Introduction

Electrocardiograms (ECGs) measure possible cardiac operation, and evaluate the heart's different heart functions including heart rate, heart rhythm etc. This medical system will use electrodes to measure the heart potential from the surface of the skin. Knowledge can be used to assist the method of diagnosis as the precise position of those anomalies may be determined. ECG signals display the numerous heart-enduring acts during contraction graphically. These graphs display a sign of five different peaks, which correspond to different heart functions. In this lab ECG will be analyzed. During the acquisition of ECG signal picks up noise and in the focus of this lab is to design filters that remove power line interference noise, high noise frequency, and base-line artifact.

Background

High-frequency noise, low-frequency noise (wandering base-line), and power-line frequency artifacts are all common contaminants in ECG data. A low-pass filter may be used to remove high-frequency noise, while a high-pass filter can be used to remove low-frequency noise. However, if the cut-off frequencies of the two filters are not chosen correctly, the ECG signal may be distorted, resulting in excessive smoothing or broadening of the QRS complex, as well as distortion of the usually isoelectric PQ and ST segments [1]. Notch filter is usually used to filter 60hz power line interference. Below is the transfer function of notch filter in z domain.

$$H(z) = \frac{(z - z_1)(z - z_2)}{z^2} = (1 - z_1 z^{-1})(1 - z_2 z^{-1})$$

The necessary zero's angular location on the unit circle is provided as:

$$\Theta_0 = 2\pi \frac{\text{notch filer frequency} \, f_0}{\text{sampling frequency} \, f_s}$$

