Ryerson University

Department of Electrical & Computer Engineering

BME 632 - Signals & Systems II

Lab Report

Lab Number: 4

Instructor: April Khademi

Section: 2

Due Date: April 3, 2020

Student Name [redacted]

Signature: [redacted]

Student Name (ID): **Pass Fail** **Kushal Shah** **500843903**

Signature: KS

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: www.ryerson.ca/senate/current/pol60.pdf.

Before submitting your report, your TA asks questions about your report. If there is no consistency between your oral answer and your report, you will lose 50% of your total mark. For this part, Pass or fail will be circled next to your name accordingly.

Introduction:

In all previous labs we have discussed how we are measuring electrical signals created by the body. Now, we are measuring the system in the body which is creating these electrical signals. Neurons are the circuitry for the body, connecting the whole body to the CNS which is the control centre for the body. Neurons have an impact on any movement for psychological processes of the body. There are three types of neurons; sensory, motor, and interneurons, each with different and important purposes. The brain, the most important part of the CNS, is made of 100 million densely packed neurons which are either myelinated or unmyelinated. These neurons are organized into different lobes which all have different roles and functionalities. As previously stated, when the neurons fire they create electrical signals which we can measure. These neurons, however, do not fire at the same frequencies and the summation of all these signals create the one main signal which we call an EEG signal. When processing, the signal can be broken down into four main components/frequency ranges. The ranges are defined as primary components of the EEG. These components are: α (alpha wave), β (beta wave), Δ (delta wave), and θ (theta wave). Each of these components have different frequency ranges and different amplitude ranges. These components also correspond to one of the previously mentioned lobes of the brain, which each are stimulated from different activities. To measure all of these activities have to place electrodes on specific parts of the head which are outlined in criteria like the 10-20 system. Medically, EEGs are very important when it comes to mental health and illnesses such as epilepsy and seizures. It is very important for biomedical engineers to be able to process such noisy signals to useful outputs such as EEGs. This lab will allow us to gain this skill.

<u>Experiment: Electroencephalogram Analysis</u>

1)

a) The EEG signal is predominantly located in the frequency range of 0.5-30 Hz. Therefore, for the bandpass filter we will use these two frequencies (0.5 Hz and 30 Hz) as our cutoff frequencies. The sampling frequency is 256Hz and this was considered when making the filter. We will also use a tenth order Chebyshev Type I bandpass filter. This will remove all of the noise created by the rest of the body. Heart beats, breathing, muscle contractions, anything that creates an electrical signal or excites the brain above the resting potential.

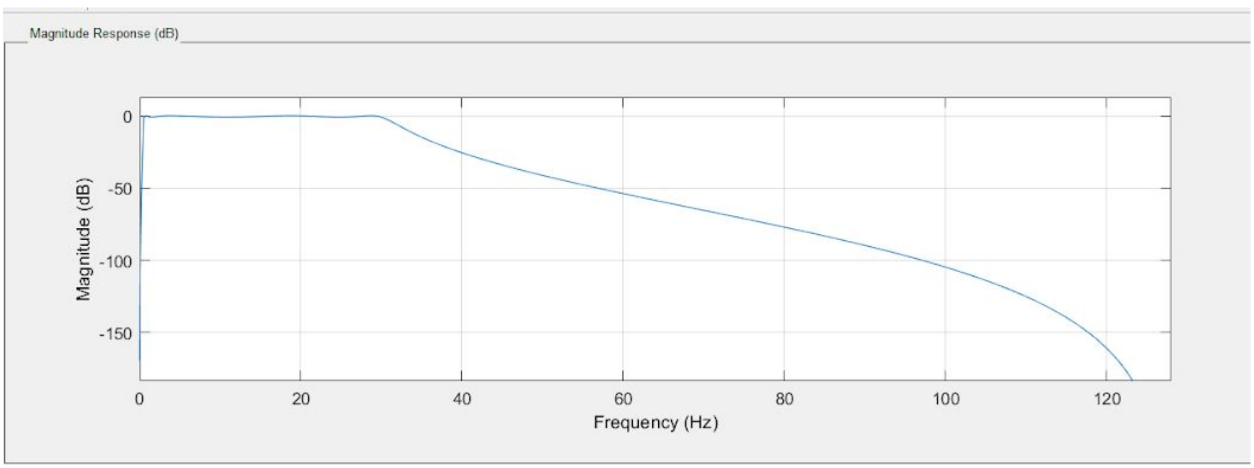

Figure 1. Magnitude response of the bandpass filter used to denoise the signal.

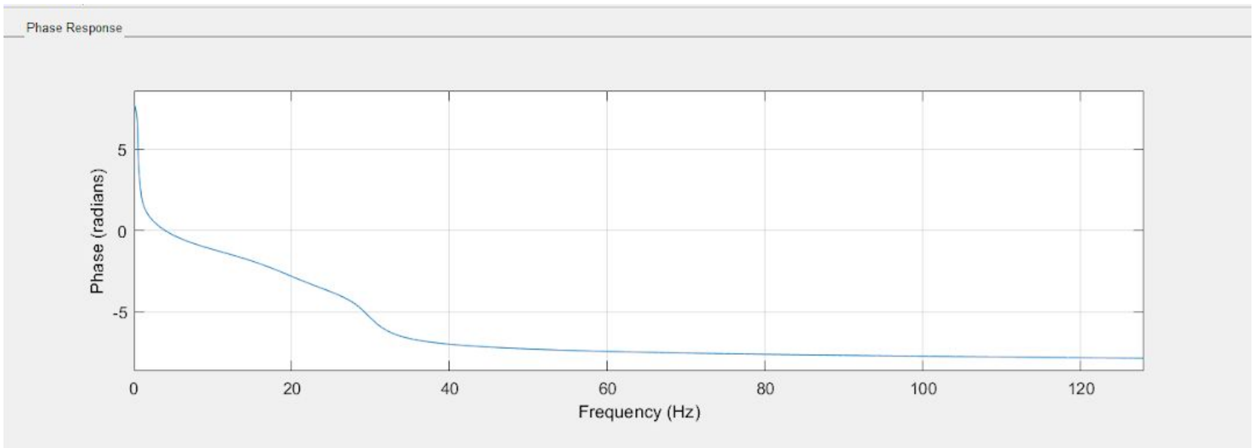

Figure 2. Phase response of the bandpass filter used to denoise the EEG signal.

b) The parameters listed above were carefully chosen for the most accurate results. The frequency range was specifically used to ensure only the EMG signal would be outputted. The tenth order Chebyshev Type 1 bandpass filter was chosen because in lab 3 we concluded that this is the best filter for biomedical signals. It has a very flat peak and sharp drop, which means it will hold the integrity of the desired signal.

2) The below plots were not normalized to make comparing the signals easier.

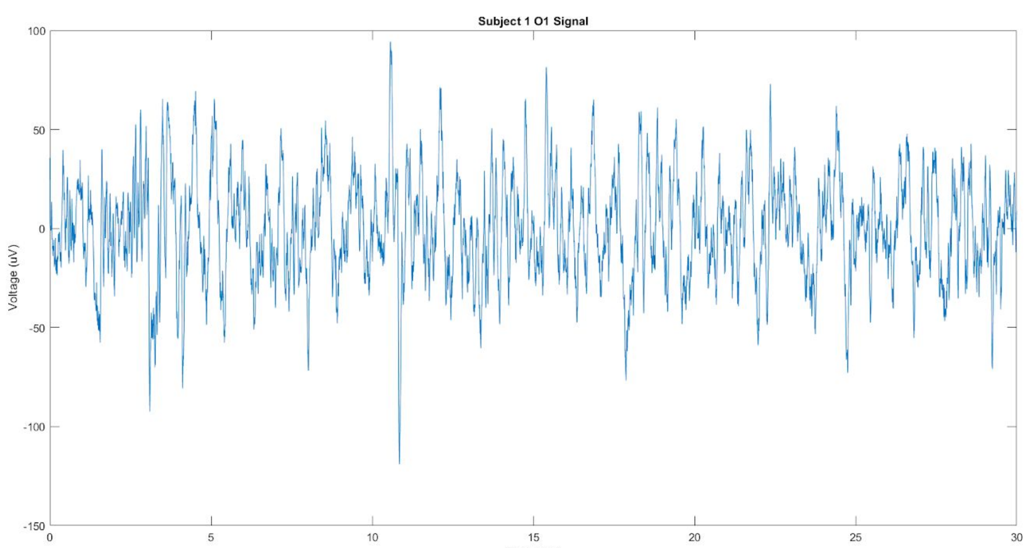

Figure 3. Plot of the unfiltered O1 signal from subject1 data.

