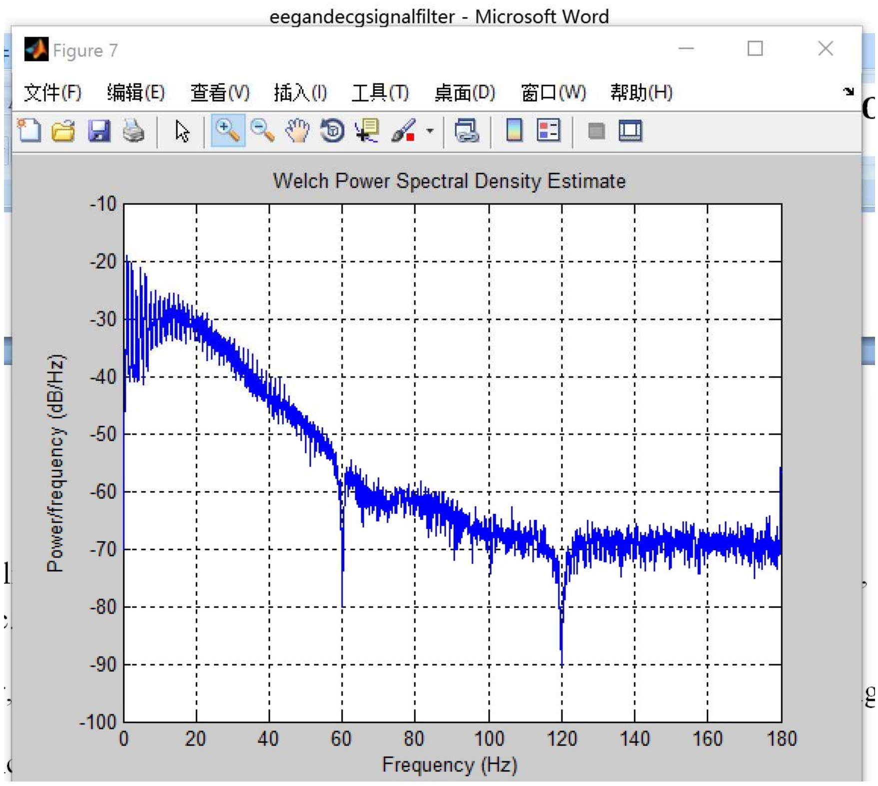


It seems like the ECG signal shows in instrumentation. We can see the P wave, R wave, etc.

However, the signal still has some noise, we will process the signal in following steps

.

(5) Recalculate PSD and analyze the waveform below



It's obviously that the power at 60Hz and 120Hz is reduced extremly, therefore the result indicates that the notch filter design meets the requirement we need.

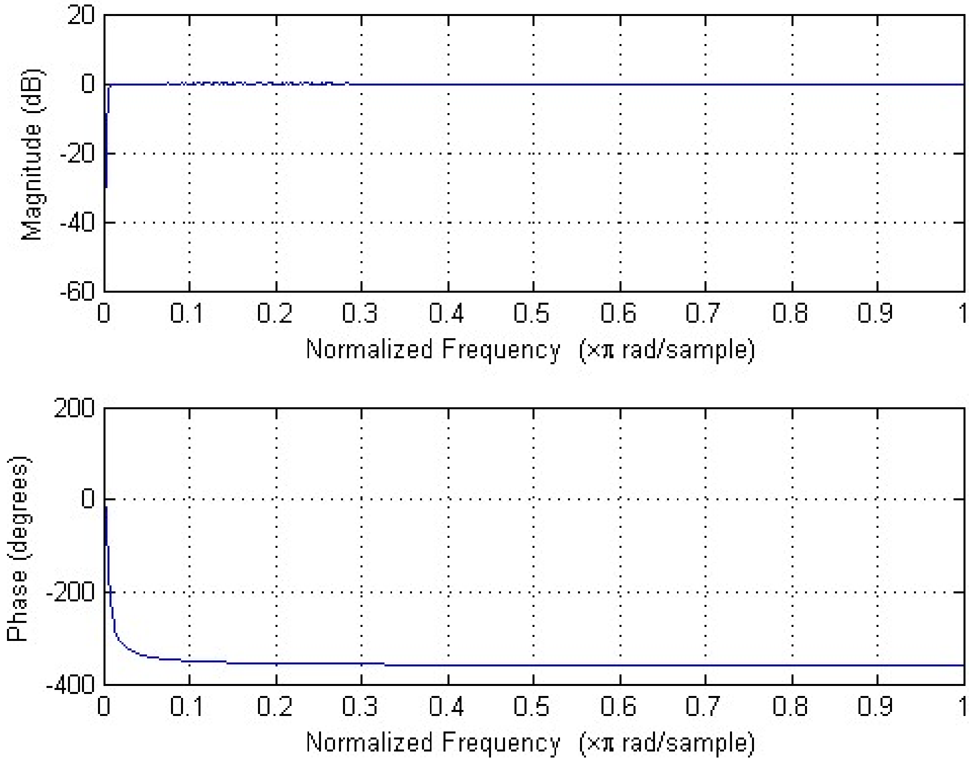
2. Eliminate other frequency noise below 5Hz and around 1Hz.

(1) IIR LBT Based High-pass Filter

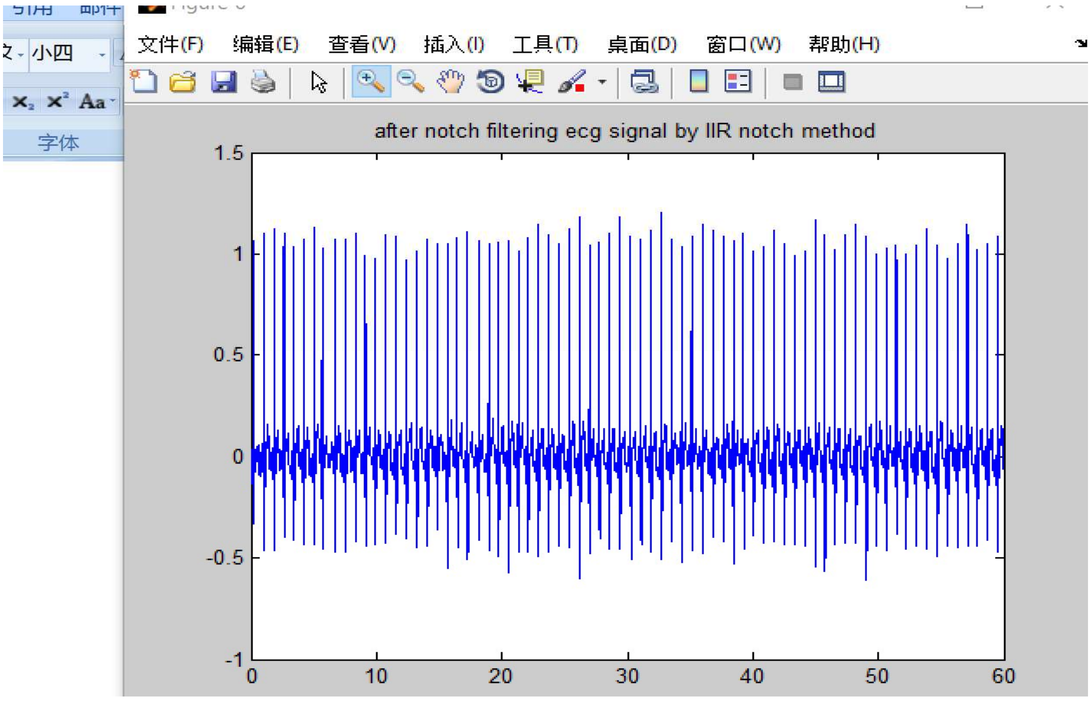
First of all, we need follow the criteria when we design the filter, we choose butter() function that means Butterworth analog filter by using BLT method.