Result

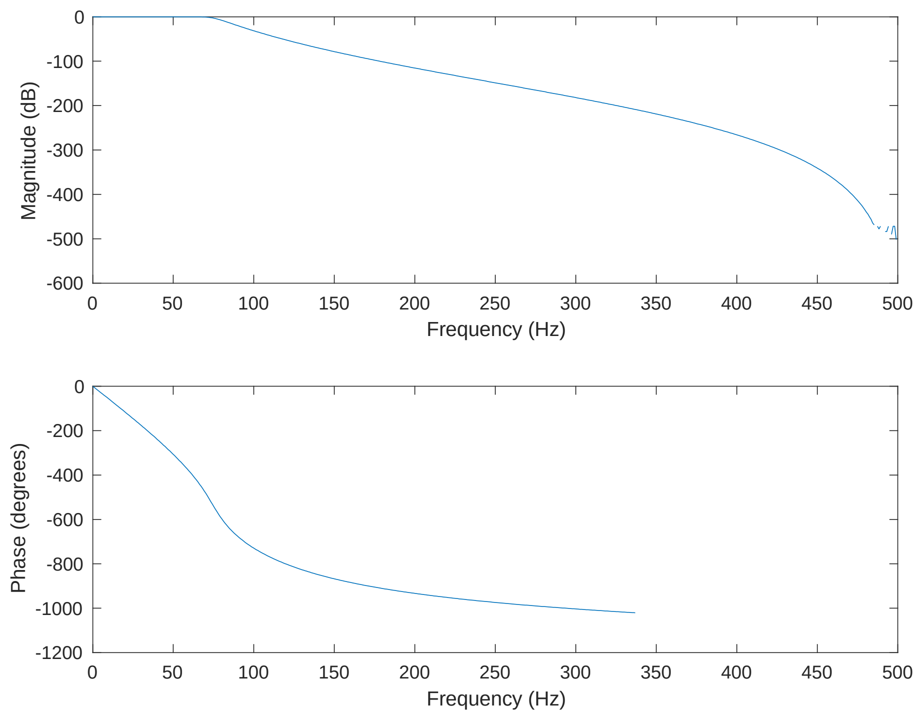

Figure 1: Butterworth filter plot

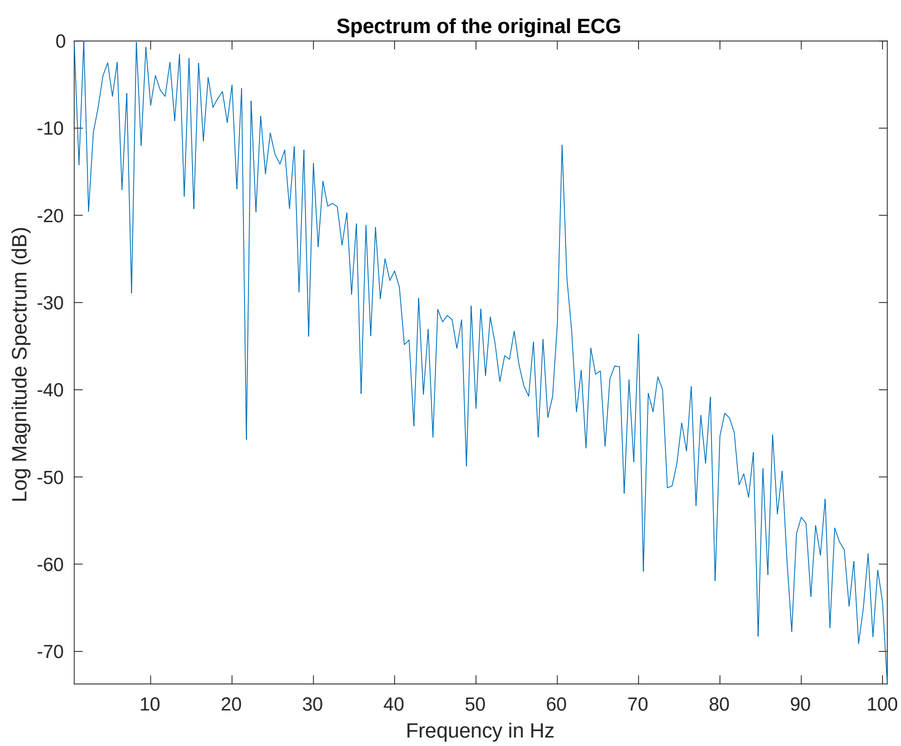

Figure 2: ECG signal in frequncy domain before applying filters

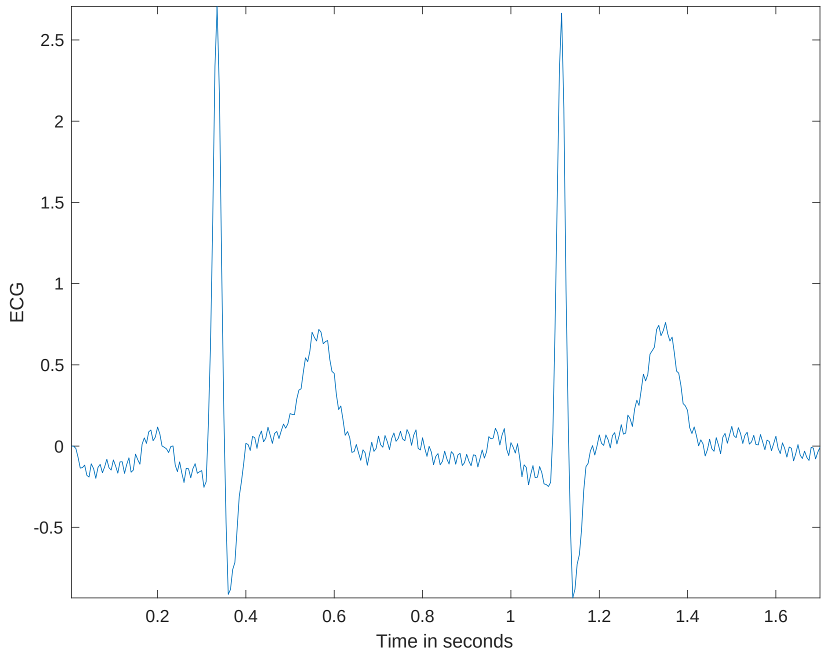

Figure 3: ECG signal before filtering in time domian

- Notch filter

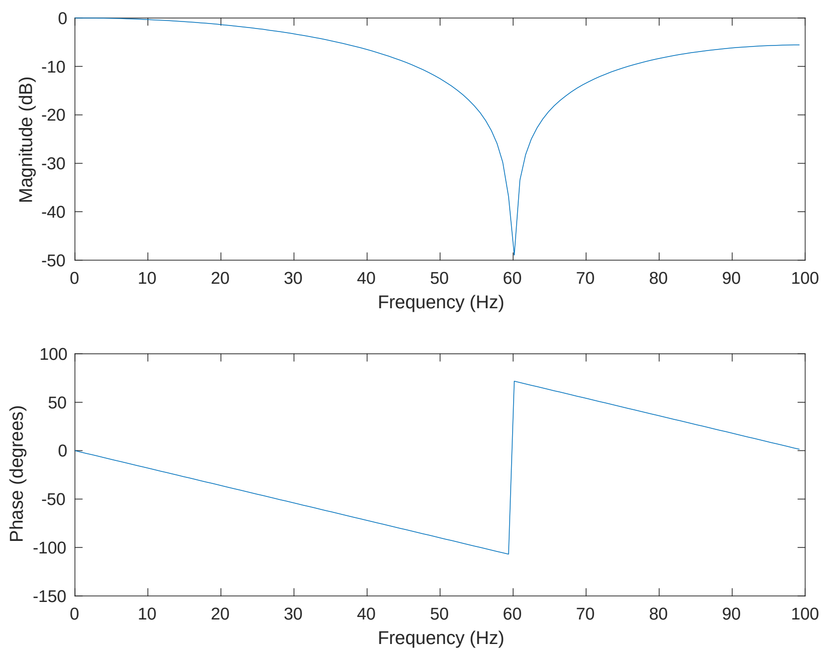

Figure 4: Frequency respnse of notch filter.

