Ryerson University

Department of Electrical & Computer Engineering
Faculty of Engineering & Architectural Science

Course Title:	Biomedical Signal Analysis
Course Number:	BME 772
Semester/Year (e.g.F2016)	F 2020

Instructor:	Sri Krishnan

Assignment/Lab Number:	2
Assignment/Lab Title:	Filtering of ECG for the removal of noise & 60Hz Power Line Interference

Submission Date:	10/21/2020
Due Date:	10/23/2020

Student LAST Name	Student FIRST Name	Student Number	Section	Signature*
Mullen	Andrew	500 787 631	4	

Reset Form

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: http://www.ryerson.ca/senate/current/pol60.pdf

Pre Lab

1. $\theta_0 = 2\pi \frac{f_0}{f_s} = 2\pi \frac{60}{200} = 1.88 \text{ rads} = 107.7°$

$z_1 = \cos(w_0) + j\sin(w_0) = \cos(107.7) + j\sin(107.7) = -0.3040 + j\,0.9527$

$z_2 = \cos(w_0) - j\sin(w_0) = \cos(107.7) - j\sin(107.7) = -0.3040 - j\,0.9527$

$H(z) = (1 - z_1 z^{-1})(1 - z_2 z^{-1})$ $\qquad z_1 z_2 = 0.0924 + 0.9076$

$= (1 - z_1 z^{-1} - z_2 z^{-1} + z_1 z_2 z^{-2})$ $\qquad z_1 z_2 = 1$

$= (1 - (-0.3040 + \cancel{j\,0.9527}) z^{-1} - (-0.3040 - \cancel{j\,0.9527}) z^{-1} + 1 z^{-2}$

$H(z) = (1 + 0.608 z^{-1} + 1 z^{-2})$

$|H(1)| = |1 + 0.608 + 1|$
$= 2.608$

$\rightarrow$ set gain to be unity $\therefore k = \frac{1}{|H(1)|} = \frac{1}{2.608}$
$= 0.383$

$$\boxed{\therefore H(z) = 0.383 [1 + 0.608 z^{-1} + 1 z^{-2}]}$$

2.) zero @ $z = 1$ $\quad \therefore H_1(z) = \frac{z - 1}{z} = 1 - z^{-1}$

$\hookrightarrow$ $\boxed{h_1(n) = \delta[n] - \delta[n-1]}$ $\therefore$ high pass filter

3.) $z_1 = -1 \qquad z_2 = -1$

$H_2(z) = \frac{(z - z_1)(z - z_2)}{z^2} = \frac{(z+1)^2}{z^2} = \frac{z^2 + 2z + 1}{z^2}$

$H_2(z) = 1 + 2z^{-1} + z^{-2}$

$\boxed{\therefore h_2(n) = \delta[n] + 2\delta[n-1] + \delta[n-2]}$

$\hookrightarrow$ low pass filter

Introduction

This lab examines the use of filters in ECGs. An ECG (electrocardiogram) is a non-invasive way of recording the electrical signals in your heart [1]. These signals are used to aid physicians in diagnosing heart problems and assessing the general health of your heart. When obtaining ECG signals there is a filtering process that must be done to remove the noise. Noise is any unwanted information that is present in the signal. Noise can be high frequency from electromyogram (EMG), low frequency (baseline wander) and specific frequency (60Hz powerline interference) [2]. Each of these can be reduced by a low pass filter, high pass filter and a notch filter respectively. Another type of noise present in signals is motion artifact noise which can be reduced by instructing the subject to remain still during the taking of the ECG [2]. The presence of this noise can alter the look of the ECG and careful removal of it must be taken to ensure that signal integrity remains. Taking a closer look at using a notch filter to remove the powerline interference, this is accomplished by placing a zero on the unit circle at an angular position that corresponds to the frequency of interference (60 Hz). Calculation of the angular position of the zero is given by [3]:

$$\theta_0 = 2\pi \frac{f_0\ (notch\ filter\ frequency)}{f_s\ (sampling\ frequency)}$$

Complex zeros come in complex conjugate pairs so the zero is reflected across the x axis such that the two zeros have the same real value but oppositely signed imaginary values. This is then used to realize the notch filter transfer function as [3]:

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

As stated, there are different types of noises and each of these require different filters to remove the noise. It is possible to combine these filters through multiplication in the frequency domain. Whatever is multiplied in the frequency domain however must be convoluted in the time domain and so as seen in the last portion of this lab, the convolution operation is employed to combine the three filters into one filter that accomplishes the same filtering as the previous three except all in one.