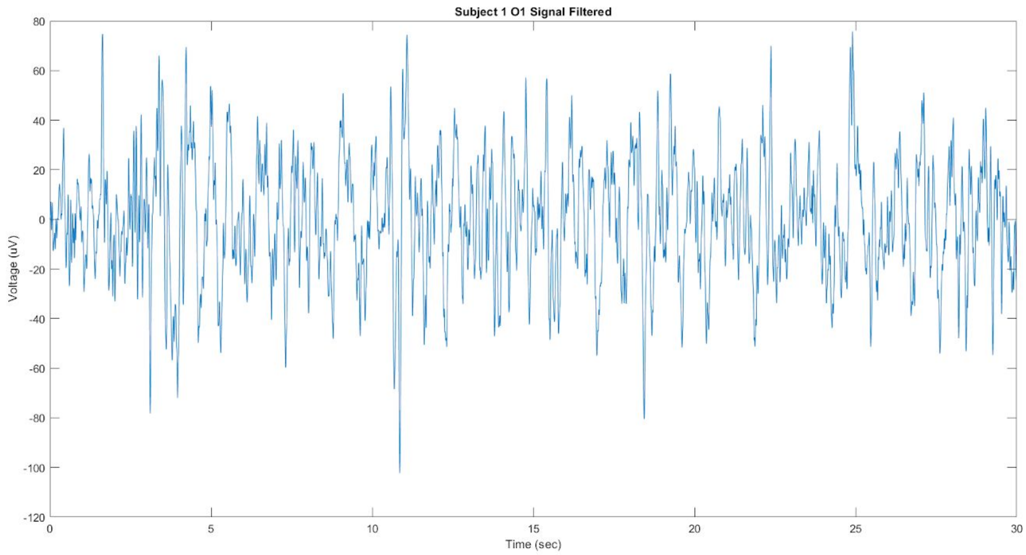

Figure 4. Plot of the filtered O1 signal from subject1 data.

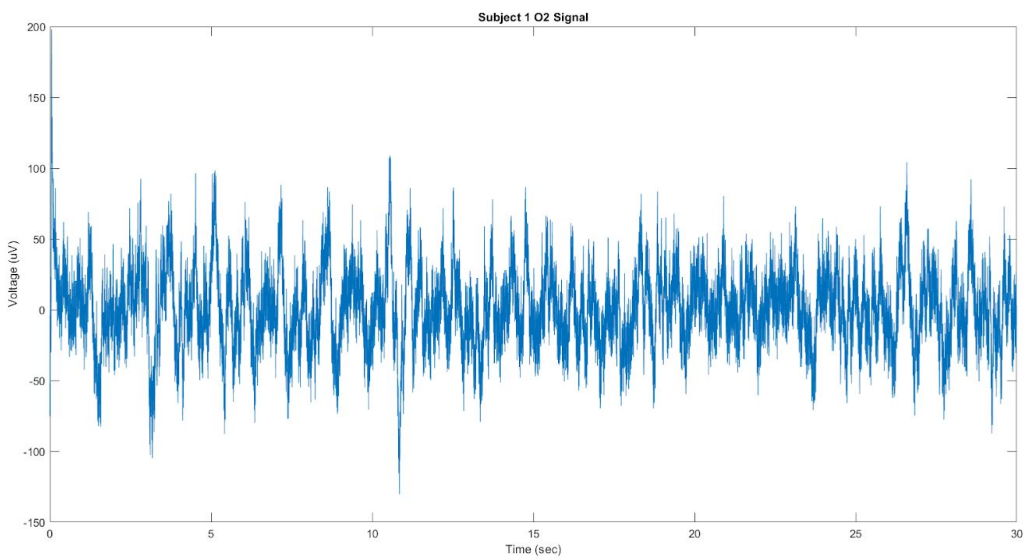

Figure 5. Plot of the unfiltered O2 signal from subject1 data.

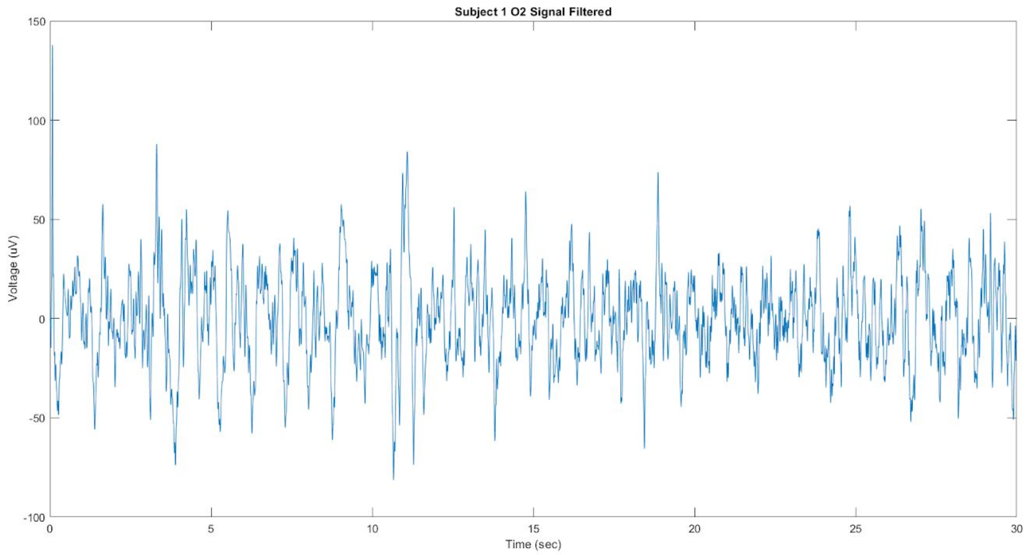

Figure 6. Plot of the filtered O2 signal from subject1 data.

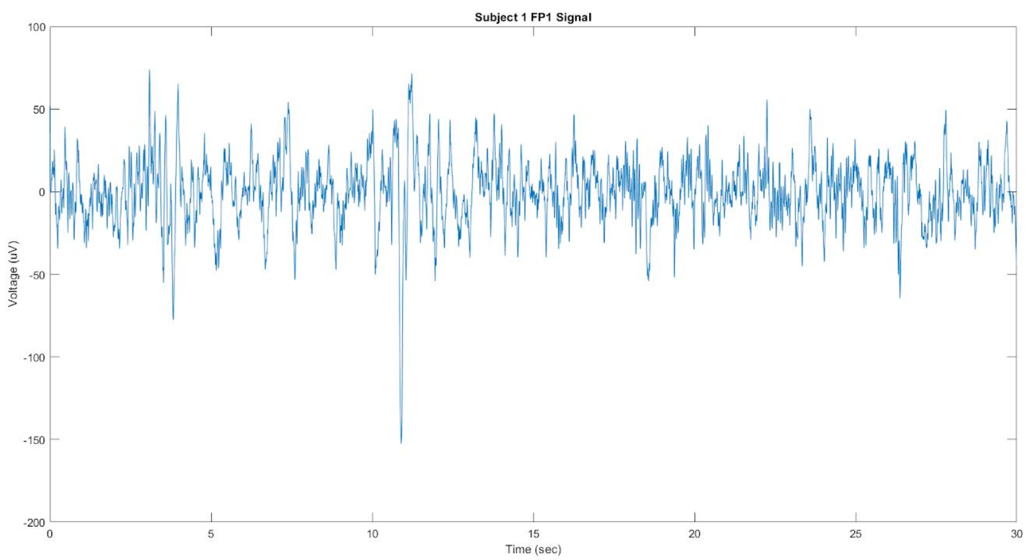

Figure 7. Plot of the unfiltered FP1 signal from subject1 data.

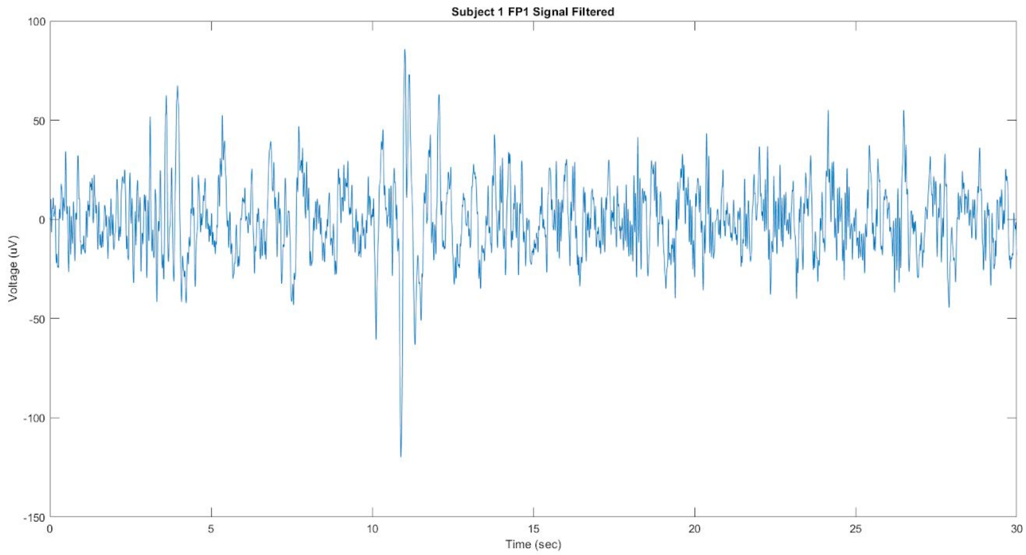

Figure 8. Plot of the filtered FP1 signal from subject1 data.