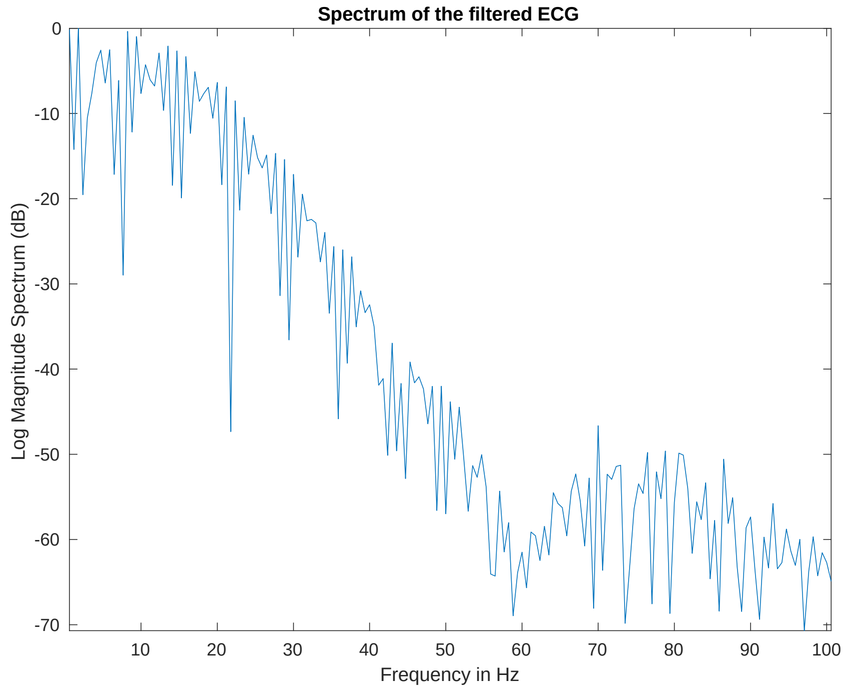

Figure 5: Plot of filtered ECG signal in frequency domain.

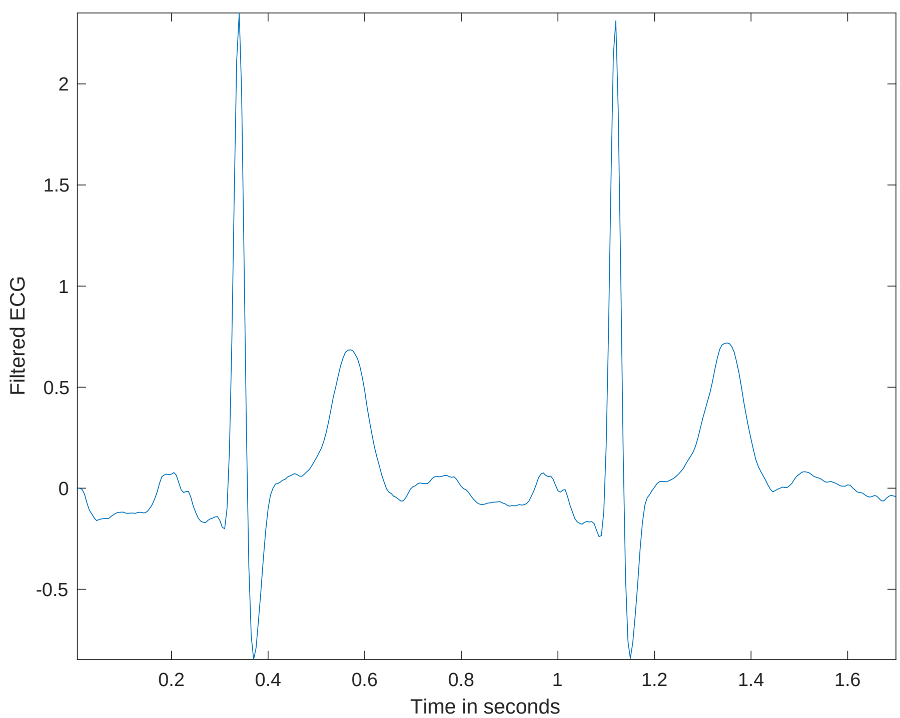

Figure 6: Plot of filtered ECG signal in time domain.

- Hanning filter

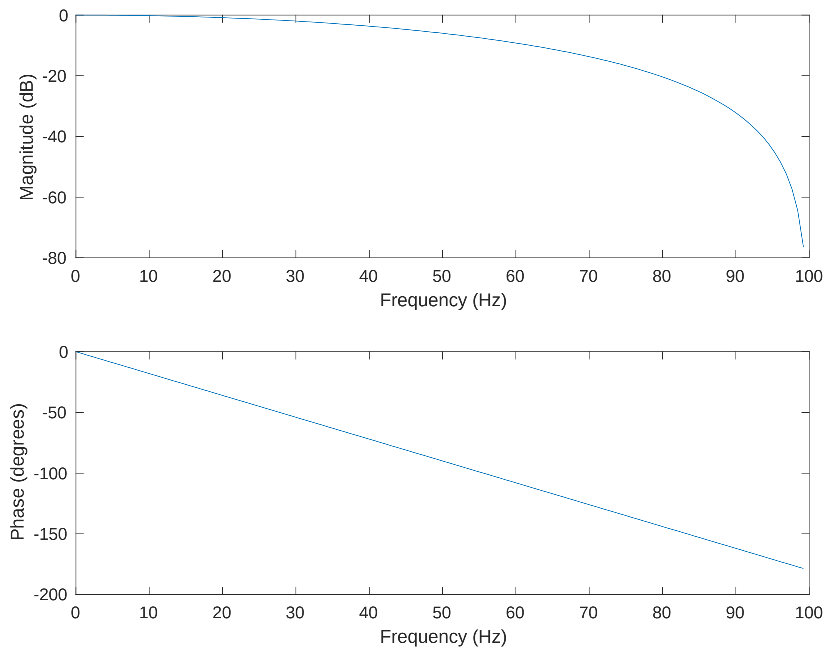

Figure 7: Frequency respnse of hanning filter.