Results

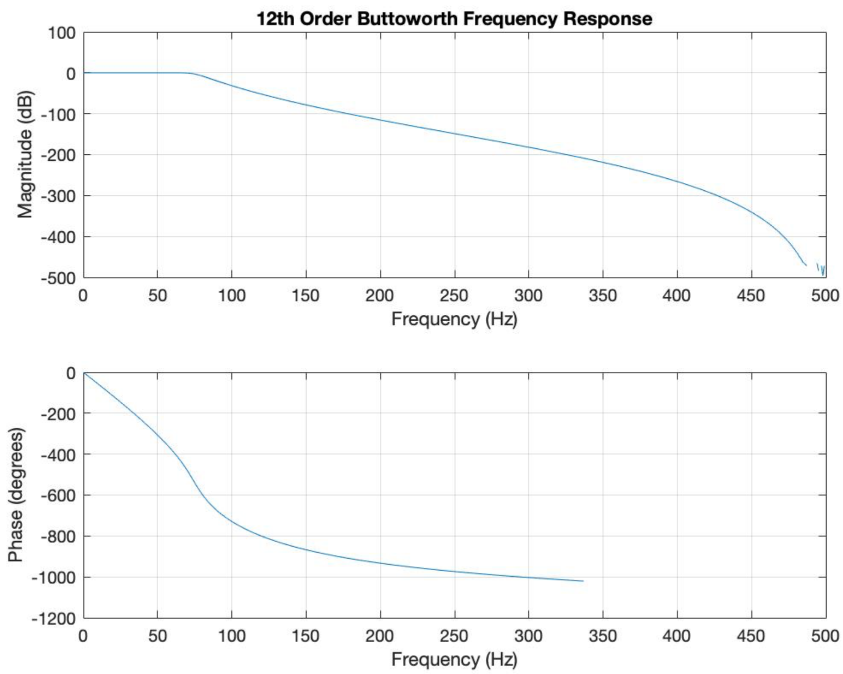

Figure 1: Frequency response of a 12th order buttworth filter

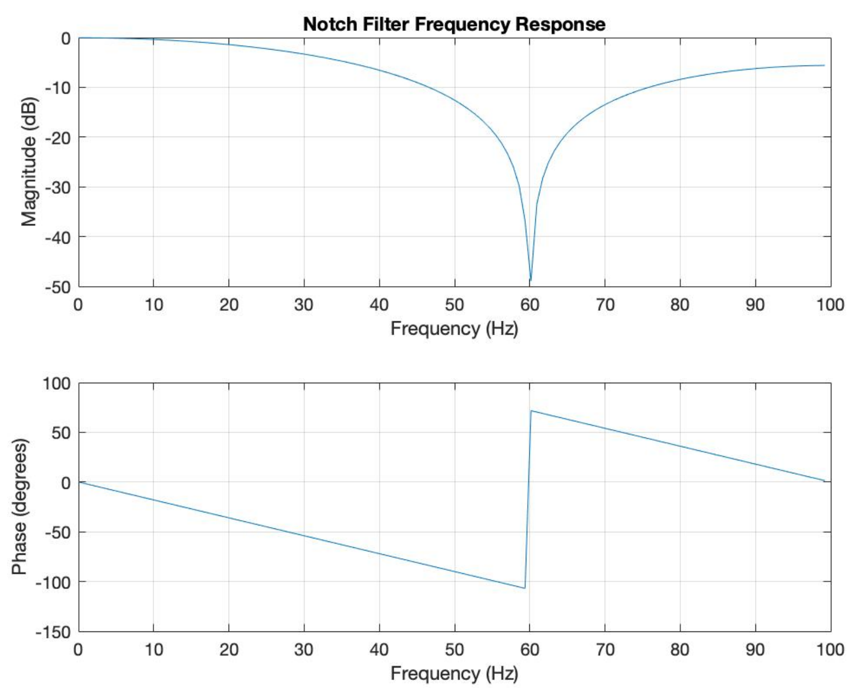

Figure 2: Frequency response of a 60Hz notch filter

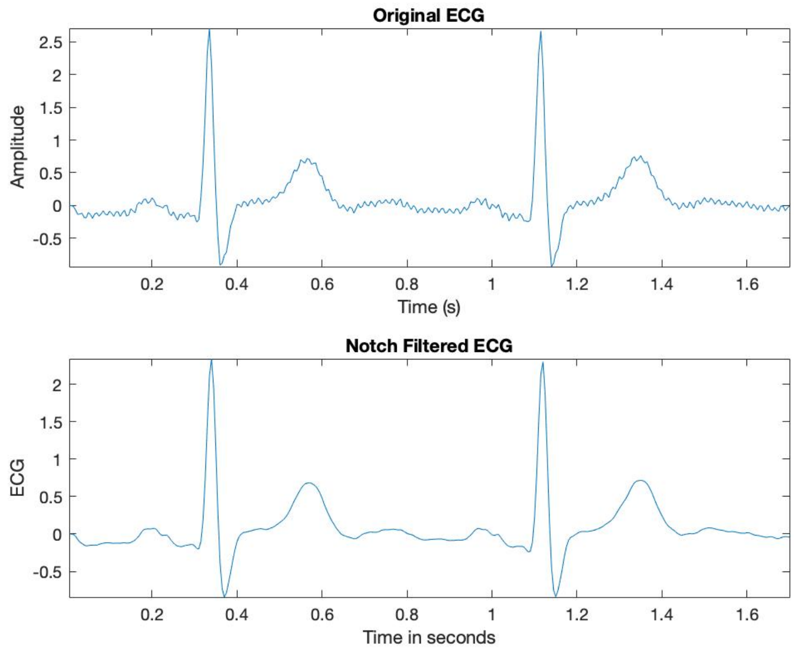

Figure 3: Original and 60 Hz notch filtered ECG signal

