Birzeit University

Faculty of Engineering and Technology
Electrical and Computer Engineering Department
DIGITAL SIGNAL PROCESSING (DSP)
ENCS4310

Course Project:
Filtering a real electrocardiographic signal

Prepared by:
Jenin Mansour 1200540
Shahd Yahya 1210249

Instructor: Dr. Qadri Mayyala

Date of submission: 3/6/2024

2023-2024

Abstract

This project focuses on the processing of raw ECG signals using digital filters to enhance signal quality and facilitate accurate diagnosis. The primary objective is to implement and evaluate the effectiveness of high-pass and low-pass filters in removing noise and preserving essential signal components. High-pass filters are employed to eliminate low-frequency noise, such as baseline wander, while low-pass filters are used to attenuate high-frequency noise, such as muscle artifacts and electrical interference.

Table of Contents

Table of Figures

Theory

Electrocardiography (ECG) is a critical tool in medical diagnostics, providing valuable insights into the electrical activity of the heart. The precise analysis of ECG signals is essential for detecting various cardiac abnormalities, such as arrhythmias and myocardial infarctions. However, raw ECG signals are often contaminated with noise from various sources, including baseline wander, power line interference, and muscle artifacts, which can obscure clinically significant information [1].

Baseline wander is a low frequency artifact in an ECG signal that often arises from different factors, including poor skin preparation, respiration, changes in electrodes impedance, and motion. It can mask important information from many signals, and if it is not properly removed, crucial diagnostic information contained in the signals of interest will be lost or corrupted [1].

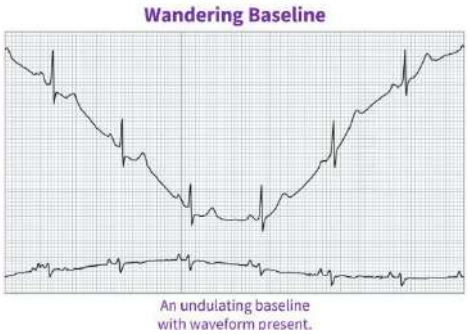

Figure 1: Baseline Wander [1]

The shape of ECG Signal

The shape of an ECG (electrocardiogram) signal is characterized by a series of waves that represent the electrical activity of the heart. The main components include the P wave, which corresponds to atrial depolarization; the QRS complex, which represents ventricular depolarization and is typically the most prominent feature; and the T wave, which reflects ventricular repolarization. The signal often starts with a small P wave, followed by a sharp rise and fall forming the QRS complex, and concludes with a broader T wave. The shape and intervals of these waves are crucial for diagnosing various cardiac conditions, as deviations from the normal pattern can indicate abnormalities such as arrhythmias, myocardial infarction, or electrolyte imbalances[2].

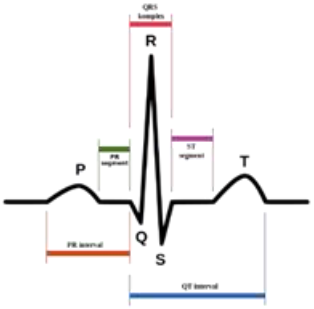

Figure 2 :The shape of ECG Signal [2]

Part 1: Data Visualization

1.1 Insert the real-time column for each signal.

We inserted the real time column based on the sampling frequency as shown:

$$Real\ time = \frac{Sample\ number}{Sampling\ frequency}$$

Creating a real-time column from the sample numbers using the known sampling frequency allows us to verify the accuracy and consistency of the `temps` column since we noticed that the temps column has duplicate values. This can be useful to ensure that there are no missing or duplicate time values, which could indicate issues in data acquisition or preprocessing.