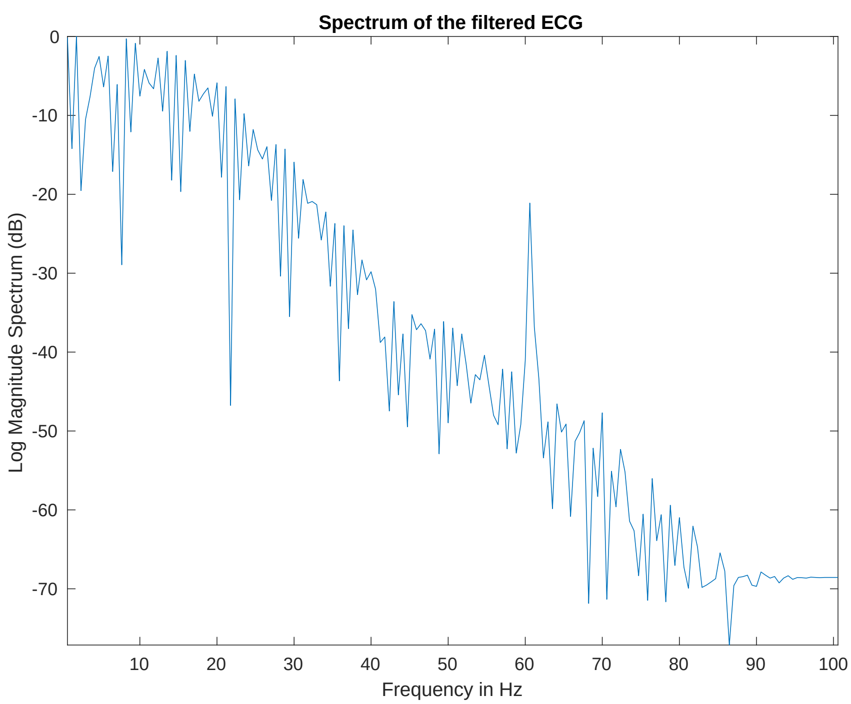

Figure 8: Plot of filtered ECG signal in frequency domain.

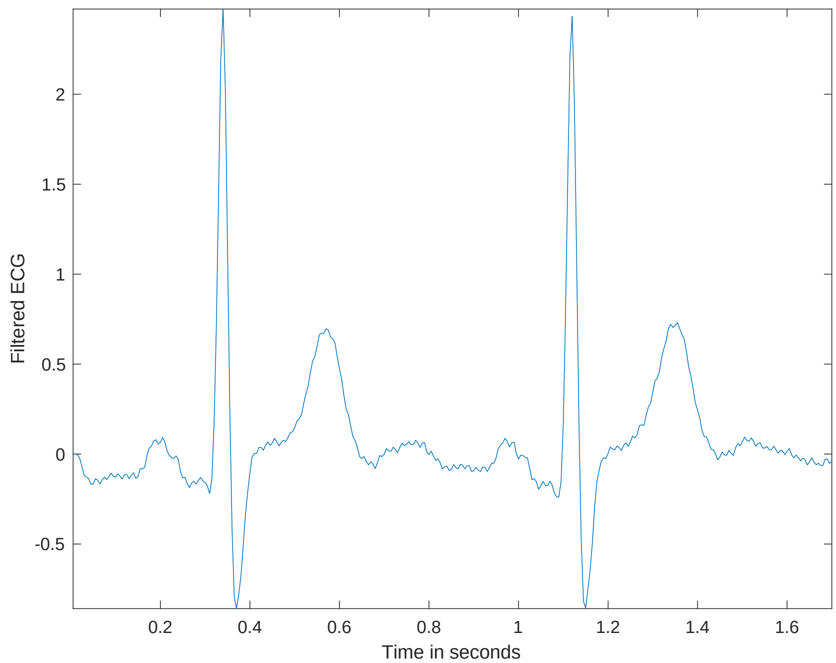

Figure 9: Plot of filtered ECG signal in time domain.

- Derivative based filter

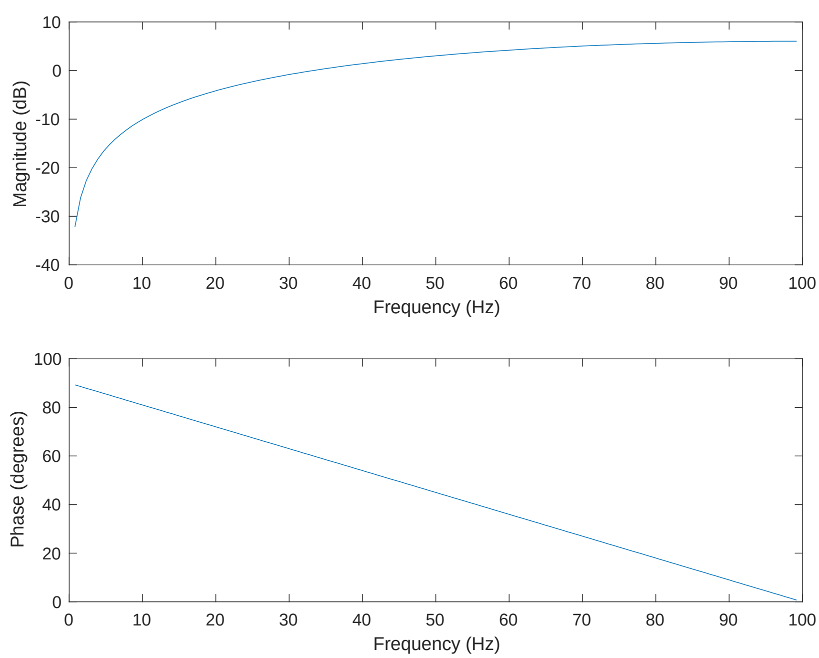

Figure 10: Frequency respnse of drivative based filter.