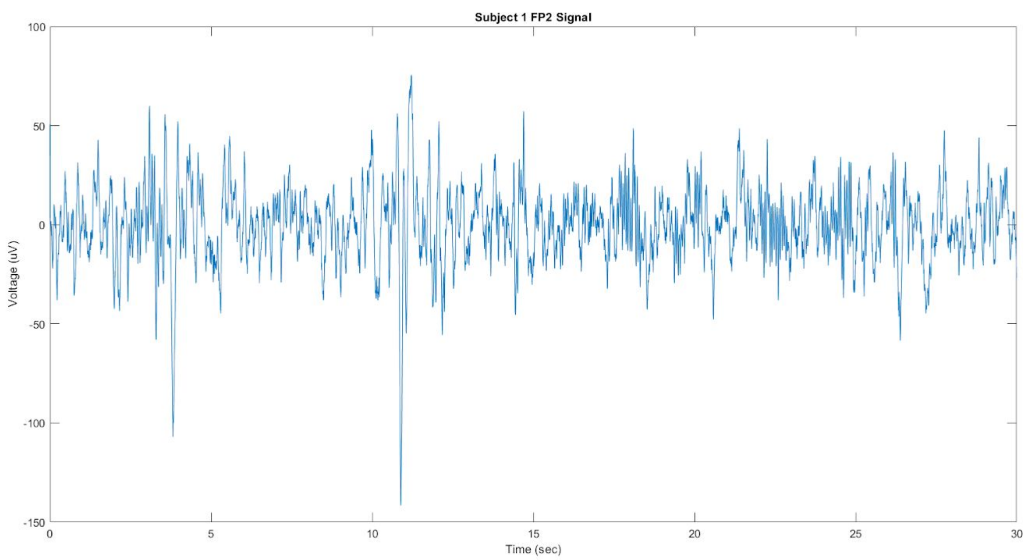

Figure 9. Plot of the unfiltered FP2 signal from subject1 data.

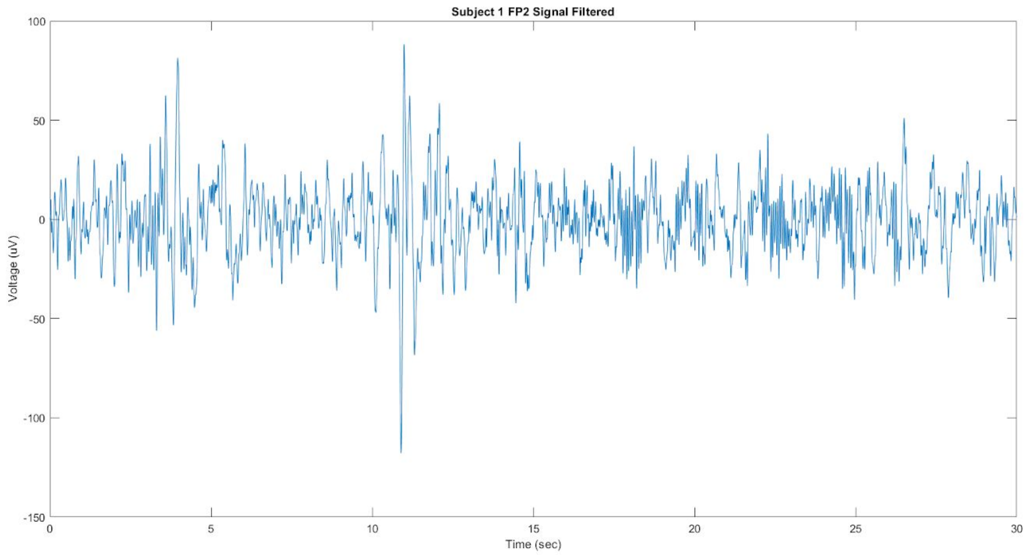

Figure 10. Plot of the filtered FP2 signal from subject1 data.

Interestingly the filter had slightly different effects on each signal, but it did complete its purpose in all four signals. When comparing all the nonfiltered plots (Figures 3,5,7,9) we can see that they are much more dense/look darker in areas (most evident in Figure 5.). When the filter is applied (Figures 4,6,8,10) the signal is less dense. Less dense lines means that certain frequencies have been removed from the signal. Another difference which can be seen are some missing peaks, which are most evident around 4 seconds in each plot.

3) These plots have been normalized.

a)

The first filter is for the alpha wave. The characteristics of the alpha wave is that it has a voltage range of 2-100 uV and a frequency range of 8-13 Hz. We used a tenth order Chebyshev Type I bandpass filter and used the frequency range for our cutoff frequencies.Therefore, the lower cutoff frequency is 8Hz and the upper cutoff frequency is 13Hz. The sampling rate frequency is 256Hz so we took that into account for the filter as well.

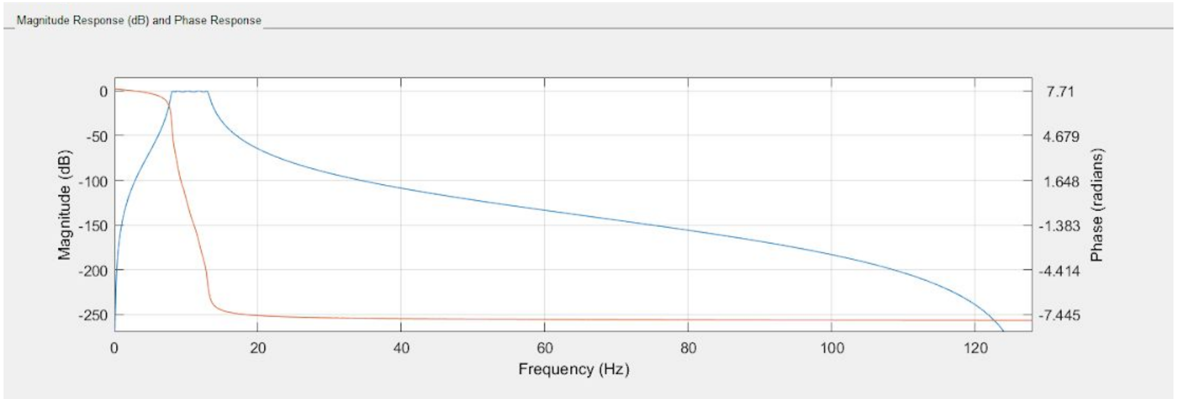

Figure 11. Plot of the magnitude and phase responses for the alpha wave filter.

The second filter is for the beta wave. The characteristics of the alpha wave is that it has a voltage range of 5-10 uV and a frequency range of 13-22 Hz. We used a tenth order Chebyshev Type I bandpass filter and used the frequency range for our cutoff frequencies.Therefore, the lower cutoff frequency is 13Hz and the upper cutoff frequency is 22Hz. The sampling rate frequency is 256Hz so we took that into account for the filter as well.

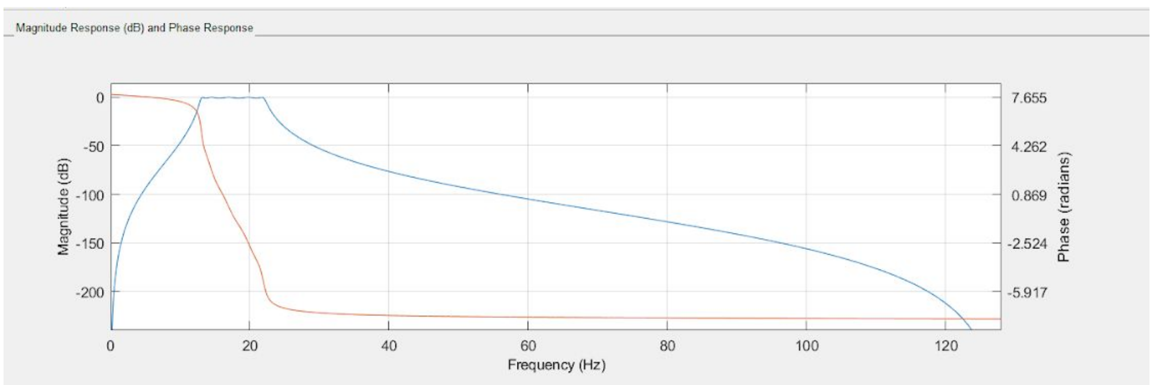

Figure 12. Plot of the magnitude and phase responses for the beta wave filter.