In butter() function, we should set the order N and cutoff frequency Wcut-off. For this high-pass filter, the Wcut-off equals to 5Hz/(Fs/2),and then we can plot the frequency response of the BLT high-pass filter.

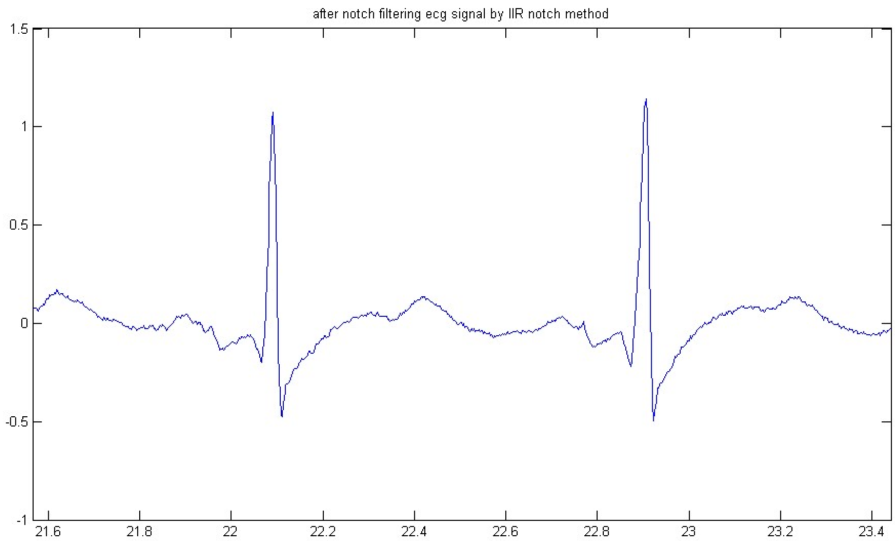
The frequency response is:



And then we use filter() function that we can get the coefficient from butter(N, Wn) to form the new filtered waveform. By analyzing the new signal waveform, we can find that the ECG signal is refined by the Butterworth filter.

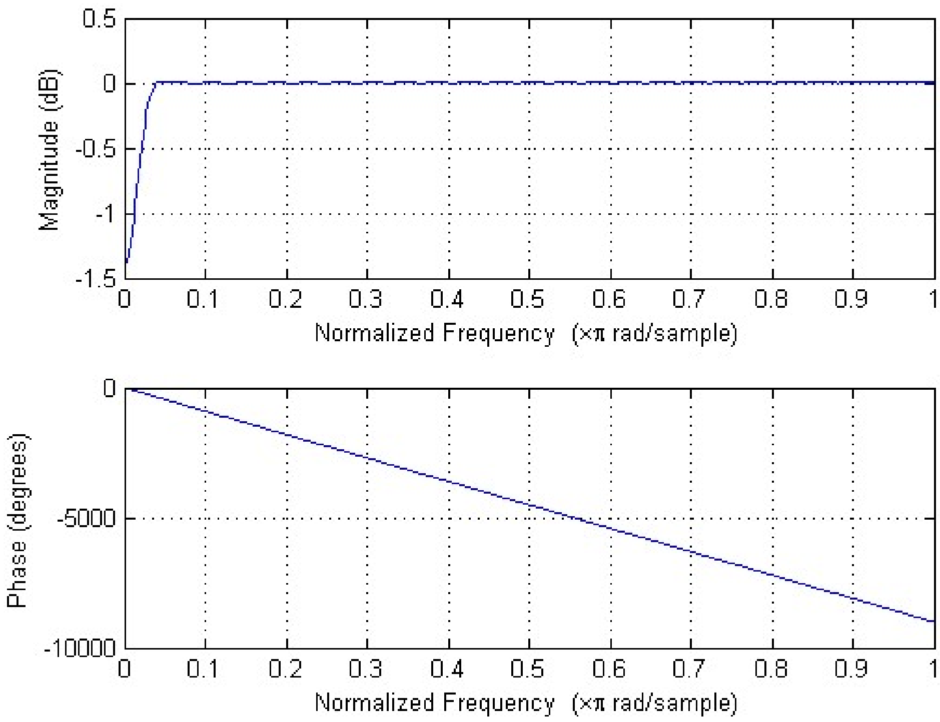


Zooming the waveform, It becomes more better than the waveform is just filtered by notch filter.



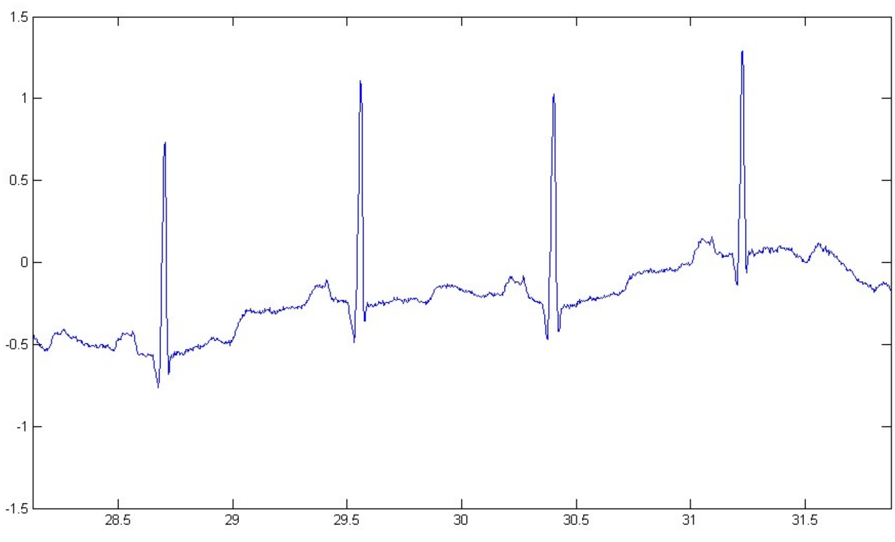
(2)FIR windowing method filter

To design this fir windowing filter, we use fir1() functions that is b = fir1(n,Wn,'ftype'), the cutoff frequency is 1Hz and mapped to normalized frequency Omega_o, we have frequency response shows below:

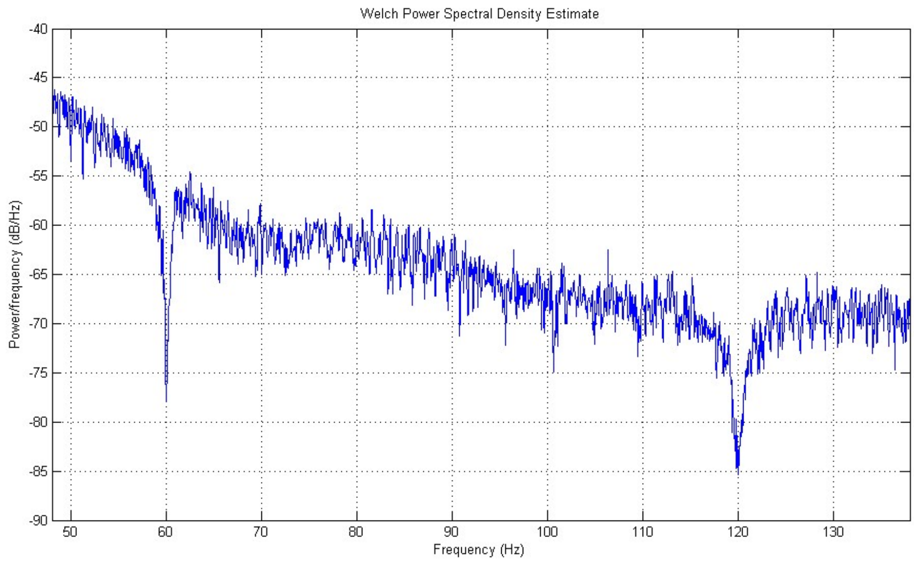


Because of FIR filter, the phase response declines linear and very smooth.

Apply the coefficients returned from fir1() function, we have the denoise ECG signal:



Recalculate PSD, we can get:

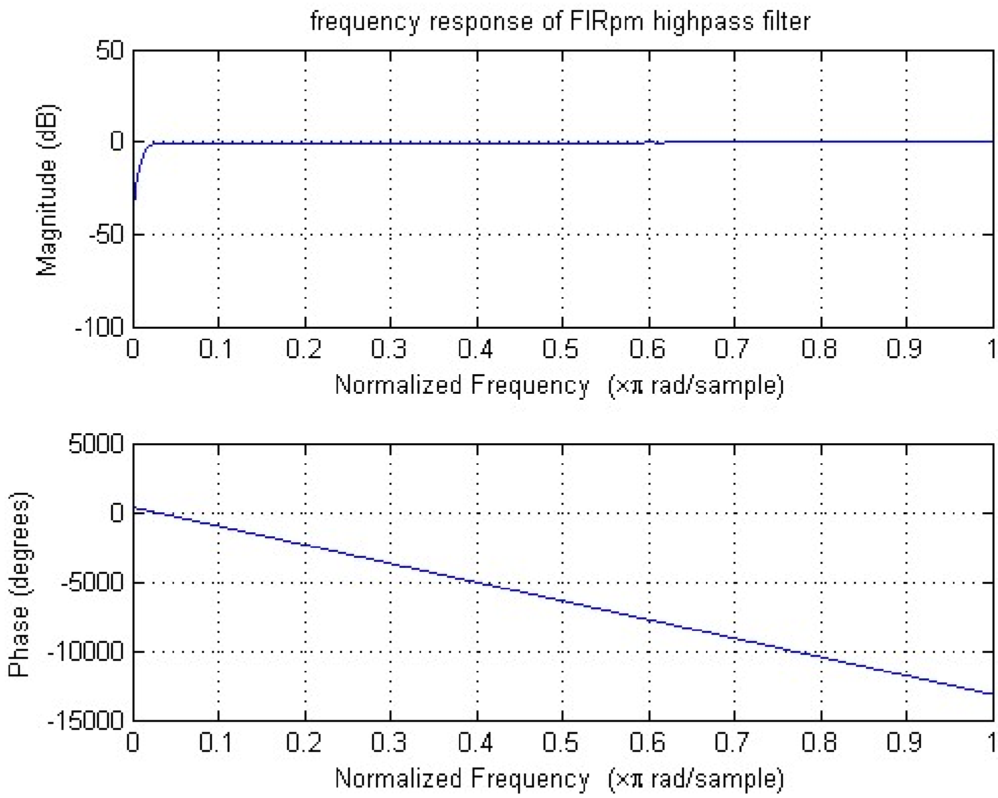


120Hz noise is reduced more lower than 60Hz noise.

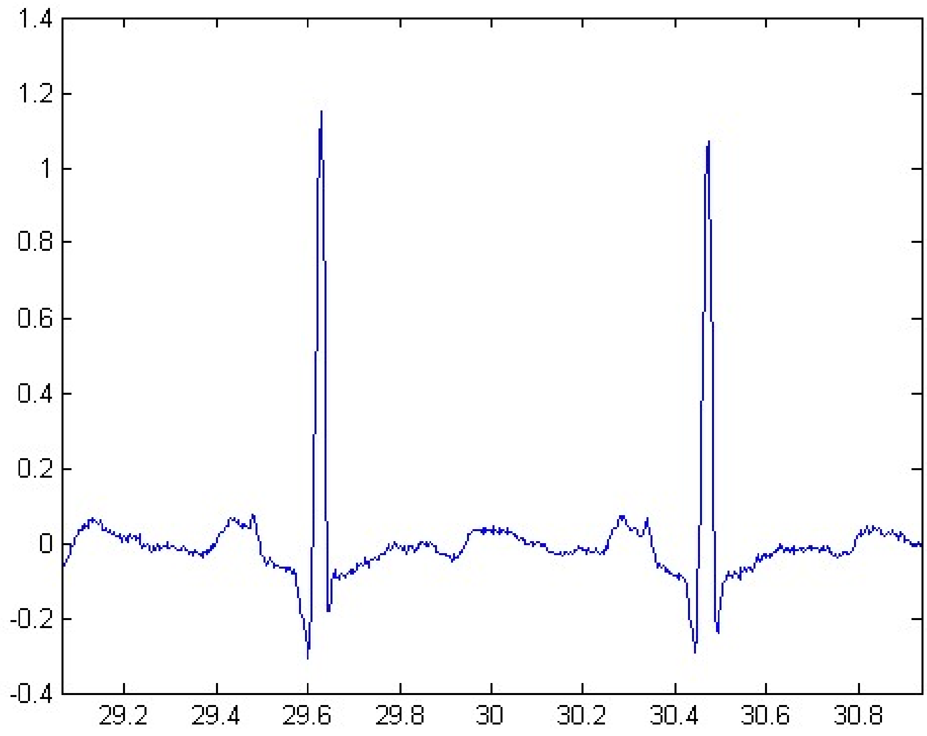
(3) FIR Parks-McClelland method

We use firpm() function to achieve the FIR filter method, The pattern of the function is b = firpm(n,f,a). And I set n=150.