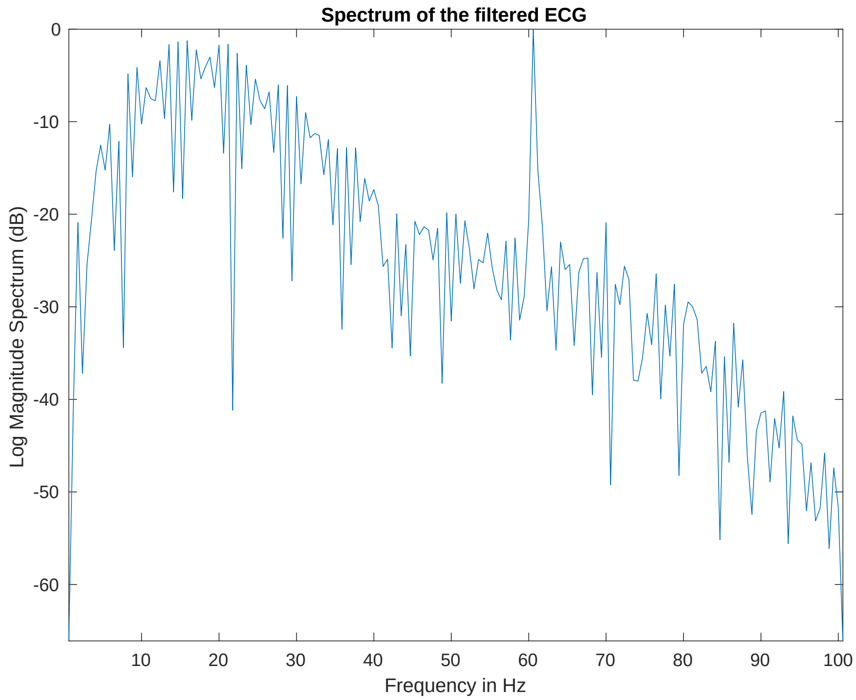

Figure 11: Plot of filtered ECG signal in frequency domain.

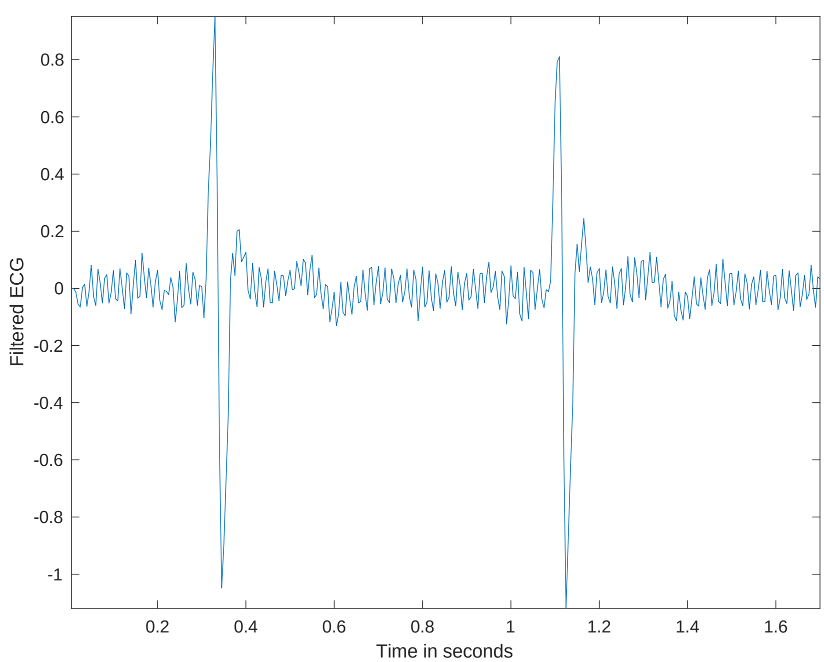

Figure 12: Plot of filtered ECG signal in frequency domain.