The third filter is for the delta wave. The characteristics of the alpha wave is that it has a voltage range of 20-100 uV and a frequency range of 0.5-4 Hz. We used a tenth order Chebyshev Type I bandpass filter and used the frequency range for our cutoff frequencies. Therefore, the lower cutoff frequency is 0.5Hz and the upper cutoff frequency is 4Hz. The sampling rate frequency is 256Hz so we took that into account for the filter as well.

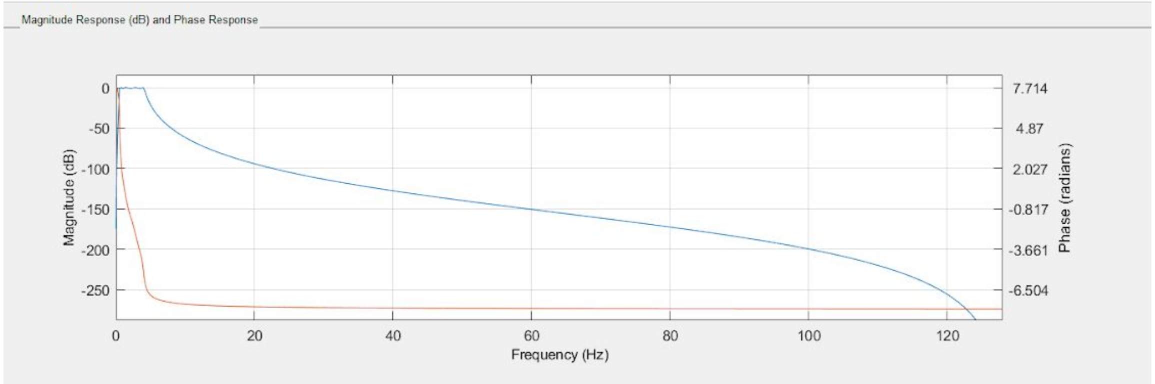

Figure 13. Plot of the magnitude and phase responses for the delta wave filter.

The fourth filter is for the theta wave. The characteristics of the alpha wave is that it has a voltage range of 10 uV and a frequency range of 4-8 Hz. We used a tenth order Chebyshev Type I bandpass filter and used the frequency range for our cutoff frequencies. Therefore, the lower cutoff frequency is 4Hz and the upper cutoff frequency is 8Hz. The sampling rate frequency is 256Hz so we took that into account for the filter as well.

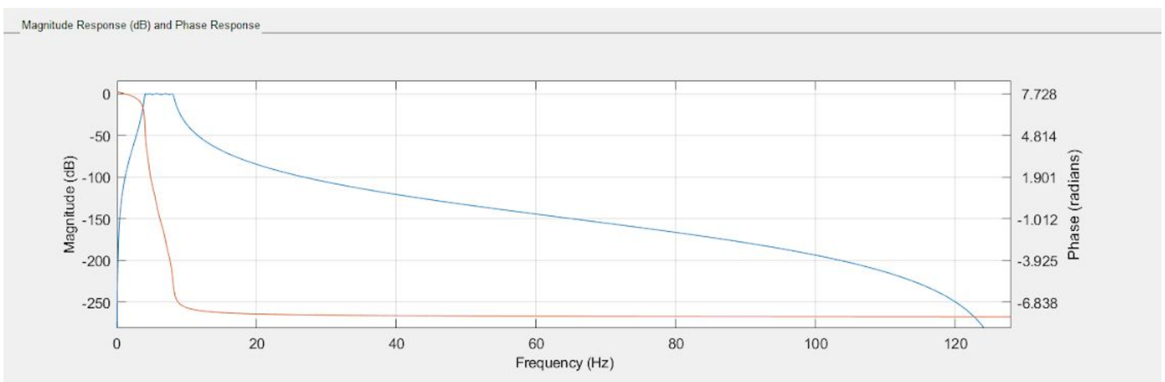

Figure 14. Plot of the magnitude and phase responses for the theta wave filter.

b) All of the parameters were carefully chosen for each of the four filters. Each had their cutoff frequencies based off of the frequency ranges given in the lab manual. These were chosen so we can have the precise areas of which the signals are shown, and subsequently this would remove all other signals which would be considered noise. The tenth order Chebyshev Type 1 bandpass filter was chosen, just like the first bandpass filter, because in the previous lab we concluded that this is the best filter for biomedical signals. It has great accuracy while maintaining the integrity of the signal.

4)

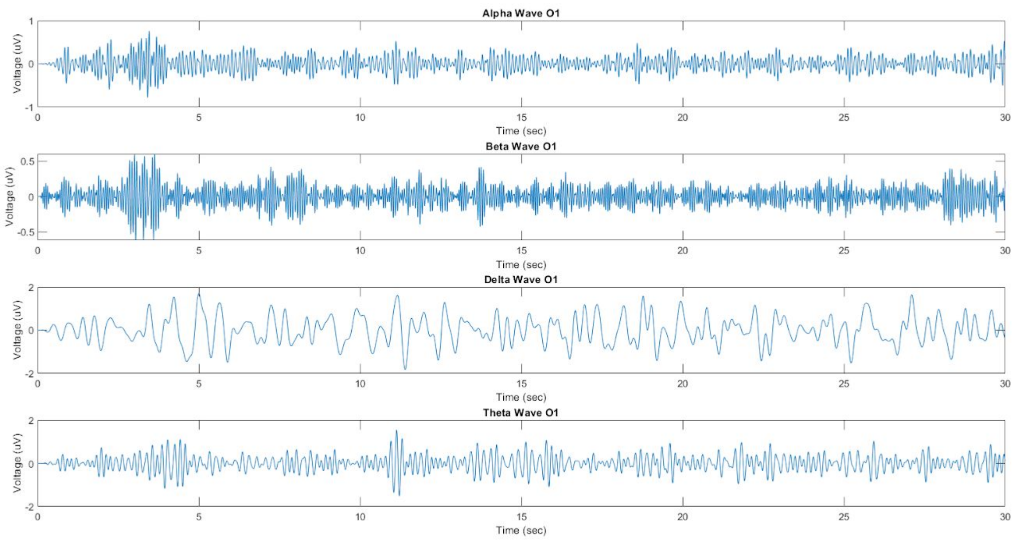

Figure 15. Plot of the alpha, beta, delta, and theta waves of the O1 signal from subject1.

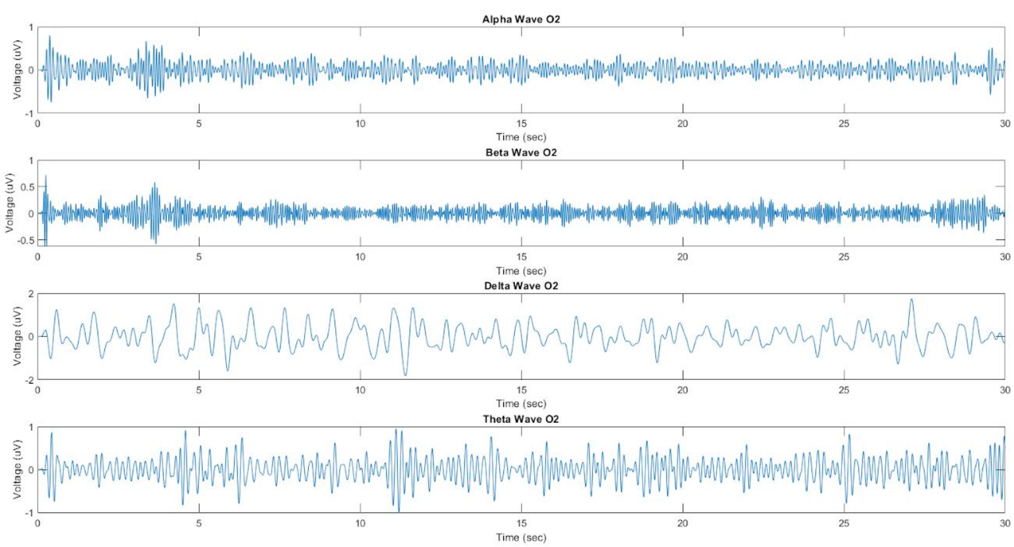

Figure 16. Plot of the alpha, beta, delta, and theta waves of the O2 signal from subject1.