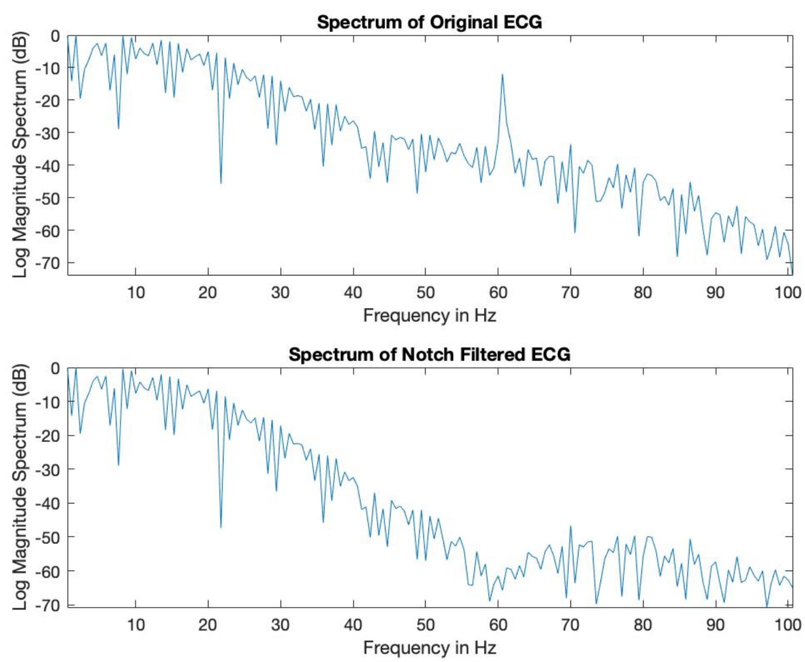

Figure 4: Frequency spectrum comparison of original and 60 Hz notch filtered ECG

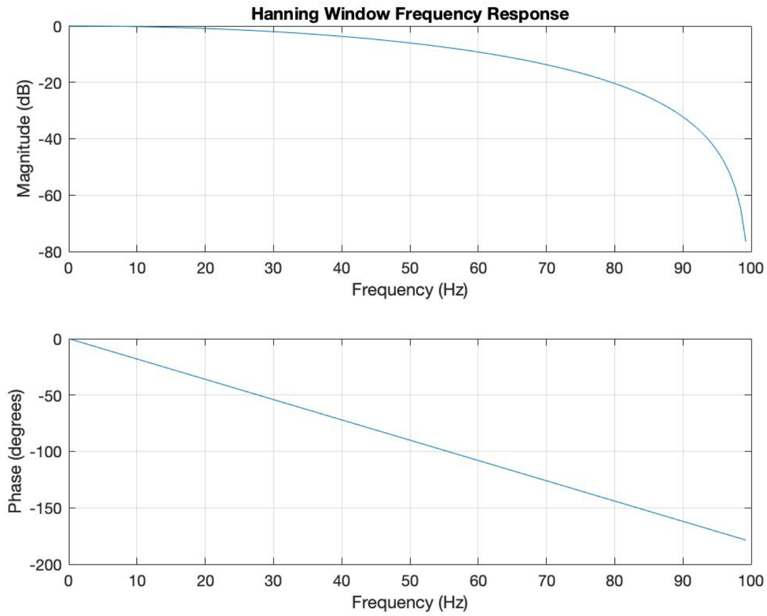

Figure 5: Frequency response of hanning window filter

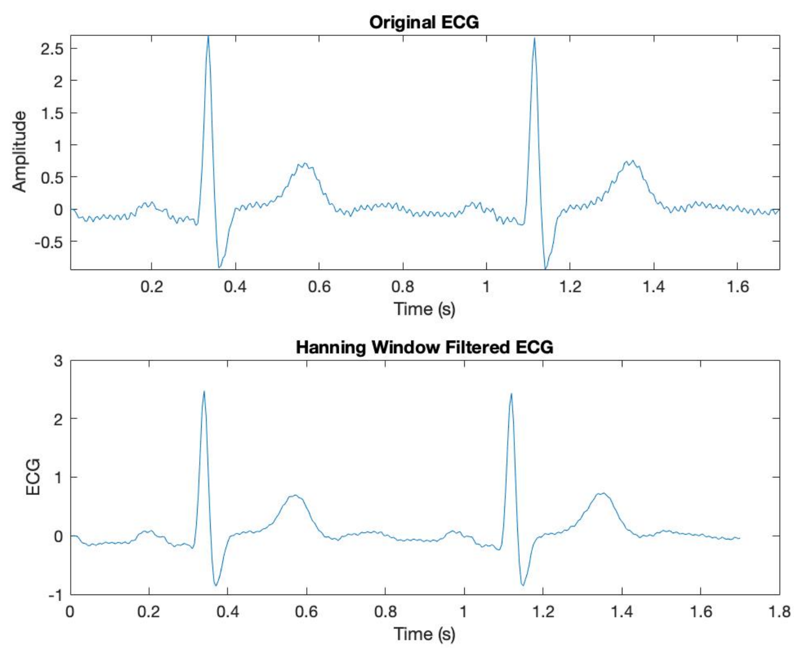