- Combined filters

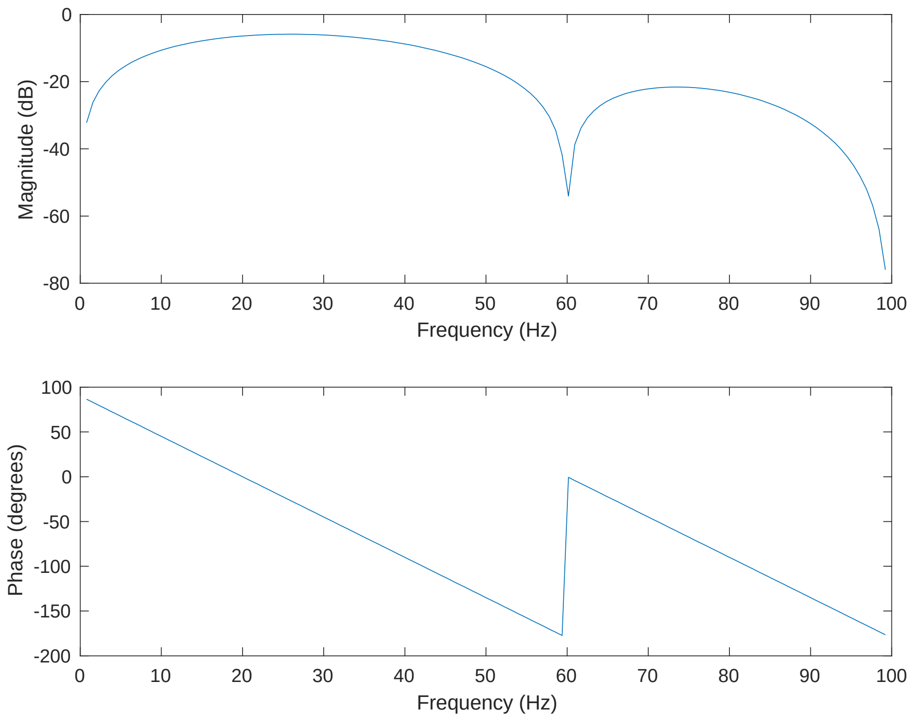

Figure 13: Frequency respnse of all the filters convlouted to each other.

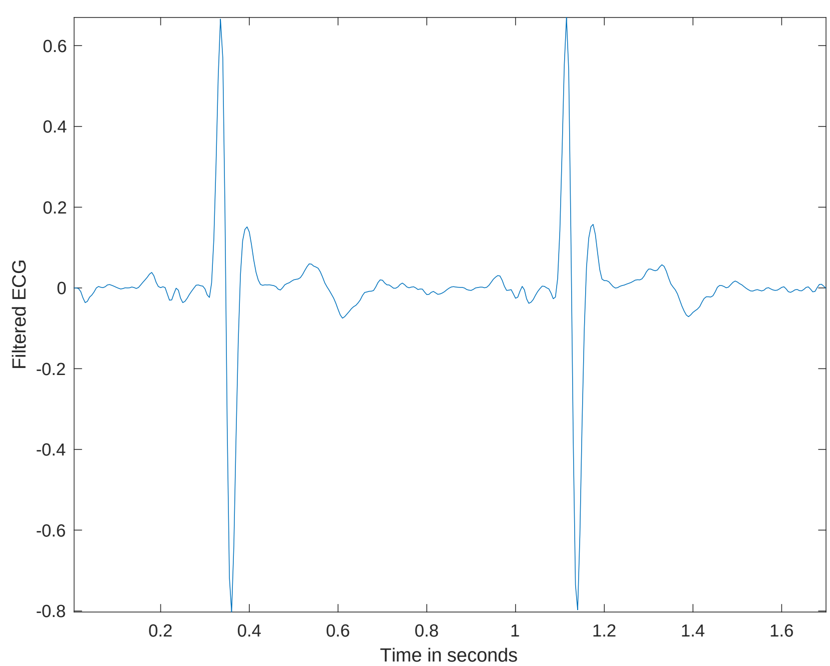

Figure 14: Plot of filtered ECG signal in time domain.

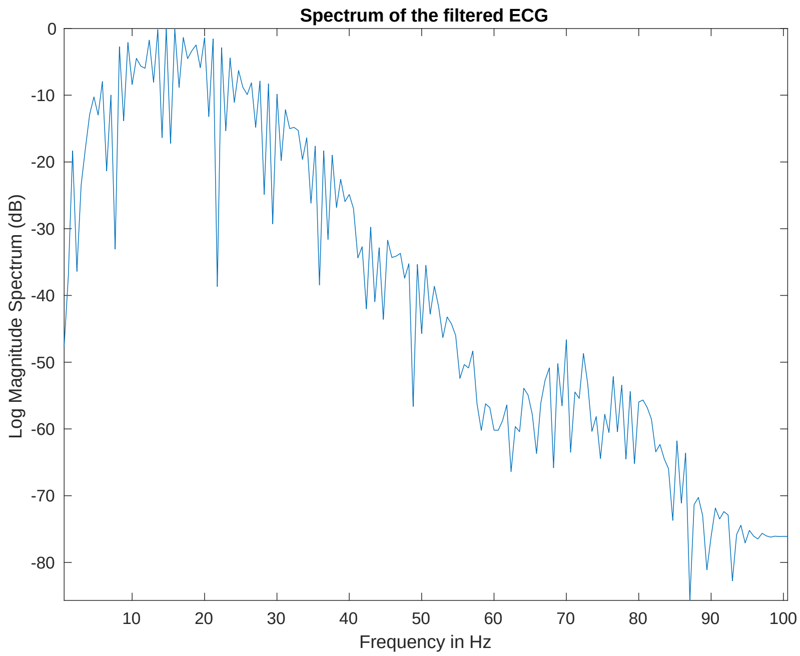

Figure 15: Plot of filtered ECG signal in frequency domain.