The frequency response is:



The refined signal waveform is:



EEG signal analysis and filter design method

My design criteria are :

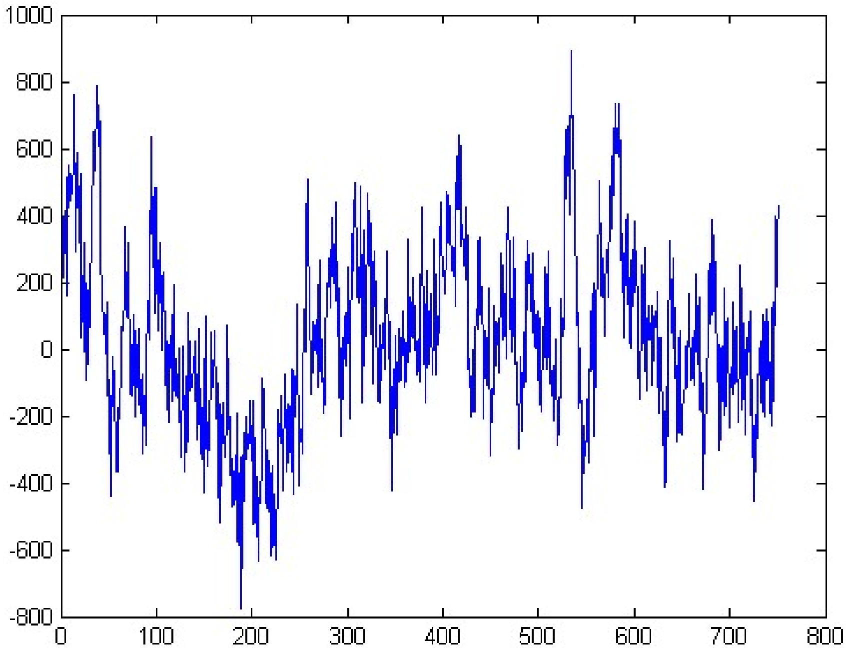
(1) Calculate PSD waveform and design notch filter to eliminate 60 Hz.

(2) Design A band pass filter which cutoff frequency is 0.5Hz and 60Hz.

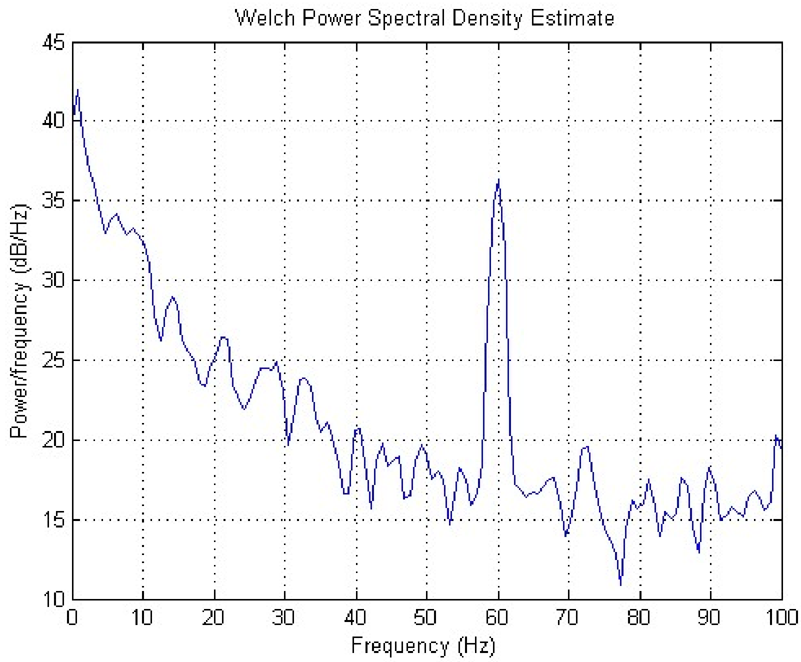
(3) Use the three methods to refine noise signal.

1. First look of ECG Frequency Noise

I set sampling rate is 200. The original EEG signal figure shows below:

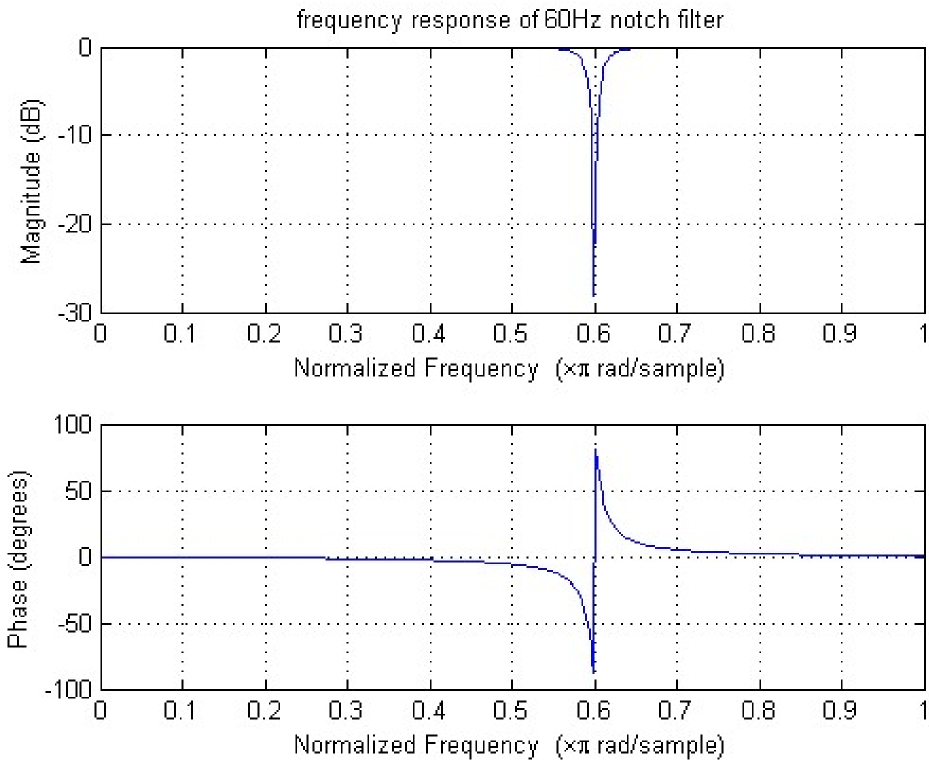


The PSD waveform is:

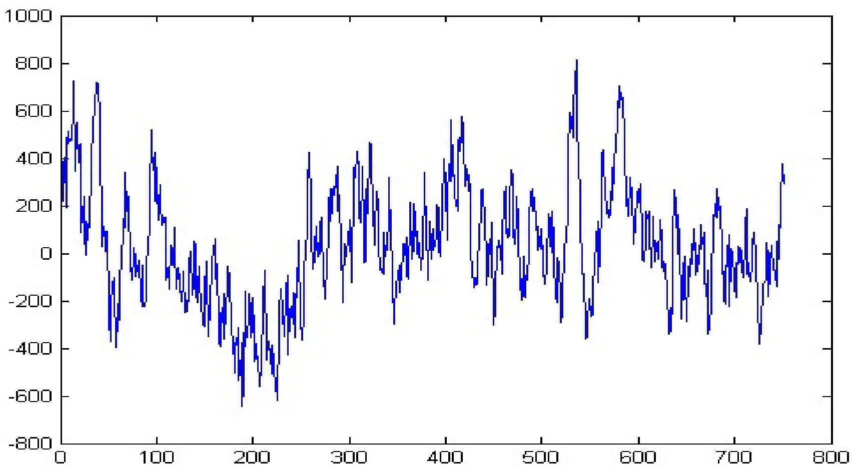


According to the waveform above, the signal get more energy at 60Hz, so we need a notch filter to eliminate it.

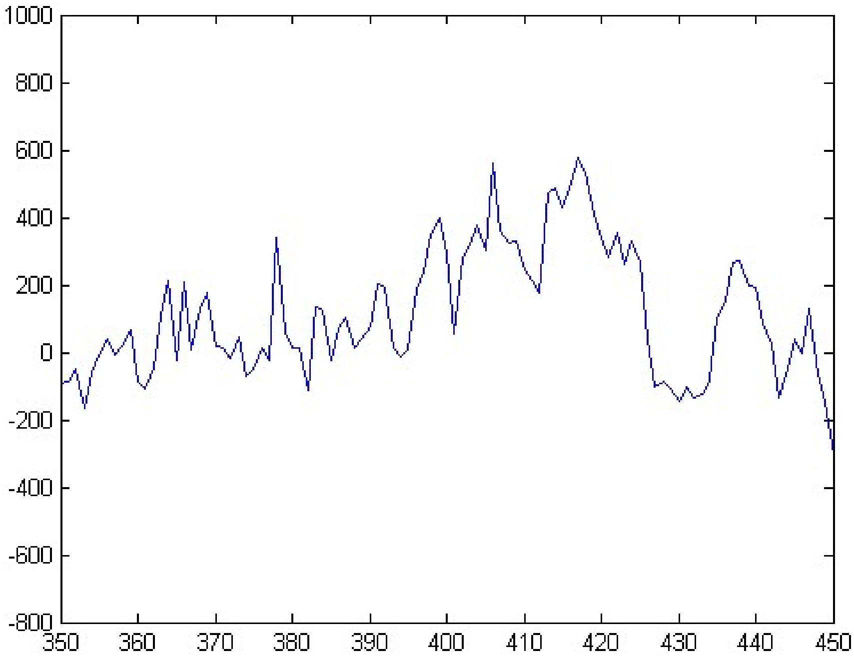
Applying iirnotch() function to eliminate the 60Hz noise, I set Q= 30 and the frequency response shows below:



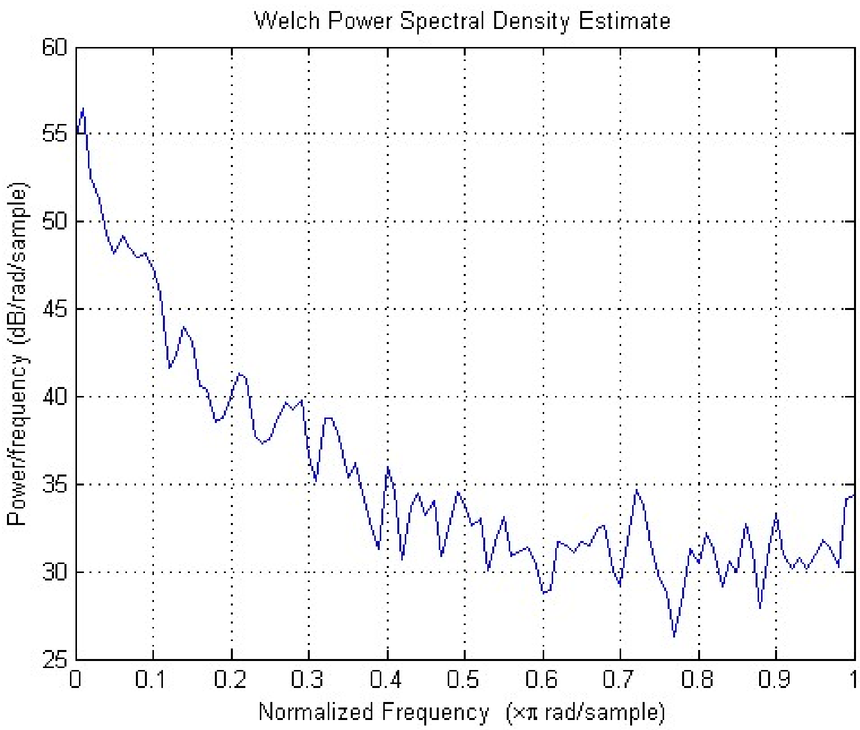
The waveform after using 60Hz notch filter



Zooming the diagram, we can get:



The PSD waveform after using notch filter is that

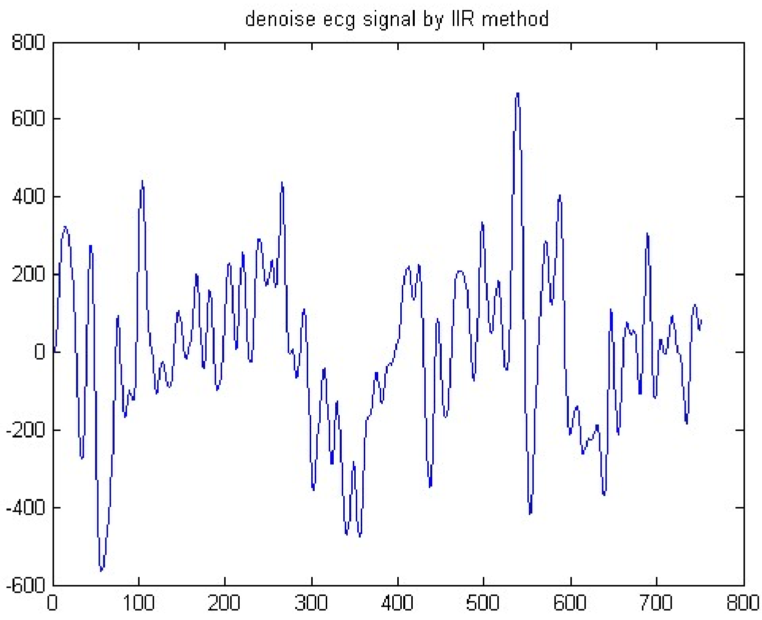


2. Eliminate other frequency noise by using different methods

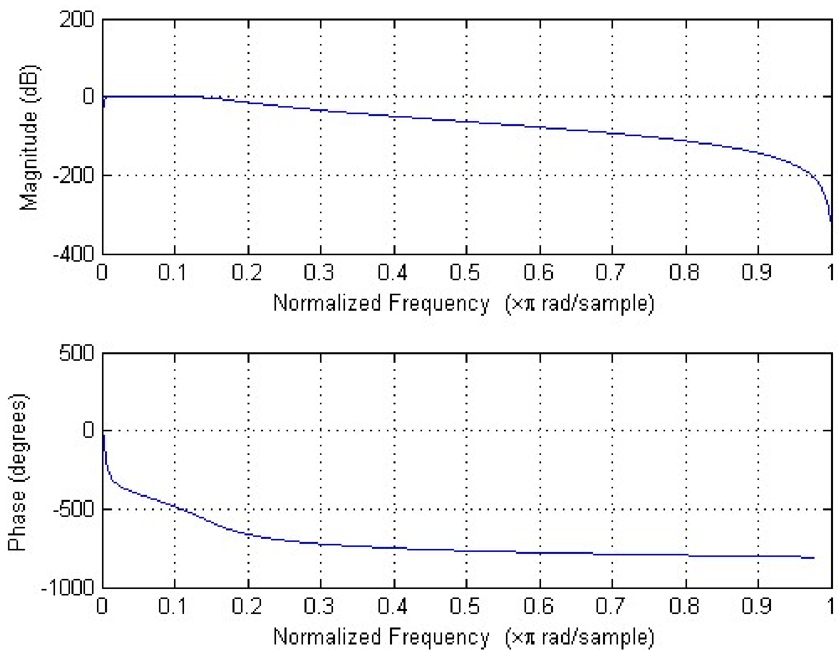
Basically, there are 4 types of EEG signal and they focus on the frequency band is 0.5 Hz to 14 Hz. So we set the frequency band is 0.5Hz to 60Hz.

(1) Band-pass Filter based on IIR BLT method

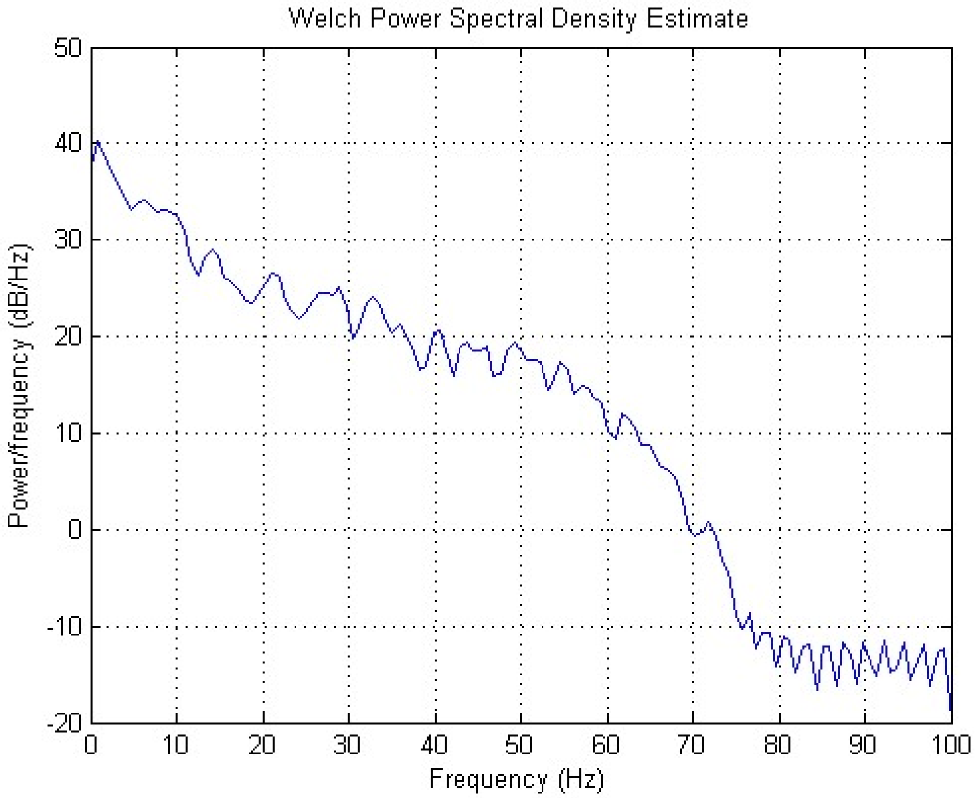
For this band-pass filter we using butter () function to achieve BLT filtering. The frequency band is from 0.5Hz to 60Hz, and I set order N as 5. The waveform is:



The frequency response of the design is:

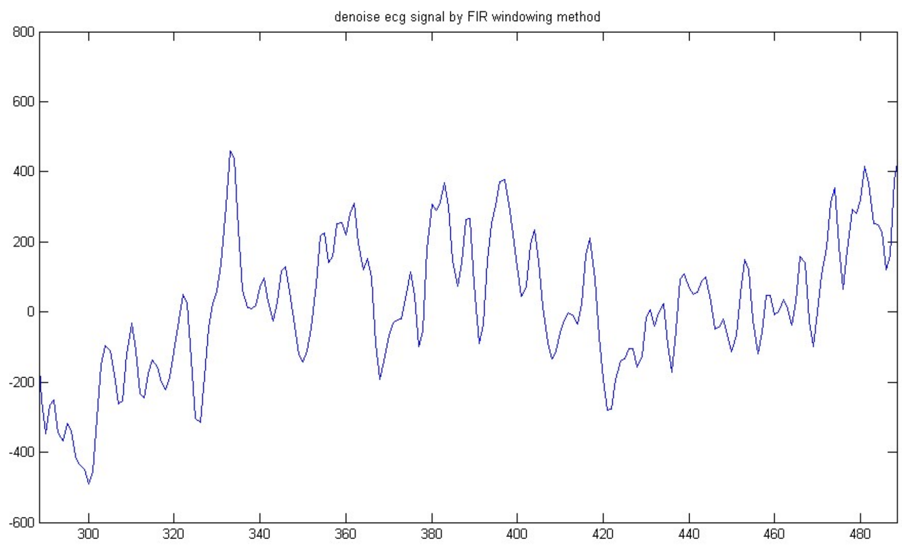


Recalculating PSD, we can see that the 60 Hz noise is reduced by the notch filter.