Figure 6: Original and hanning window filtered ECG signal

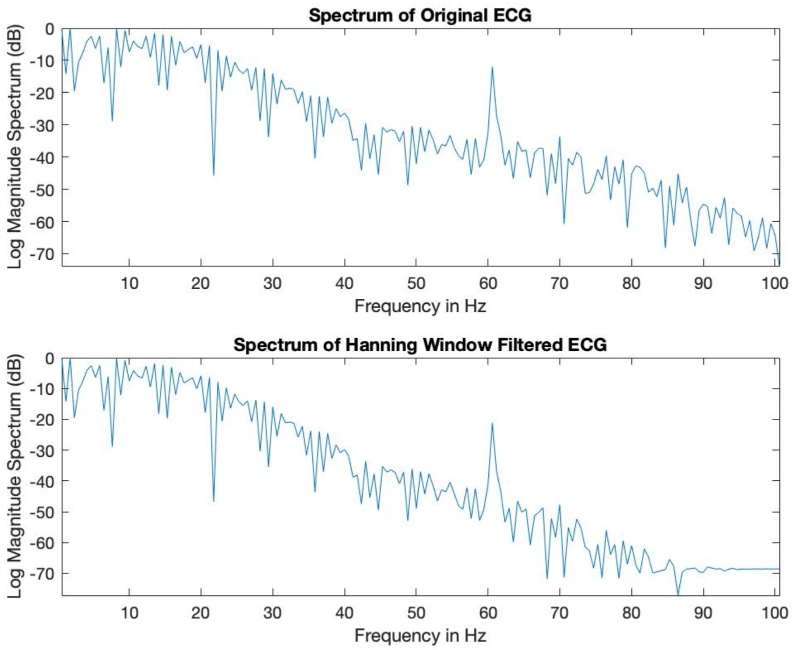

Figure 7: Frequency spectrum of original and hanning window filtered ECG signals

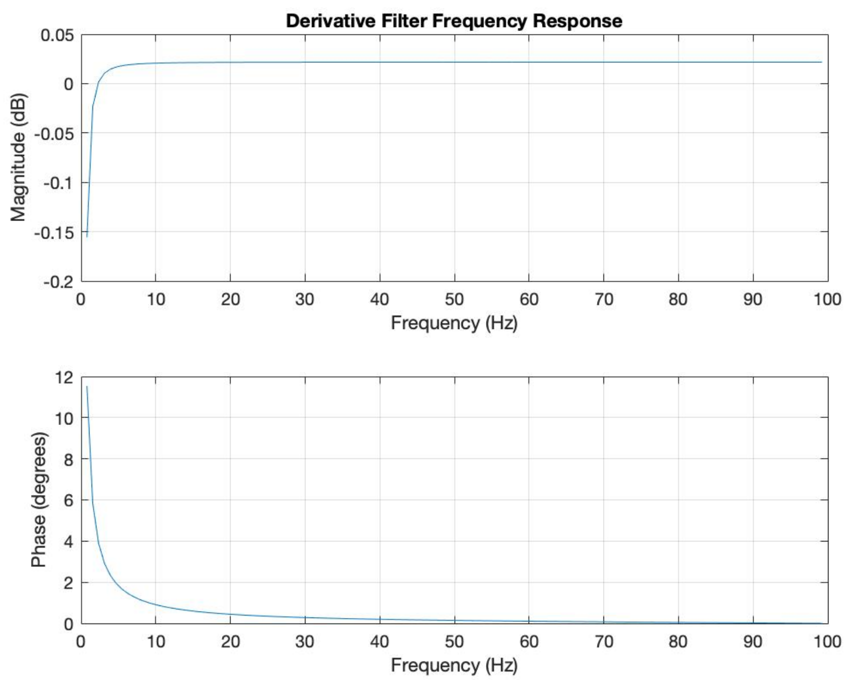

Figure 8: Frequency response of derivative filter

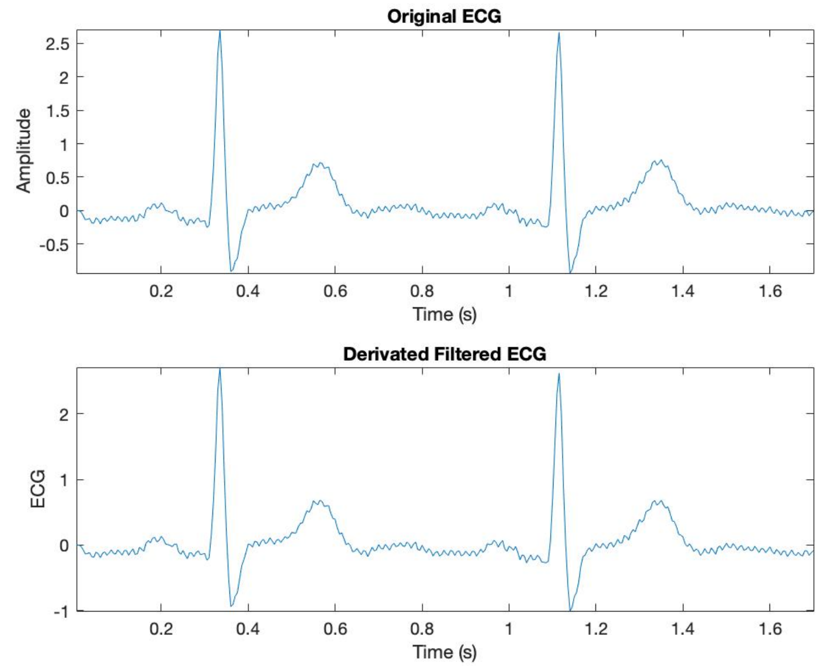