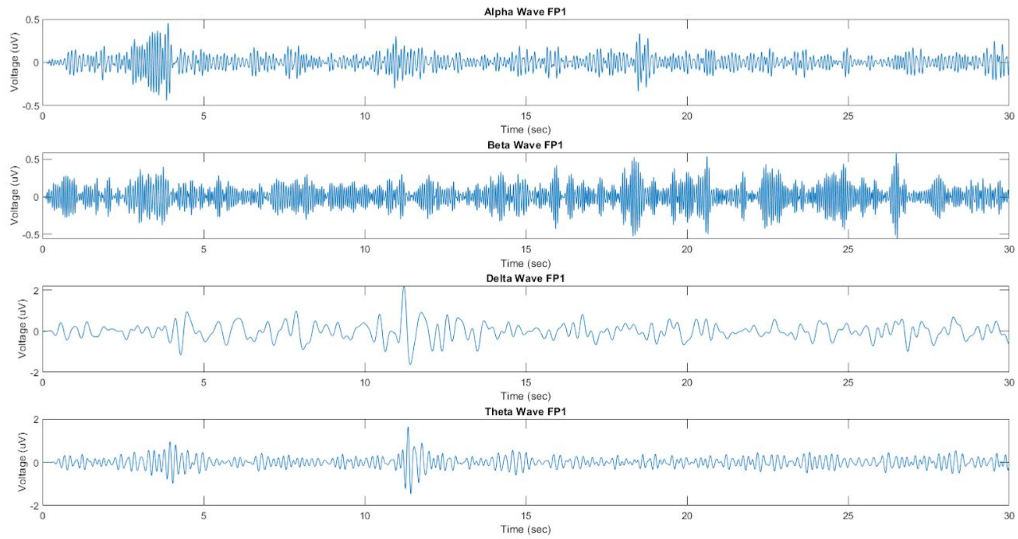

Figure 17. Plot of the alpha, beta, delta, and theta waves of the FP1 signal from subject1.

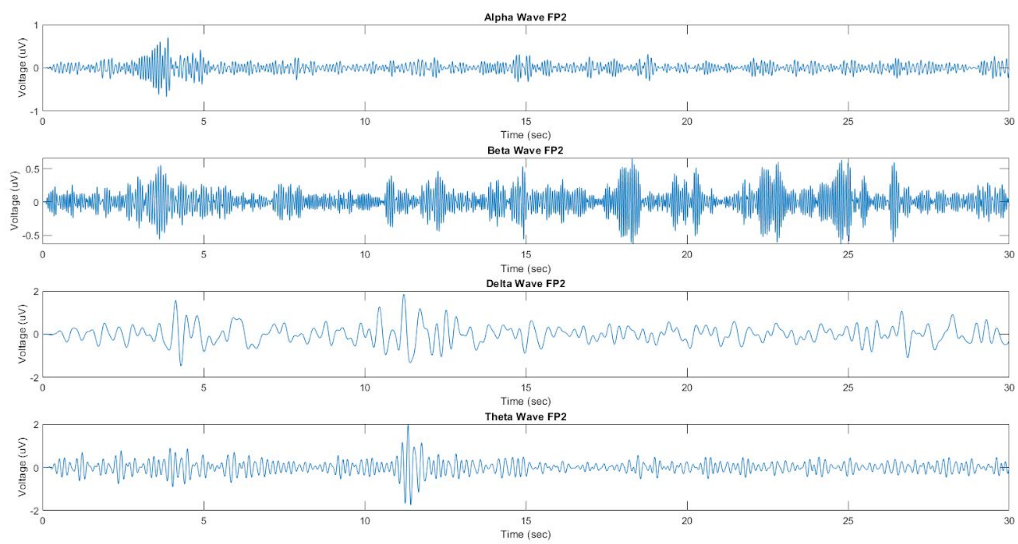

Figure 18. Plot of the alpha, beta, delta, and theta waves of the FP2 signal from subject1.

Firstly, we can see that all four of the above plots look very similar in their basic form. Secondly, we can use visual inspection to ensure that our filters are working. We know that the alpha wave is in the frequency range of (8-13Hz), beta wave in the range (13-22Hz), delta wave in the range (0.5-4Hz), and the theta wave is in the frequency range (4-8Hz). Therefore, when we look at the filtered data we should expect that the beta wave would be the most compressed wave and the delta wave being the least compressed wave. When we look at the plots this is exactly what we see. The beta waves are the most compressed waves and the theta waves are the least compressed.

5)

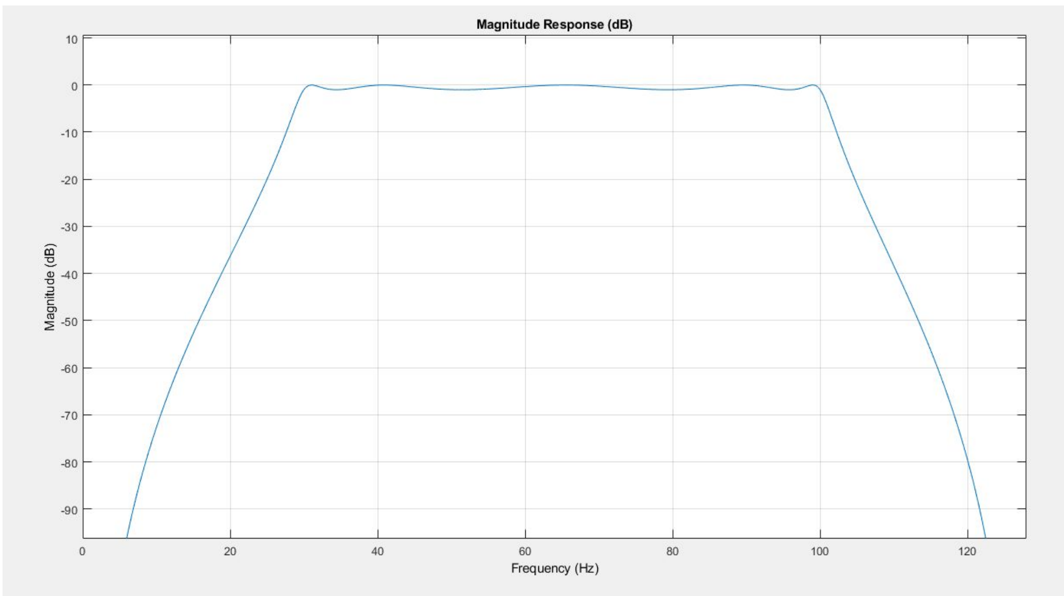

Figure 19. Plot for the magnitude plot of the noise filter.

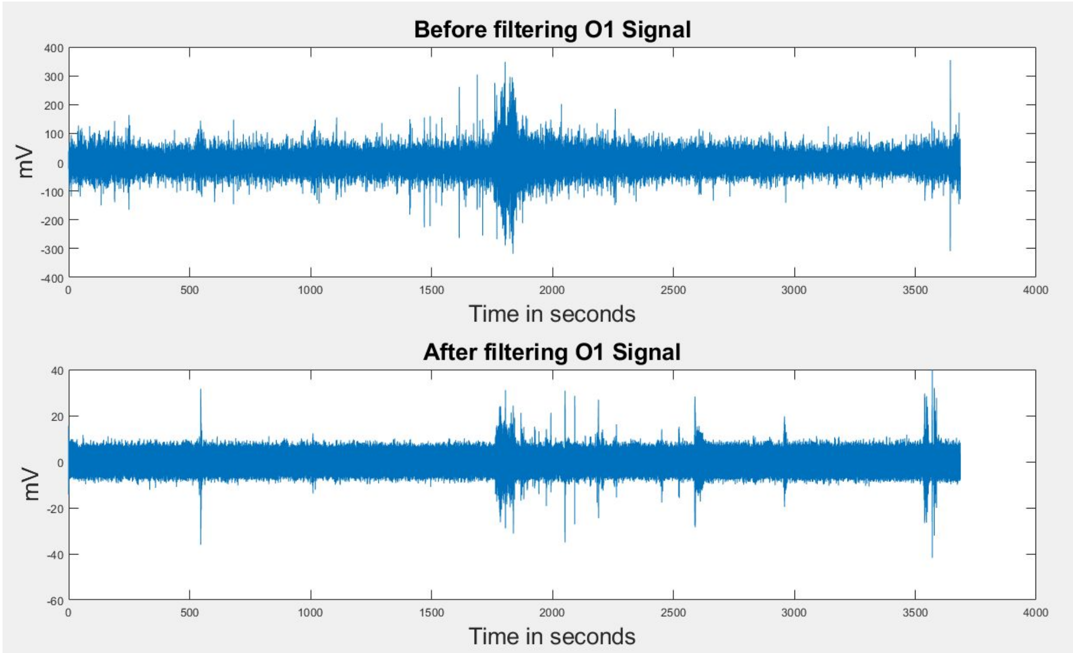

Figure 20. Plot of O1 signal before and after filtering.