Discussion

The lab was divided into four parts and in each part original ECG signal was filtered and shown in time and frequency domain.

Part 1: In this part a notch filer was designed to remove 60 Hz frequency due to power line interference. In order to have a unity gain for the notch filter , the transfer function was divided by H(1). It was observed in the frequency domain that after applying the filter, 60 Hz frequency was eliminated from the signal. In Fig. 6, the effect of notch filter can be seen compare to Fig. 3. The signal is less noisy.

Part 2: Hanning filter is a lowpass filter. Applying hanning filter on the ECG signal filters out the high frequncy noise.

$$H(z) = \frac{1}{4}[1 + 2z^{-1} + z^{-2}] = \frac{1}{4}[1 + z^{-1}]^2.$$

Above is the trasfer function of hanning filter. It can be seen in the Fig. 8 that frequencies after 80 Hz got reduced or eliminated from the ECG signal.

Part 3: Derivative based filter is used to remove low-frequency noise (wandering base-line). It acts as a high pass filter.

$$H(z) = \frac{1}{T}[1 - z^{-1}]$$

Above is the general format of derivative based filter transfer function. In this part, 1/T was taken out to have unit gain at the maximum frequency. The result of filter can be seen in Fig.11 & 12. A simple comparison between Fig.11 and Fig.2 illustrates the effect of the filter on the low frequency signals.

Part 4: This part will combine notch, Hanning, and derivative based filter together. In the time domain simply the transfer function of the filters are multiplied together. However, in frequency domain transfer functions need to convolve to each other. The result of the convolution in z domain is shown below:

```
From input "Combined Filter" to output:

0.25 + 0.4045 z^-1 + 0.1545 z^-2 - 0.1545 z^-3 - 0.4045 z^-4

                                                 - 0.25 z^-5

----------------------------------------------------------------

                            2.618
```

It can be seen in Fig. 14 that ECG signal is smother than the one in Fig.3. Therfore, by combining these three filters together a smooth ECG signal can be obtanined. There can be improvemnts to the filters in order to keep the orignal signal more intact.

Conclusion

To wrap up, the lab familiarized students with filter design and the effect of zeros and poles of the filters' transfer function. It was observed how lowpass, high pass, and notch filter works. Also, how to improve the filter in such way that there is less damage to the original signals. Students also utilized the effect of the filters together on the signal by having only one transfer function.

References

[1] Sirdhar Krishnan, Department Electrical and Computer Engineering (2021), BME772 Lab Manual, Toronto, Ryerson University, Retrieved from class website: https://courses.ryerson.ca/d2l/le/content/516288/viewContent/3744946/View

Appendix

Pre-Lab:

3

$f_0 = 60Hz$
$f_s = 200Hz$

$$\theta = 2\pi \frac{3 \cancel{60}}{5 \cancel{10} \cancel{200}} = \frac{3\pi}{5}$$

$z_1 = e^{j\theta}$
$z_2 = e^{-j\theta}$

$$H'(z) = (z - z_1)(z - z_2)$$

$$H'(z) = (z - e^{j\theta})(z - e^{-j\theta})$$

$$H'(z) = z^2 - z(e^{j\theta} + e^{-j\theta}) + e^{j\theta} \cdot e^{-j\theta}$$

$$H'(z) = z^2 - 2z\cos(\theta) + 1$$

$\rightarrow$ Transfer function

$$H(z) = \frac{z^2 - 2z\cos(\theta) + 1}{z^2}$$

$$H(z) = 1 - 2z^{-1}\cos(\theta) + z^{-2}$$

for having DC gain 1, we normalize the transfer function $H(1) = 1$

$$H(z) = \frac{1 - 2z^{-1}\cos(\theta) + z^{-2}}{2 - 2\cos(\theta)}$$

plug in θ:

$$H(z) = \frac{1 + 0.61z^{-1} + z^{-2}}{2.618}$$

$$H(z) = 0.382 + 0.233\, z^{-1} + 0.382\, z^{-2}$$

+ Ask about the Bandwidth

(2)

$H(z) = [1 - z^{-1}]$ zero at $z=1$

$h[n] = x[n] - x[n-1]$

$H(z) = (1 - e^{-j\omega})$

* NOTE: In discrete time low freq at $2n\pi$ and high freq at $(2n+1)\pi$

$\omega = 0 \rightarrow H(0) = 0$

$\omega = \pi \rightarrow H(\pi) = (1 - (\cancel{\cos(\pi)}^{-1} - j\cancel{\sin(\pi)}^{0}))$

$H(\pi) = 2$

=> it shows that this filter eliminates signals at low freq. and enhances signals at high freq.

3

$$H(z) = (1 + z^{-1})(1 + z^{-1}) \qquad z = +1$$

$$H(z) = 1 + 2z^{-1} + z^{-2}$$

$$z = e^{j\omega} \rightarrow H(e^{j\omega}) = 1 + 2e^{-j\omega} + e^{-2j\omega}$$

$$\omega = 0,\ H(0) = 1 + 2 + 1 = 4$$

$$= \pi,\ H(\pi) = 1 + (2(\overset{-1}{\cancel{\cos(\pi)}} - j\overset{0}{\cancel{\sin(\pi)}}) + \overset{1}{\cancel{\cos(2\pi)}} - j\overset{0}{\cancel{\sin(2\pi)}})$$

$$H(\pi) = 0$$

$\Rightarrow$ we can conclude that the filter is low pass b/c it is zeroing high freq. signals.