Figure 9: Original and derivative filtered ECG signal

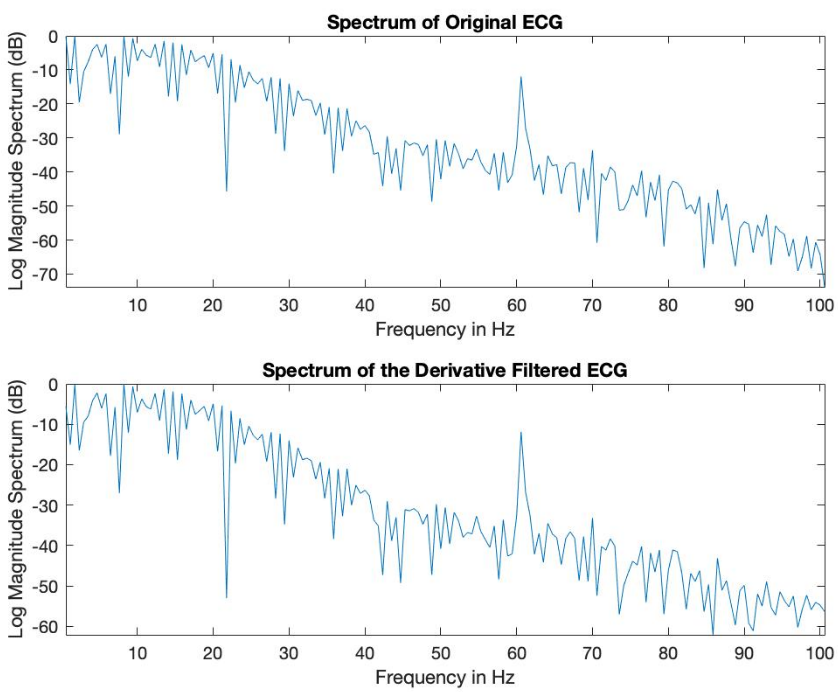

Figure 10: Frequency spectrum of original and derivative filtered ECG signals

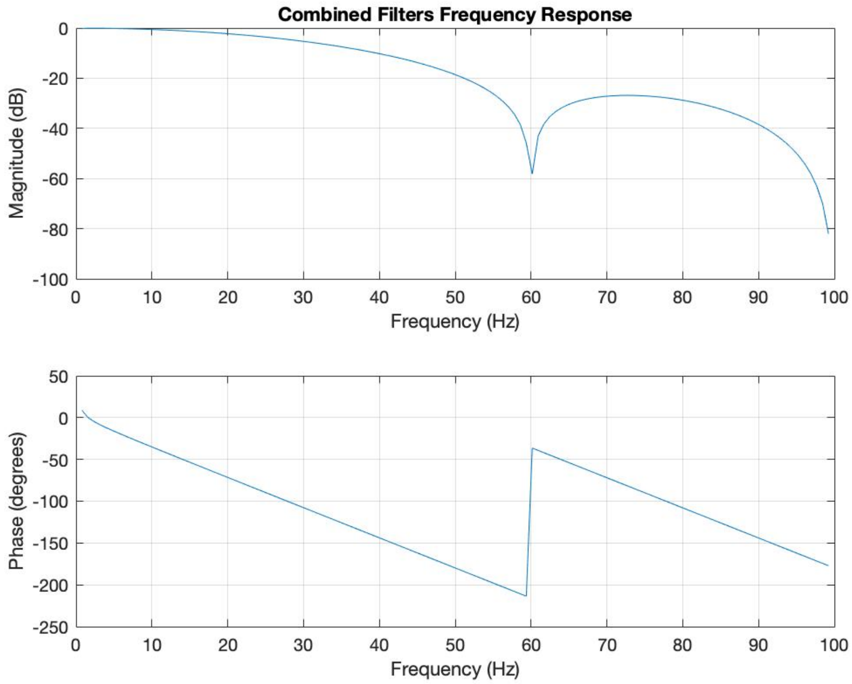

Figure 11: Frequency response of notch, hanning and derivative filter combination