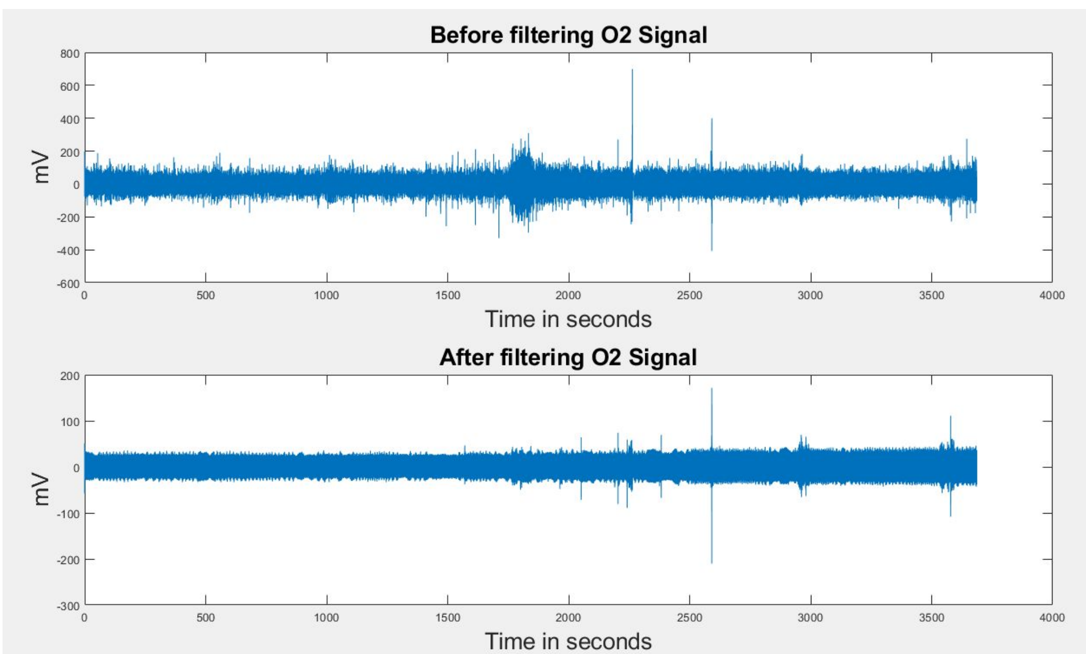

Figure 21. Plot of O2 signal before and after filtering.

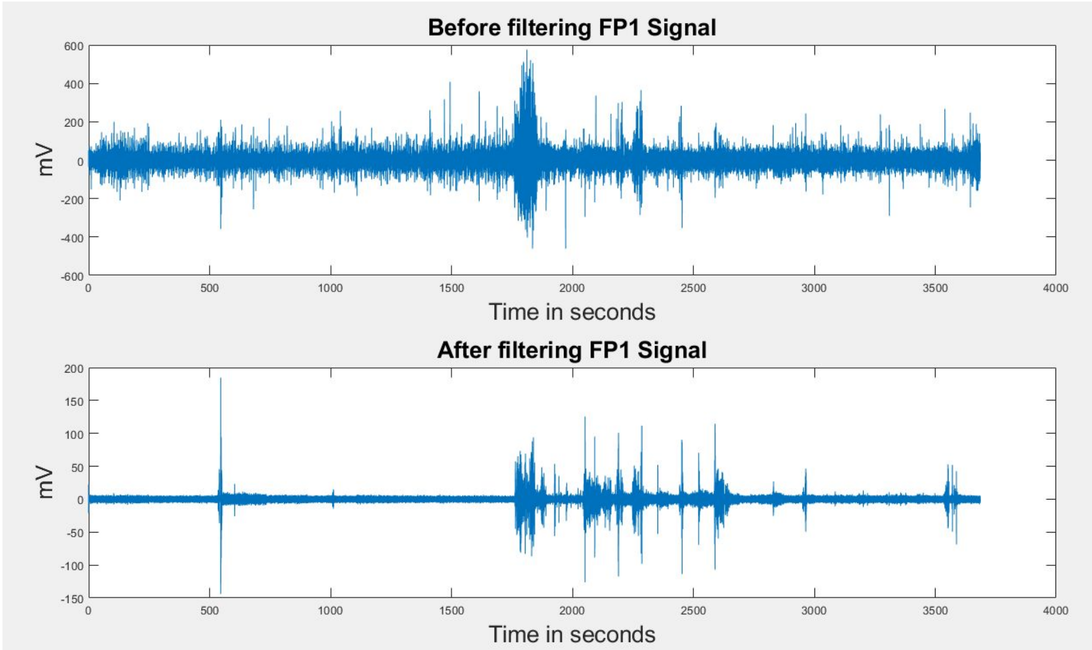

Figure 22. Plot of FP1 signal before and after filtering.

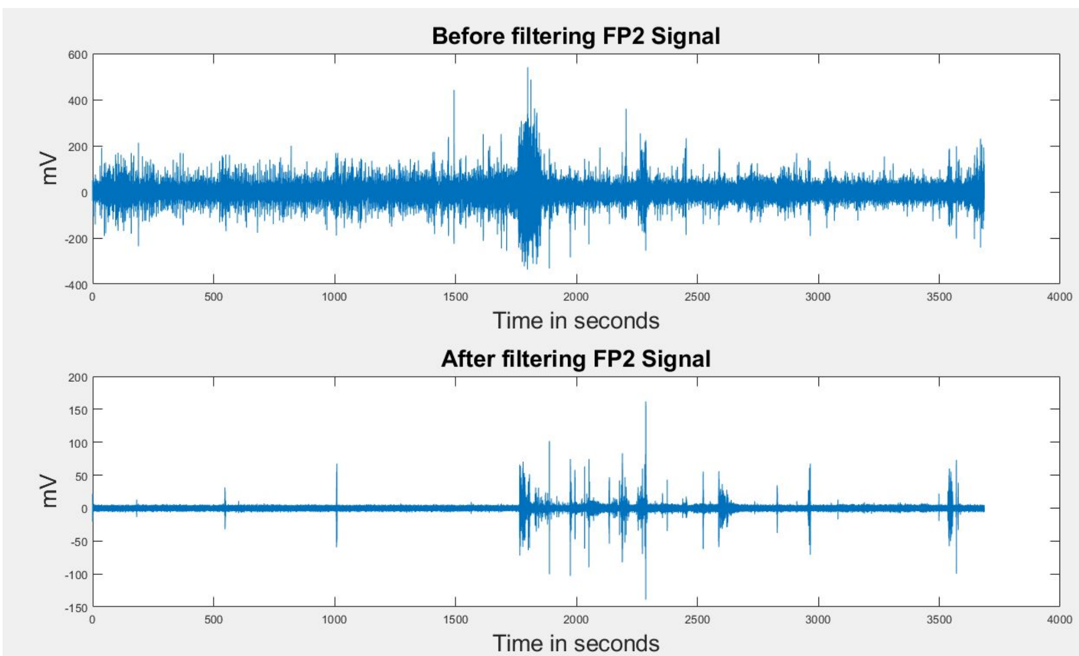

Figure 23. Plot of FP2 signal before and after filtering.

b)

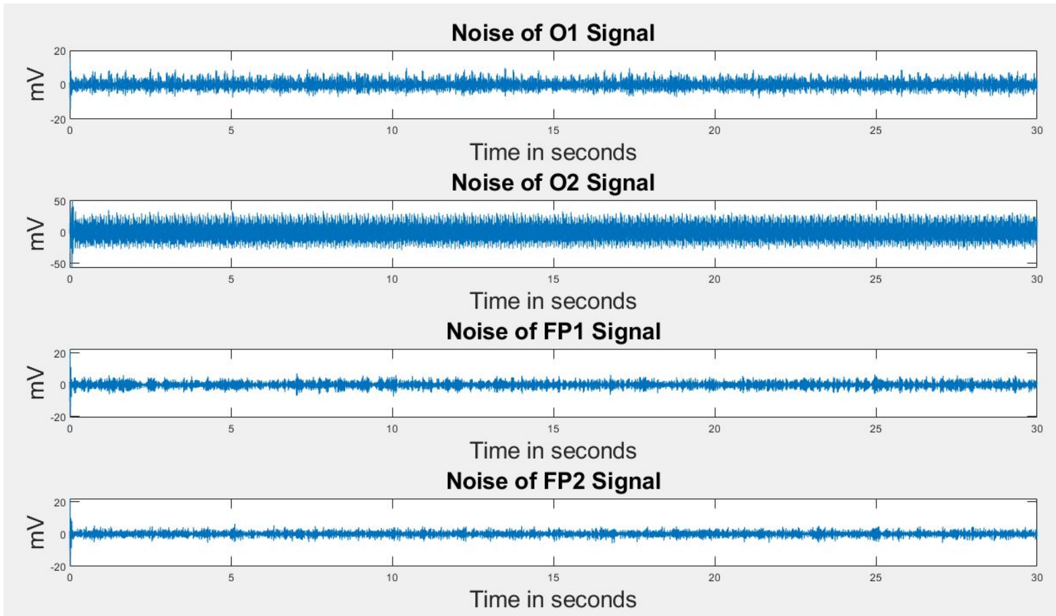

Figure 24. Plot of the noises of the four signal channels.

c) As seen above, the majority of the noise plots appear like boxes which indicate the intensity of the high frequency of the noise of the signals. For the O2 signal, the noise is high to the extent that the oscillations of the signal cannot be clearly recognised. The noise could have been from external electrical sources such as cell phones, human muscle potentials, or motion artifacts. Another crucial source of noise could have been from the setup of EEG signal acquisition where the tangling of the wires could have created a magnetic field and significantly added noise to the acquired signals. These noise signals demonstrate our denoising filter design is adequate because it is able to extract huge amounts of interference noise from the EEG signals as shown in the above blue blocks.