1.2 Display each of these signals (reduce the width of the display lines for better readability).

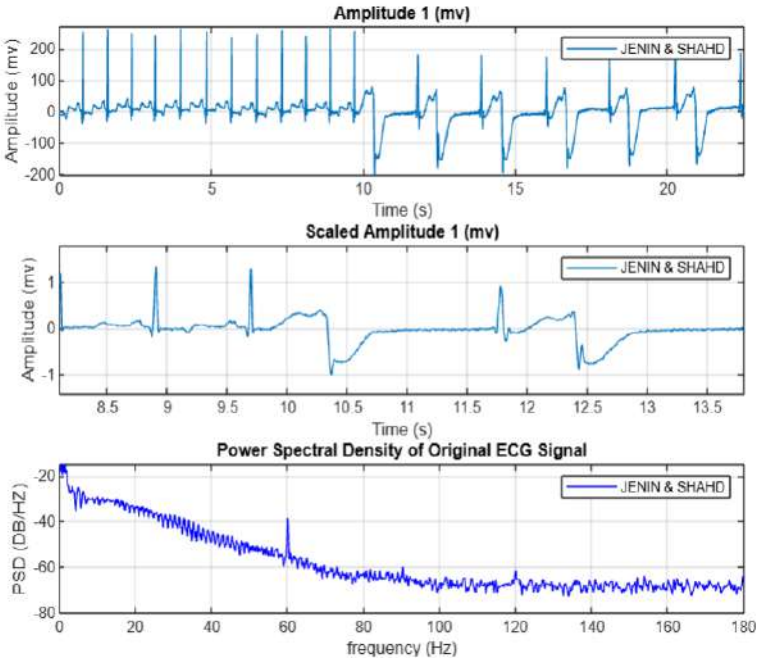

Figure 4: ECG1

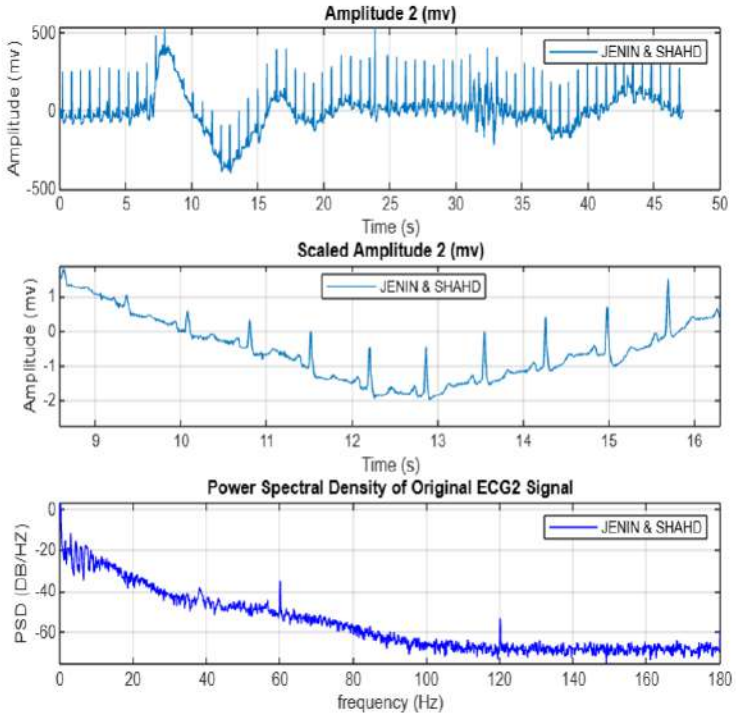

Figure 3: ECG2

Based on what is shown in Figure 4 about ECG1:

The first Subplot shows the representation of the raw data without any modifications, while in the second Subplot, I adjusted the width and reduced the length to display the signal details more clearly.

What we can observe in the second Subplot is the shape of the ECG signal, which contains P, QRS, and T waves. It also shows the impact of low-frequency interference, known as baseline wander, as well as the effect of high-frequency noise on the signal.

Finally, in the last Subplot, we represented the signal in the frequency domain. The impact of both high and low frequencies on the signal is clearly visible, as illustrated below.

Low Frequency effect (Baseline Wander)

high-pass filter with a cutoff frequency around 0.5 Hz is used to eliminate this noise.

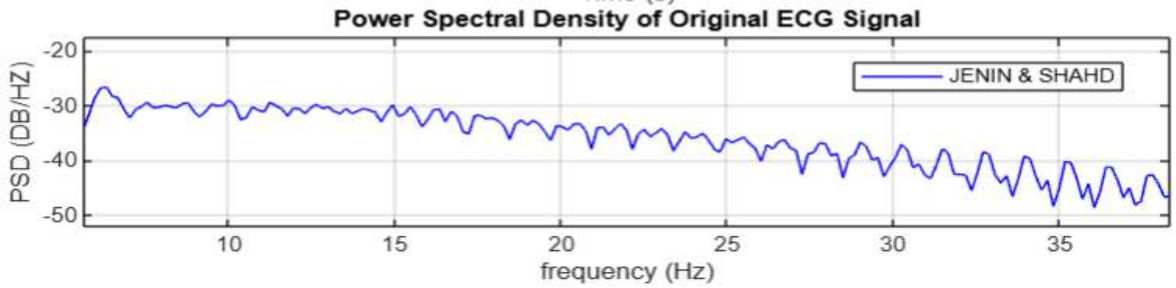

Figure 5: Baseline Wander effect

Power line interference at 60Hz

A notch filter at 60 Hz (or 50 Hz) is used to eliminate this interference.