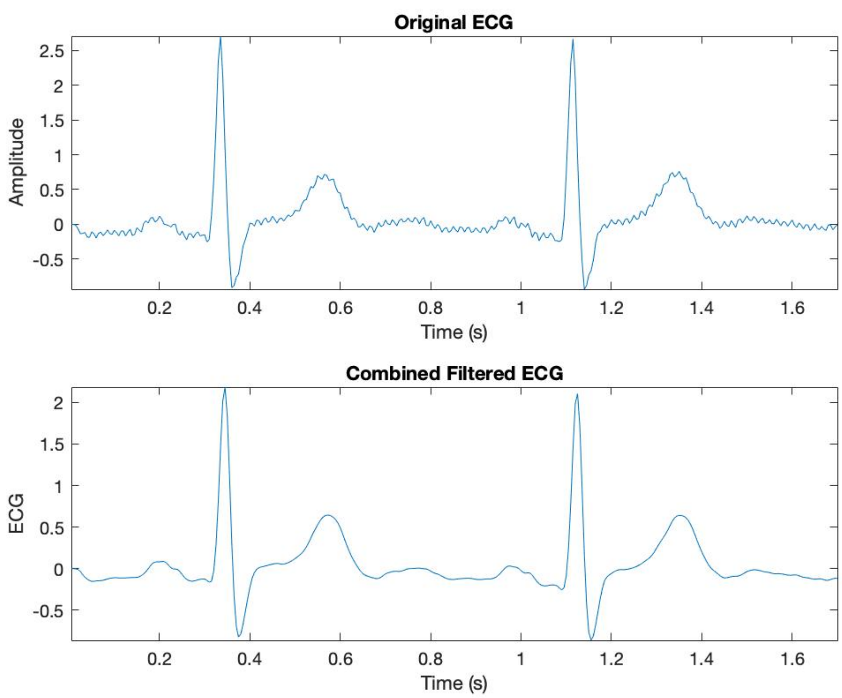

Figure 12: Original and combined filter ECG signal

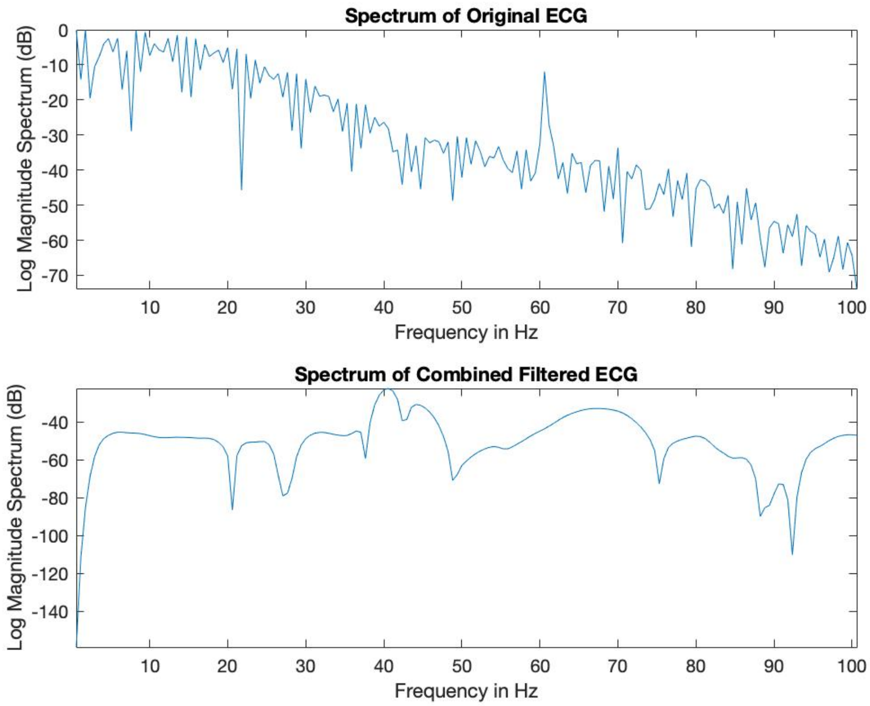

Figure 13: Frequency spectrum of original and combined filtered ECG signals

Conclusion

This lab showcases different filters and their noise filtering effects on ECG signals. The first filter explored was the notch filter that was utilized to remove 60Hz power line interference. The frequency response (figure 2) shows a dip in the magnitude at 60 Hz which is then represented in the frequency spectrum (figure 4) as the value around 60Hz is lowered. This shows up in the signal (figure 3) by the removal of the constant minor variations seen throughout the signal. The second filter explored was the hanning window filter and as seen in the frequency response (figure 5) it is a low pass filter that is used for EMG noise reduction. In the signal (figure 6) you can see that same high frequency minor variation is removed as well as frequencies higher than that. The derivative filter was applied to treat any baseline wander in the signal as well as other lower frequencies. The frequency response (figure 8) shows that this is a high pass filter with a cut-off frequency of approximately 4Hz. This is not as noticeable in the signal (figure 9) as the original ECG does not showcase noticeable baseline wander. The last filter tried was a combination of the previous three filters described, the notch, hanning window and derivative filter. This filter is supposed to combine the benefits of the constituent filters. The frequency response (figure 11) makes visible the reduction in frequencies 60 Hz and higher. This is showcased in the signal (figure 12) as well as a smooth ECG signal is seen with less noise. The frequency spectrum (figure 13) showcased