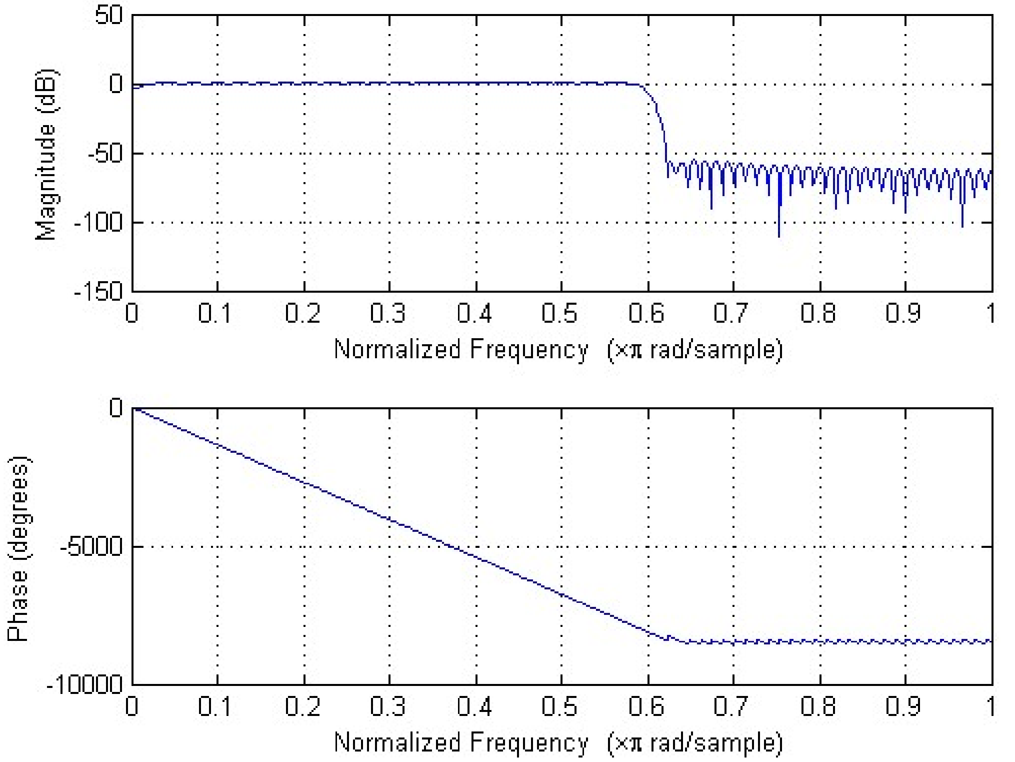


(2) Band-pass Filter Based on FIR windowing method

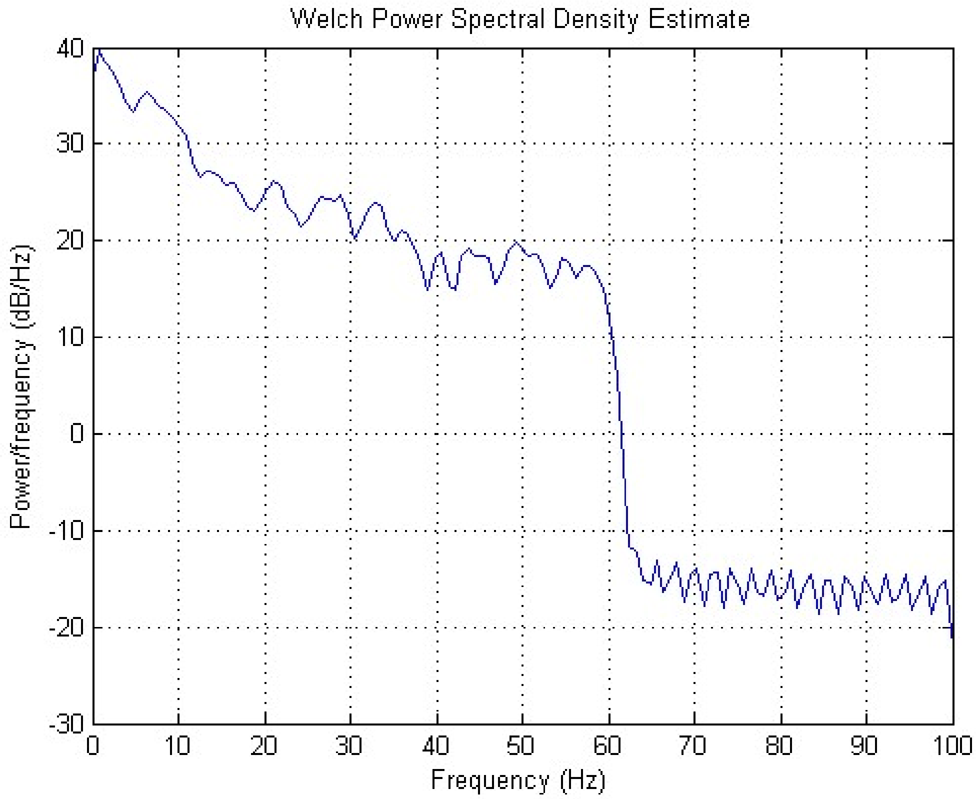
by using fir1() function in for this band-pass filter, I set windowing length as 150, the refined EEG signal is:



And the frequency response is that



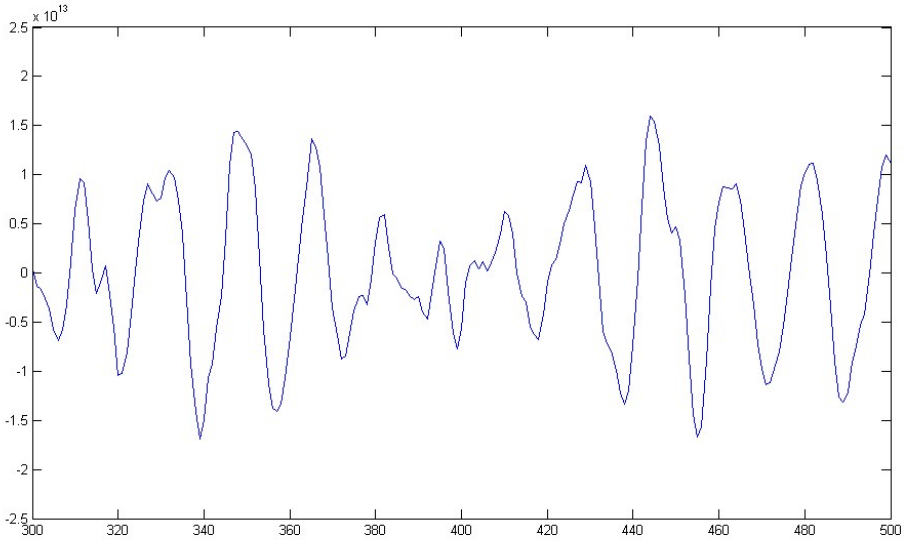
Recalculating PSD waveform is:



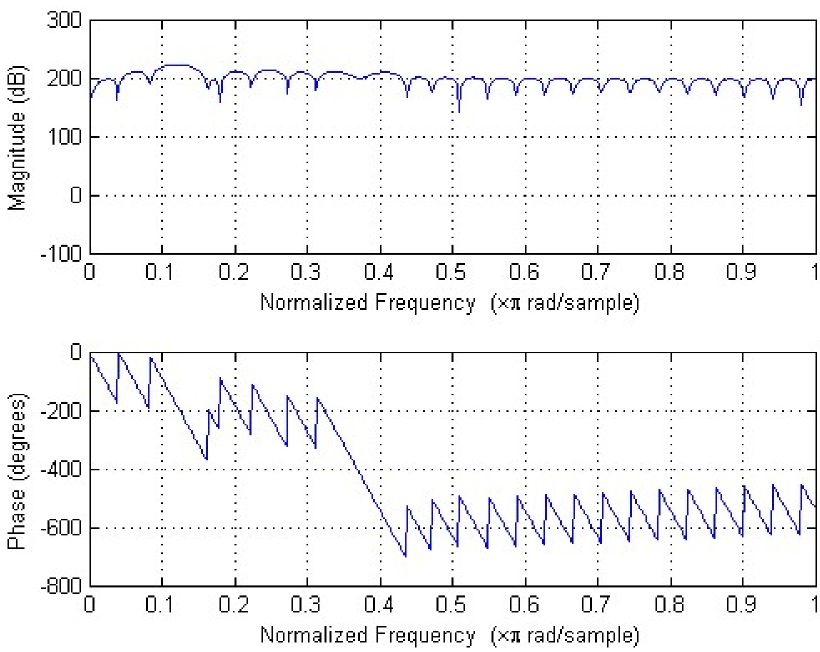
(3) band-pass filter by using FIR Parks-McClelland method

We use firpm() function to achieve FIR Parks-McClelland filtering method . Then I set f1=[0,w1,w2,1],a1=[0,1,1,0], and make the orders of filter is 50.

The refined waveform is



And the frequency response is



Summary

Analyzing the performance of different filters above, firstly, IIR method will have steep tradeoff around the cutoff frequencies, while the phase response appears to have offset. Secondly, cutoff frequencies of FIR method based filter present smoother amplitude compare with IIR filtering method, but they have more linear phase response. In addition, focus on the order for IIR filter and FIR filter, FIR filter usually need much higher order than IIR filter. It seems that IIR filter has better performance than FIR filters in high-pass filter, while in the band-pass filter design, the three methods show the similar result.

Code I used

ECG signal processing code

```
clear all;
load('ecg_plus_noise (1).mat');

M=ecg;
Fs=360;%sample frequency=360
%%%%%%plot original ecg signal%%%
t=M(:,1); %column 1
y=M(:,2); %column 2 power of the ecg signal with noise
figure(1)
plot(t,y)
title('original ecg signal')

%%%%%%caculate the power spectral density (PSD) estimation%%%%
figure(2)
pwelch(y,[],[],[],Fs)

%%%%% notch filter eliminate 60Hz noise %%%%%
Omega_o = 60/(Fs/2)
BW = Omega_o/30;
[b,a] = iirnotch(Omega_o,BW);
figure(3)
freqz(b,a)
title('frequency response of 60Hz notch filter')

%%%%%% plot signal after notch filter %%%%
figure(4)
y1 = filter(b,a,y)
```

```
%%%%%% notch filter eliminate 120Hz noise %%%%%
Omega_o2 = 120/(Fs/2)
BW2 = Omega_o2/30;
[d,c] = iirnotch(Omega_o2,BW2);
figure(4)
freqz(d,c)
title('frequency response for 120Hz notch filter')

%%%%% plot ecg signal after eliminate 120Hz noise %%%%%
y2 = filter(d,c,y1)

%%%%%%   IIR BLT method Highpass filter %%%%%
Omega_o=5/(Fs/2)
[B A]=butter(5,Omega_o,'high')
figure(5)
freqz(B,A)
y3=filter(B,A,y2)
figure(6)
plot(t,y3)
title('after notch filtering ecg signal by IIR notch method')
%%%%%%% REcalculate the PSD of the signal %%%%%%%
figure(7)
pwelch(y3,[],[],[],Fs)

%%%%%%% FIR windowing   highpass filter %%%%%%
Omega_o=1/(Fs/2)
[B1 A1]=fir1(150,Omega_o,'high')
figure(8)
freqz(B1,A1)
%%%%%%%% plot FIR1 highpass filter %%%%%%%
y4=filter(B1,A1,y2)
figure(9)
plot(t,y4)
%%%%%%% RE-REcalculate the PSD after FIR windowing %%%%%%%
figure(10)
pwelch(y4,[],[],[],Fs)

%%%%%%% fir notch filter design%%%%%%%
Omega_o=5/(Fs/2)
figure(11)
f1=[0,0.001,Omega_o,1];
a1=[0,0.001,0.9,1]
b1=firpm(150,f1,a1);
```

```
freqz(b1,1)
title('frequency response of FIRpm highpass filter')
y5=filter(b1,1,y2)
figure(12)
plot(t,y5)
```

EEG signal processing code

```
clear all;
clc;
load('eeg_plus_noise.mat');
Z=eeg_plus_noise;
Fs=200;%sample frequency= 200

%%%%%%% plot original eeg signal %%%%%%%
figure(1)
plot(Z)

%%%%% caculate the power spectral density estimation(PSD) %%%%
figure(2)
pwelch(Z,[],[],[],Fs)

%%%%%%%%%%%%% notch filter (60Hz)    %%%%%%%%%%%%%%%%%%%

Omega_o = 60/(Fs/2)
BW = Omega_o/30;
[b,a] = iirnotch(Omega_o,BW);
figure(3)
freqz(b,a)
title('frequency response of 60Hz notch filter')

%%%%%% plot the signal after notch filter %%%%%%


Z1=filter(b,a,Z)
figure(4)
plot(Z1)
figure(13)
pwelch(Z1,[],[],Fs)
%%%%%%% IIR BLT bandpass filter %%%%%%%
w1=0.5/(Fs/2)
w2=60/(Fs/2)

wn = [w1,w2]
[B A]=butter(5,wn,'bandpass')
figure(5)
```

```
freqz(B,A)
Z2=filter(B,A,Z1)
figure(6)
plot(Z2)
title('denoise ecg signal by IIR method')

%%%%%% calculate the PSD after IIR BLT method %%%%%%
figure(7)
pwelch(Z2,[],[],[],Fs)

%%%%% FIR windowing   bandpass filter %%%%%%%%
figure(8)
[B1 A1]=fir1(150,wn,'bandpass')
freqz(B1,A1)
Z3=filter(B1,A1,Z1)
figure(9)
plot(Z3)
title('denoise ecg signal by FIR windowing method')

%%%%%% calculate the PSD after FIR windowing %%%%%%
figure(10)
pwelch(Z3,[],[],[],Fs)

%%%%%% FIR Parks-McClelland   bandpass filter %%%%%%%
f=[0,w1,w2,1];
a=[0,1,1,0];
b1=firpm(50,f,a);
figure(11)
freqz(b1,1)
Z4=filter(b1,1,Z1)
figure(12)
plot(Z4)
```