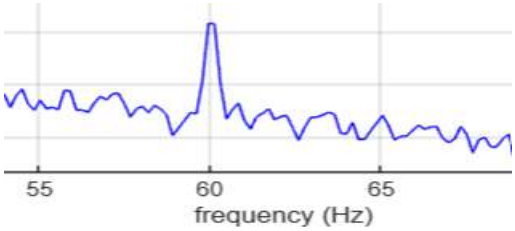

Figure 6:Power line interference

High frequency noise

To remove this noise, a low-pass filter with a cutoff frequency around 150 Hz is used.

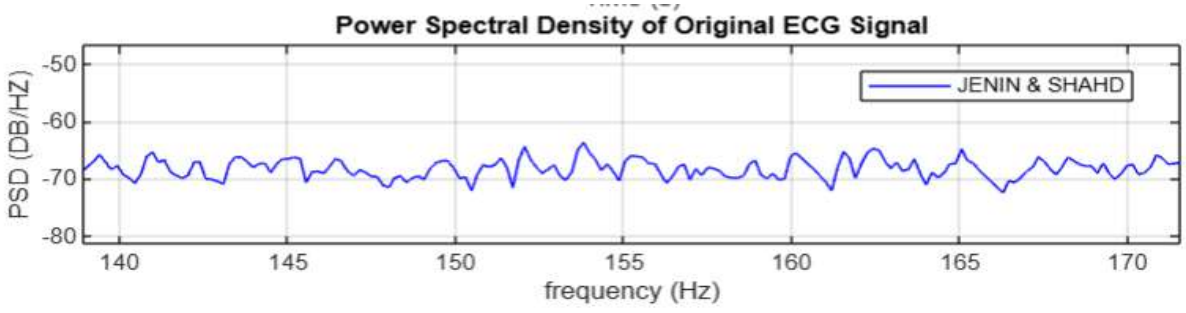

Figure 7:High frequency noise

1.3 What filtering types are necessary to improve the readability of data and make automatic processing possible?

To improve the readability of ECG data and enable accurate automatic processing, it is good practice to use bandpass filters. ECG frequencies typically fall within the range of 0.5 to 150 Hz, encompassing both the low-frequency components that reflect baseline wander and the higher-frequency components associated with muscle artifacts and electrical noise. A bandpass filter is designed to retain the frequencies within this specific range while attenuating the frequencies outside it, and for removing the power line interference that occur at 60 Hz we can use a notch filter. Also, to improve the readability of ECG data, we need to remove baseline wander and high-frequency noise. This can be achieved using a high-pass filter (to remove low-frequency baseline wander) and then using a low-pass filter (to remove high-frequency noise) [3].

Part 2: Filtering the ECG

2.1 High-pass filter

Baseline wander on an ECG can be caused by a patient's breathing If there's not a good electrical connection, this would present as a rolling of the signal on the baseline. Respiration artifacts come in below 0.5 Hz and a clinician can address them with the use of high-pass filters, which block frequencies below a specific cut point, making the QRS complexes more prominent and easier to analyze.

$$HP(n) = HP(n-1) - \frac{1}{32}X(n) + X(n-16) - X(n-17) + \frac{1}{32}X(n-32)$$

2.1.1 Calculate its transfer function.

$$\mathbf{HP(n) - HP(n-1) = -\frac{1}{32}X(n) + X(n-16) - X(n-17) + \frac{1}{32}X(n-32)}$$

$$\mathbf{HP(Z) - HP(Z)Z^{-1} = -\frac{1}{32}X(Z) + Z^{-16}X(Z) - Z^{-17}X(Z) + \frac{1}{32}Z^{-32}X(Z)}$$

$$\mathbf{HP(Z)(1 - Z^{-1}) = X(Z)(-\frac{1}{32} + Z^{-16} - Z^{-17} + \frac{1}{32}Z^{-32})}$$

$$\mathbf{H(Z) = \frac{HP(Z)}{X(Z)} = \frac{-\frac{1}{32} + Z^{-16} - Z^{-17} + \frac{1}{32}Z^{-32}}{1 - Z^{-1}}}$$

2.1.2 Find and plot the Frequency response.

$$\mathbf{H}(e^{jw}) = \frac{\boldsymbol{HP}(e^{jw})}{\boldsymbol{X}(e^{jw})} = \frac{-\frac{\mathbf{1}}{\mathbf{32}} + \boldsymbol{e}^{-j16w} - \boldsymbol{e}^{-j17w} + \frac{\mathbf{1}}{\mathbf{32}}\boldsymbol{e}^{-j32w}}{\mathbf{1} - \boldsymbol{e}^{-jw}}$$