the biggest change. Instead of a spectrum that was highly variable and constantly decreasing from 0dB to -70dB from 0Hz to 100Hz, the new filtered spectrum was smoother and started at -50dB at 4Hz and finished at -50dB at 100Hz. The only variations in there were smooth in nature and less volatile with a couple sharp declines.

The implementation of the notch filter in Matlab was done using the equations showcased in the introduction. The resulting transfer function had numerator coefficients of [1, 0.618, 1] with a gain of 0.318 and the denominator coefficients were [1, 0, 0]. These coefficients were sent to the Matlab function 'filter' along with the signal to be filtered. This would then output the filtered signal. To get the frequency response plots the function 'freqz' was used with arguments of numerator coefficients, denominator coefficients, length of the signal and the sampling frequency. The other plots formulated were the frequency spectrum of the filtered ECGs and that was accomplished by taking the fast fourier transform of the ECG using the 'fft' function. This was then plotted in decimals by taking 20 times the log of the absolute value of the fast fourier transform signal.

References

[1] “Electrocardiogram (ECG or EKG) - Mayo Clinic", *Mayoclinic.org*, 2020. [Online]. Available: https://www.mayoclinic.org/tests-procedures/ekg/about/pac-20384983. [Accessed: 22- Oct- 2020].

[2]]R. Kher, "Signal Processing Techniques for Removing Noise from ECG Signals", *Journal of Biomedical Engineering and Research*, 2019. Available: 10.17303/jber.2019.3.101.

[3] Krishna, Sridhar. LAB 2 – Filtering of ECG for the Removal of Noise and 60 Hz Power-Line Interference, BME772 (Biomedical Signal Analysis - FALL2020) -Laboratory Manual. Toronto. Ryerson University, 2020.