Conclusion:

The purpose of this lab is to examine EEG signals and process them to find the four main components of the signal. Firstly, we looked at the frequency range of a typical EEG signal. This range was used to create a denoise filter, which was used in every process of the lab. The range used for the filter was 0.5-30Hz, which was outlined as the best range in the lab manual. The bandpass filter we used was a Chebyshev 1 tenth order filter, and the cutoff frequencies were 0.5Hz and 30Hz as previously outlined. We used this filter because in the previous lab we determined it was the best for biomedical signals. With the filtered signals, we then proceeded to find the four characteristic waves of the EEG signal. Them being the alpha (8-13Hz), beta (13-22Hz), delta (0.5-4Hz), and theta (4-8Hz) waves. Similarly, Chebyshev 1 tenth order bandpass filters were used to find these waves. We examined the physical waves themselves, understanding that the higher the frequencies, the more compressed the waves would look in the plots. This was true with our plots with the beta waves being the most compressed and the delta waves being the least compressed. Finally, we looked at the noise of the signal. Knowing the range of the useful signal, we created a filter which took the waves which were outside of the range of the denoise filter and plotted it. All of the noise shown can be due to a multitude of reasons such as the wires, external interference, etc. In the end we have learned the delicate nature of biomedical signals and how crucial it is to obtain the exact frequency ranges. We have also learned the characteristics of an EEG signal, and the different waveforms which are contained in them.

References:

[1] Khademi, April . "Lab Manual" BME 632 - Signals and System II, 17 March 2020, Ryerson University

[2] Khademi, April . "Lab Experiment" BME 632 - Signals and System II, 17 March 2020, Ryerson University

[3]R. Green and B.P. Lathi, *Linear Systems and Signals*: Oxford University Press,2018.

Appendix:

MAIN CODE

```
%% Retrieving Data
load('EEG_sub1.mat');
O1sub1 = EEG(8).ch;
O2sub1 = EEG(12).ch;
FP1sub1 = EEG(5).ch;
FP2sub1 = EEG(9).ch;

time = length(O1sub1);
time = 0 : time - 1;
time = time * 0.004;

%% B1

denoise = EEGBandpassFilterDenoise;

%% B2

% Normalize

normO1sub1 = normalize(O1sub1);
normO2sub1 = normalize(O2sub1);
normFP1sub1 = normalize(FP1sub1);
normFP2sub1 = normalize(FP2sub1);


% Denoise
filteredO1sub1 = filter(denoise,normO1sub1);
filteredO2sub1 = filter(denoise,normO2sub1);
filteredFP1sub1 = filter(denoise,normFP1sub1);
filteredFP2sub1 = filter(denoise,normFP2sub1);
%uncomment above when done part B2
% filteredO1sub1 = filter(denoise,O1sub1);
% filteredO2sub1 = filter(denoise,O2sub1);
% filteredFP1sub1 = filter(denoise,FP1sub1);
% filteredFP2sub1 = filter(denoise,FP2sub1);
```

```
%comment above when done part B2

%%
figure(1)
plot(time(1:7500),O1sub1(1:7500))
title('Subject 1 O1 Signal')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

figure(2)
plot(time(1:7500),filteredO1sub1(1:7500))
title('Subject 1 O1 Signal Filtered')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

figure(3)
plot(time(1:7500),O2sub1(1:7500))
title('Subject 1 O2 Signal')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

figure(4)
plot(time(1:7500),filteredO2sub1(1:7500))
title('Subject 1 O2 Signal Filtered')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

figure(5)
plot(time(1:7500),FP1sub1(1:7500))
title('Subject 1 FP1 Signal')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

figure(6)
plot(time(1:7500),filteredFP1sub1(1:7500))
title('Subject 1 FP1 Signal Filtered')
ylabel('Voltage (uV)')
xlabel('Time (sec)')
```

```
figure(7)
plot(time(1:7500),FP2sub1(1:7500))
title('Subject 1 FP2 Signal')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

figure(8)
plot(time(1:7500),filteredFP2sub1(1:7500))
title('Subject 1 FP2 Signal Filtered')
ylabel('Voltage (uV)')
xlabel('Time (sec)')

%% B3

% Wave Exstraction

alpha = alphaWave;
beta = betaWave;
delta = deltaWave;
theta = thetaWave;

alphaO1sub1 = filter(alpha,filteredO1sub1);
betaO1sub1 = filter(beta,filteredO1sub1);
deltaO1sub1 = filter(delta,filteredO1sub1);
thetaO1sub1 = filter(theta,filteredO1sub1);

alphaO2sub1 = filter(alpha,filteredO2sub1);
betaO2sub1 = filter(beta,filteredO2sub1);
deltaO2sub1 = filter(delta,filteredO2sub1);
thetaO2sub1 = filter(theta,filteredO2sub1);

alphaFP1sub1 = filter(alpha,filteredFP1sub1);
betaFP1sub1 = filter(beta,filteredFP1sub1);
deltaFP1sub1 = filter(delta,filteredFP1sub1);
thetaFP1sub1 = filter(theta,filteredFP1sub1);

alphaFP2sub1 = filter(alpha,filteredFP2sub1);
betaFP2sub1 = filter(beta,filteredFP2sub1);
deltaFP2sub1 = filter(delta,filteredFP2sub1);
```

```
thetaFP2sub1 = filter(theta,filteredFP2sub1);

%% Looking at filtered signal

figure(97)
subplot(411)
plot(time(1:7500),alphaO1sub1(1:7500))
title('Alpha Wave O1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(412)
plot(time(1:7500),betaO1sub1(1:7500))
title('Beta Wave O1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(413)
plot(time(1:7500),deltaO1sub1(1:7500))
title('Delta Wave O1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(414)
plot(time(1:7500),thetaO1sub1(1:7500))
title('Theta Wave O1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')

figure(98)
subplot(411)
plot(time(1:7500),alphaO2sub1(1:7500))
title('Alpha Wave O2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(412)
plot(time(1:7500),betaO2sub1(1:7500))
title('Beta Wave O2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(413)
plot(time(1:7500),deltaO2sub1(1:7500))
```

```
title('Delta Wave O2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(414)
plot(time(1:7500),thetaO2sub1(1:7500))
title('Theta Wave O2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')

figure(99)
subplot(411)
plot(time(1:7500),alphaFP1sub1(1:7500))
title('Alpha Wave FP1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(412)
plot(time(1:7500),betaFP1sub1(1:7500))
title('Beta Wave FP1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(413)
plot(time(1:7500),deltaFP1sub1(1:7500))
title('Delta Wave FP1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(414)
plot(time(1:7500),thetaFP1sub1(1:7500))
title('Theta Wave FP1')
xlabel('Time (sec)')
ylabel('Voltage (uV)')

figure(100)
subplot(411)
plot(time(1:7500),alphaFP2sub1(1:7500))
title('Alpha Wave FP2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(412)
plot(time(1:7500),betaFP2sub1(1:7500))
```

```
title('Beta Wave FP2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(413)
plot(time(1:7500),deltaFP2sub1(1:7500))
title('Delta Wave FP2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
subplot(414)
plot(time(1:7500),thetaFP2sub1(1:7500))
title('Theta Wave FP2')
xlabel('Time (sec)')
ylabel('Voltage (uV)')
```