MATLAB code:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%      Lab 2: Filtering of the ECG for Noise and Artifact Removal     %
%                                                                       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all                 % clears all active variables
close all

% the ECG signal in the file is sampled at 1000 Hz
% lowpass filter the signal at 75 Hz and downsample by a factor of 5
% this will retain the 60 Hz noise but cause some aliasing artifacts
%
usecg = load('/Users/sadafsafa/Desktop/ecg_60hz (1).dat');
fs = 1000; %sampling rate
fsh = fs/2; %half the sampling rate
[b,a] = butter(12, 75/fsh);
% Butterworth filter frequency response
figure;
M = 512;
```

```
freqz(b, a, M,fs);

lpusecg = filter(b, a, usecg);
usecg = lpusecg;
clear lpusecg;

len = length(usecg);
k = 1;
for i = 1 : length(usecg)
if (rem(i,5) == 0)
ecg(k) = usecg(i);
k = k+1;
end;
end;

fs = 200; %effective sampling rate after downsampling

% Plot of the ECG before filtering
slen = length(ecg);
t = [1:slen]/fs;
figure
plot(t, ecg)
xlabel('Time in seconds');
ylabel('ECG');
axis tight;


% Plot of the spectrum of the ECG before filtering
ecgft = fft(ecg);
ff= fix(slen/2) + 1;
maxft = max(abs(ecgft));
f = [1:ff]*fs/slen; % frequency axis up to fs/2.
ecgspec = 20*log10(abs(ecgft)/maxft);
figure
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz');
ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of the original ECG');
axis tight;

%% define notch filter coefficient arrays a and b
fo=60;
fs=200;
theta=2*pi*(fo/fs);

unit=2-2*cos(theta);
b_notch= [1 -2*cos(theta) 1];
a_notch = unit;
H1=filt(b_notch,a_notch ,'inputname',{'Notch Filter'})
M = 128;

% Notch filter frequency response

figure;
freqz(b_notch, a_notch , M, fs);
```

```
% Output of the notch filter %
y1=filter(b_notch,a_notch,ecg);

% Plot of the ECG after filtering
t= [1:slen]/fs;
figure
plot(t, y1)
xlabel('Time in seconds');
ylabel('Filtered ECG');
axis tight;

% Plot of the spectrum of the ECG after filtering
ecgft = fft(y1);
ff= fix(slen/2) + 1;
maxft = max(abs(ecgft));
f = [1:ff]*fs/slen; % frequency axis up to fs/2.
ecgspec = 20*log10(abs(ecgft)/maxft);
figure
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz');
ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of the filtered ECG');
axis tight;
%
%% Apply hanning


b_han=[1/4 1/2 1/4];
a_han=1;
H2=filt(b_han,a_han,'inputname',{'Hanning Filter'});

figure;
freqz(b_han, a_han, M,fs);
y2=filter(b_han,a_han,ecg);

% Plot of the ECG after filtering
t= [1:slen]/fs;
figure
plot(t, y2)
xlabel('Time in seconds');
ylabel('Filtered ECG');
axis tight;

% Plot of the spectrum of the ECG after filtering
ecgft = fft(y2);
ff= fix(slen/2) + 1;
maxft = max(abs(ecgft));
f = [1:ff]*fs/slen; % frequency axis up to fs/2.
ecgspec = 20*log10(abs(ecgft)/maxft);
figure
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz');
ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of the filtered ECG');
```

```
axis tight;

%% Apply derivative filter

b_deri=[1 -1];
a_deri=1;
% a_deri=[1 -0.995];

H3=filt(b_deri,a_deri,'inputname',{'Derivative Filter'})

figure;
freqz(b_deri,a_deri,M,fs);
y3=filter(b_deri,a_deri,ecg);

% Plot of the ECG after filtering
t= [1:slen]/fs;
figure
plot(t, y3)
xlabel('Time in seconds');
ylabel('Filtered ECG');
axis tight;

% Plot of the spectrum of the ECG after filtering
ecgft = fft(y3);
ff= fix(slen/2) + 1;
maxft = max(abs(ecgft));
f = [1:ff]*fs/slen; % frequency axis up to fs/2.
ecgspec = 20*log10(abs(ecgft)/maxft);
figure
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz');
ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of the filtered ECG');
axis tight;

%% combining all the filters

b_com=conv(b_deri, conv(b_notch, b_han));
a_com=conv(a_deri, conv(a_notch, a_han));

H3=filt(b_com,a_com,'inputname',{'Combined Filter'})

figure;
freqz(b_com,a_com,M,fs);

y4=filter(b_com,a_com,ecg);

% Plot of the ECG after filtering
t= [1:slen]/fs;
figure
plot(t, y4)
xlabel('Time in seconds');
ylabel('Filtered ECG');
axis tight;
```

```
% Plot of the spectrum of the ECG after filtering
ecgft = fft(y4);
ff= fix(slen/2) + 1;
maxft = max(abs(ecgft));
f = [1:ff]*fs/slen; % frequency axis up to fs/2.
ecgspec = 20*log10(abs(ecgft)/maxft);
figure
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz');
ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of the filtered ECG');
axis tight;
```