Appendix

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%     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('ecg_60hz.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);
title('12th Order Buttoworth Frequency Response');

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;

%% PART 1 Notch filter
% define notch filter coefficient arrays a and b

b_notch = 0.38*[1 0.618 1];     % Numerator coefficients
a_notch = [1 0 0];              % Denominator Coefficients

M = 128;

% Notch filter frequency response
figure;
freqz(b_notch, a_notch, M, fs);
title('Notch Filter Frequency Response');

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

% Plot of the ECG after filtering
figure;
subplot(211)
plot(t, ecg);
xlabel('Time (s)'); ylabel('Amplitude'); title('Original ECG'); axis tight;

subplot(212)
plot(t, ecg_notch);
xlabel('Time in seconds'); ylabel('ECG'); title('Notch Filtered ECG'); axis
tight;

% Plot of the spectrum of the ECG after filtering
ecgft_notch = fft(ecg_notch);
ff= fix(slen/2) + 1;
maxft = max(abs(ecgft_notch));
f = [1:ff]*fs/slen; % frequency axis up to fs/2.
ecgspec_notch = 20*log10(abs(ecgft_notch)/maxft);

figure;
subplot(211);
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Original ECG'); axis tight;

subplot(212)
plot(f, ecgspec_notch(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Notch Filtered ECG'); axis tight;

%% Part 2 Hanning filter
b_han = [0.25 0.5 0.25];  % Numerator Coeffecients
a_han = [1 0 0];          % Denominator Coefficients
```

```
% Hanning Window Frequency Response
figure;
freqz(b_han, a_han, M, fs);
title('Hanning Window Frequency Response');

% ECG output with hanning window
ecg_han = filter(b_han, a_han, ecg);

% Plot Filtered Output
figure;
subplot(211)
plot(t, ecg);
xlabel('Time (s)'); ylabel('Amplitude'); title('Original ECG'); axis tight;

subplot(212)
plot(t, ecg_han);
title('Hanning Window Filtered ECG'); xlabel('Time (s)'); ylabel('ECG');

% Frequency Spectrum of ECG after filtering
ecgft_han = fft(ecg_han);
maxft = max(abs(ecgft_han));
ecgspec_han = 20*log10(abs(ecgft_han)/maxft);
figure;
subplot(211);
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Original ECG'); axis tight;

subplot(212)
plot(f, ecgspec_han(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Hanning Window Filtered ECG'); axis tight;

%% Part 3 Derivative filter
b_der = [1 -1];            % Numerator Coeffecients
a_der = [1 -0.995];         % Denominator Coeffecients

% Derivative Filter Frequency Response
figure;
freqz(b_der, a_der, M, fs);
title('Derivative Filter Frequency Response');

% Output of Derivative Filter
ecg_der = filter(b_der, a_der, ecg);

% Plot Filtered Output
figure;
subplot(211)
plot(t, ecg);
xlabel('Time (s)'); ylabel('Amplitude'); title('Original ECG'); axis tight;

subplot(212)
plot(t, ecg_der);
xlabel('Time (s)'); ylabel('ECG'); title('Derivated Filtered ECG'); axis
tight;
```

```
% Frequency Spectrum of ECG after Filtering
ecgft_der = fft(ecg_der);
maxft = max(abs(ecgft_der));
ecgspec_der = 20*log10(abs(ecgft_der)/maxft);

figure;
subplot(211);
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Original ECG'); axis tight;

subplot(212)
plot(f, ecgspec_der(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of the Derivative Filtered ECG'); axis tight;

%% Part 4 Combined Filters
% Convolute filters
b1 = conv(b_notch, b_han);
b2 = conv(b_der, b1);

a1 = conv(a_notch, a_han);
a2 = conv(a_der, a1);

figure;
freqz(b2, a2, M, fs);
title('Combined Filters Frequency Response');

% Output of Combined Filter ECG
ecg_comb = filter(b2, a2, ecg);

% Plot Filtered Output
figure;
subplot(211)
plot(t, ecg);
xlabel('Time (s)'); ylabel('Amplitude'); title('Original ECG'); axis tight;

subplot(212)
plot(t, ecg_comb);
xlabel('Time (s)'); ylabel('ECG'); title('Combined Filtered ECG'); axis
tight;

% Frequency Spectrum of ECG after Filtering
ecgft_comb = fft(ecg_comb);
maxft = max(abs(ecgft_comb));
ecgspec_comb = 20*log10(abs(ecg_comb)/maxft);

figure;
subplot(211);
plot(f, ecgspec(1:ff));
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Original ECG'); axis tight;

subplot(212)
plot(f, ecgspec_comb(1:ff));
```

```
xlabel('Frequency in Hz'); ylabel('Log Magnitude Spectrum (dB)');
title('Spectrum of Combined Filtered ECG'); axis tight;
```