DENOISE BANDPASS FILTER

```
function Hd = EEGBandpassFilterDenoise
%EEGBANDPASSFILTERDENOISE Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.7 and Signal Processing Toolbox 8.3.
% Generated on: 28-Mar-2020 16:56:17

% Chebyshev Type I Bandpass filter designed using FDESIGN.BANDPASS.

% All frequency values are in Hz.
Fs = 256;  % Sampling Frequency

N      = 10;   % Order
Fpass1 = 0.5;  % First Passband Frequency
Fpass2 = 30;   % Second Passband Frequency
Apass  = 1;    % Passband Ripple (dB)

% Construct an FDESIGN object and call its CHEBY1 method.
h  = fdesign.bandpass('N,Fp1,Fp2,Ap', N, Fpass1, Fpass2, Apass, Fs);
Hd = design(h, 'cheby1');

% [EOF]
```

ALPHA WAVE BANDPASS FILTER

```
function Hd = alphaWave
%ALPHAWAVE Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.7 and Signal Processing Toolbox 8.3.
% Generated on: 29-Mar-2020 14:20:49

% Chebyshev Type I Bandpass filter designed using FDESIGN.BANDPASS.

% All frequency values are in Hz.
Fs = 256;  % Sampling Frequency

N      = 10;  % Order
Fpass1 = 8;   % First Passband Frequency
Fpass2 = 13;  % Second Passband Frequency
Apass  = 1;   % Passband Ripple (dB)

% Construct an FDESIGN object and call its CHEBY1 method.
h  = fdesign.bandpass('N,Fp1,Fp2,Ap', N, Fpass1, Fpass2, Apass, Fs);
Hd = design(h, 'cheby1');

% [EOF]
```

BETA WAVE BANDPASS FILTER

```
function Hd = betaWave
%BETAWAVE Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.7 and Signal Processing Toolbox 8.3.
% Generated on: 29-Mar-2020 14:52:57

% Chebyshev Type I Bandpass filter designed using FDESIGN.BANDPASS.
```

```
% All frequency values are in Hz.
Fs = 256;  % Sampling Frequency

N      = 10;  % Order
Fpass1 = 13;  % First Passband Frequency
Fpass2 = 22;  % Second Passband Frequency
Apass  = 1;   % Passband Ripple (dB)

% Construct an FDESIGN object and call its CHEBY1 method.
h  = fdesign.bandpass('N,Fp1,Fp2,Ap', N, Fpass1, Fpass2, Apass, Fs);
Hd = design(h, 'cheby1');

% [EOF]
```

DELTA WAVE BANDPASS FILTER

```
function Hd = deltaWave
%DELTAWAVE Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.7 and Signal Processing Toolbox 8.3.
% Generated on: 29-Mar-2020 14:53:59

% Chebyshev Type I Bandpass filter designed using FDESIGN.BANDPASS.

% All frequency values are in Hz.
Fs = 256;  % Sampling Frequency

N      = 10;   % Order
Fpass1 = 0.5;  % First Passband Frequency
Fpass2 = 4;    % Second Passband Frequency
Apass  = 1;    % Passband Ripple (dB)

% Construct an FDESIGN object and call its CHEBY1 method.
h  = fdesign.bandpass('N,Fp1,Fp2,Ap', N, Fpass1, Fpass2, Apass, Fs);
Hd = design(h, 'cheby1');

% [EOF]
```

THETA WAVE BANDPASS FILTER

```
function Hd = thetaWave
%THETAWAVE Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.7 and Signal Processing Toolbox 8.3.
% Generated on: 29-Mar-2020 14:54:47

% Chebyshev Type I Bandpass filter designed using FDESIGN.BANDPASS.

% All frequency values are in Hz.
Fs = 256;  % Sampling Frequency

N      = 10;  % Order
Fpass1 = 4;   % First Passband Frequency
Fpass2 = 8;   % Second Passband Frequency
Apass  = 1;   % Passband Ripple (dB)

% Construct an FDESIGN object and call its CHEBY1 method.
h  = fdesign.bandpass('N,Fp1,Fp2,Ap', N, Fpass1, Fpass2, Apass, Fs);
Hd = design(h, 'cheby1');

% [EOF]
```

% Part 5

```
% 5a.

%% Retrieving Data
load('EEG_sub1.mat');
O1sub1 = EEG(8).ch;
O2sub1 = EEG(12).ch;
FP1sub1 = EEG(5).ch;
FP2sub1 = EEG(9).ch;

time = length(O1sub1); % frequency domain conversion for O1 signal
```

```
time = 0 : time - 1;
time = time * 0.004;

time2 = length(O1sub1);  % frequency domain conversion for O2 signal
time2 = 0 : time2 - 1;
time2 = time2 * 0.004;

time3 = length(O1sub1); % frequency domain conversion for FP1 signal
time3 = 0 : time3 - 1;
time3 = time3 * 0.004;

time4 = length(O1sub1); % frequency domain conversion for FP2 signal
time4 = 0 : time4 - 1;
time4 = time4 * 0.004;

%CHEBYSHEV FILTERS ORDER 10, WITH FREQUENCY 30-100Hz applied
[b,a] = sos2tf(SOS, G);          % filters applied to OP1 signal
o1ecgcheby = filter(b,a,O1sub1);

[b,a] = sos2tf(SOS, G);          % filters applied to OP2 signal
o2ecgcheby = filter(b,a,O2sub1);

[b,a] = sos2tf(SOS, G);          % filters applied to FP1 signal
fp1ecgcheby = filter(b,a,FP1sub1);

[b,a] = sos2tf(SOS, G);          % filters applied to FP2 signal
fp2ecgcheby = filter(b,a,FP2sub1);

% OP1 PLOT BEFORE AND AFTER
figure(1)
subplot(2,1,1); plot(time, O1sub1); title('Before filtering O1 Signal','FontSize', 20);xlabel('Time
in seconds','FontSize', 20);ylabel('mV','FontSize', 20);
subplot(2,1,2); plot(time,ecgcheby); title('After filtering O1 Signal','FontSize', 20);xlabel('Time
in seconds','FontSize', 20); ylabel('mV','FontSize', 20);

% OP2 PLOT BEFORE AND AFTER
figure(2)
subplot(2,1,1); plot(time2, O2sub1); title('Before filtering O2 Signal','FontSize', 20);xlabel('Time
in seconds','FontSize', 20);ylabel('mV','FontSize', 20);
```

```
subplot(2,1,2); plot(time2,o2ecgcheby); title('After filtering O2 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);

%  FP1 PLOT BEFORE AND AFTER
figure(3)
subplot(2,1,1); plot(time3, FP1sub1); title('Before filtering FP1 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20);ylabel('mV','FontSize', 20);
subplot(2,1,2); plot(time3,fp1ecgcheby); title('After filtering FP1 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);

%  FP2 PLOT BEFORE AND AFTER
figure(4)
subplot(2,1,1); plot(time4, FP2sub1); title('Before filtering FP2 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20);ylabel('mV','FontSize', 20);
subplot(2,1,2); plot(time4,fp2ecgcheby); title('After filtering FP2 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);


%5c

%% Retrieving Data
load('EEG_sub1.mat');
O1sub1 = EEG(8).ch;
O2sub1 = EEG(12).ch;
FP1sub1 = EEG(5).ch;
FP2sub1 = EEG(9).ch;

time = length(O1sub1); % frequency domain conversion for O1 signal
time = 0 : time - 1;
time = time * 0.004;

time2 = length(O1sub1);  % frequency domain conversion for O2 signal
time2 = 0 : time2 - 1;
time2 = time2 * 0.004;

time3 = length(O1sub1); % frequency domain conversion for FP1 signal
time3 = 0 : time3 - 1;
time3 = time3 * 0.004;
```

```
time4 = length(O1sub1); % frequency domain conversion for FP2 signal
time4 = 0 : time4 - 1;
time4 = time4 * 0.004;



[b,a] = sos2tf(SOS, G);
o1ecgcheby = filter(b,a,O1sub1);


[b,a] = sos2tf(SOS, G);
o2ecgcheby = filter(b,a,O2sub1);

[b,a] = sos2tf(SOS, G);
fp1ecgcheby = filter(b,a,FP1sub1);

[b,a] = sos2tf(SOS, G);
fp2ecgcheby = filter(b,a,FP2sub1);


figure(1)
subplot(4,1,1); plot(time(1:7500),ecgcheby(1:7500)); title('Noise of O1 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);
subplot(4,1,2); plot(time2(1:7500),o2ecgcheby(1:7500)); title('Noise of O2 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);
subplot(4,1,3); plot(time3(1:7500),fp1ecgcheby(1:7500)); title('Noise of FP1 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);
subplot(4,1,4); plot(time4(1:7500),fp2ecgcheby(1:7500)); title('Noise of FP2 Signal','FontSize',
20);xlabel('Time in seconds','FontSize', 20); ylabel('mV','FontSize', 20);
```