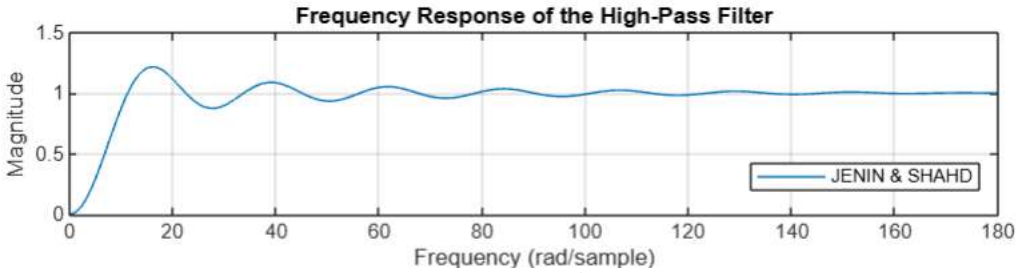

Figure 8: Frequency response of High pass Filter

Using the filter function in MATLAB with the filter coefficients, we obtained the above plot

2.1.3 Which family does it belong to (FIR, IIR)?

The high-pass filter shown in the zero-pole plot in Fig. 9 is classified as a Finite Impulse Response (FIR) filter. This conclusion is drawn from the absence of poles. FIR filters are characterized by having only zeros and no poles (except possibly at the origin), while Infinite Impulse Response (IIR) filters have both poles and zeros. Although there is a pole at z=1, it is canceled by a zero, affirming that the filter is FIR. Therefore, the zero-pole plot clearly indicates that this filter cannot be classified as an IIR filter.

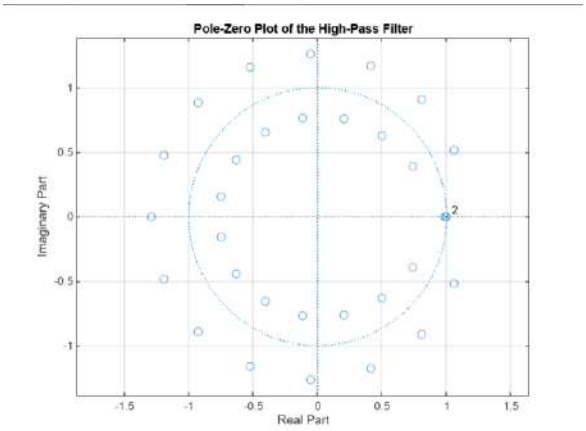

Figure 9: pole - zero plot of the high pass filter

2.1.4 Apply this filter to the ECG1 and then ECG2 signal.

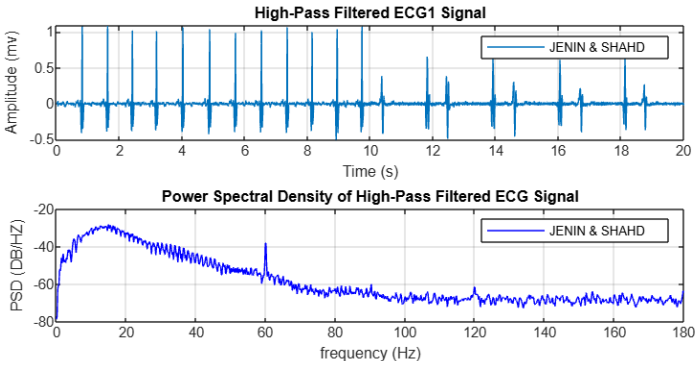

Figure 11: High pass filtered ECG1 signal

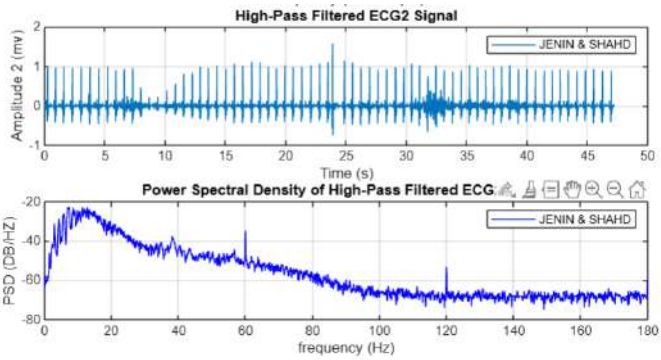

Figure 10: High pass filtered ECG2 signal

The first thing we can notice in the signal after it passes through the high-pass filter is the cancel of low-frequency effect (baseline wander). The signal is now clearly centered around zero, as shown in the first subplot of both images and illustrated below:

ECG1 Signal:

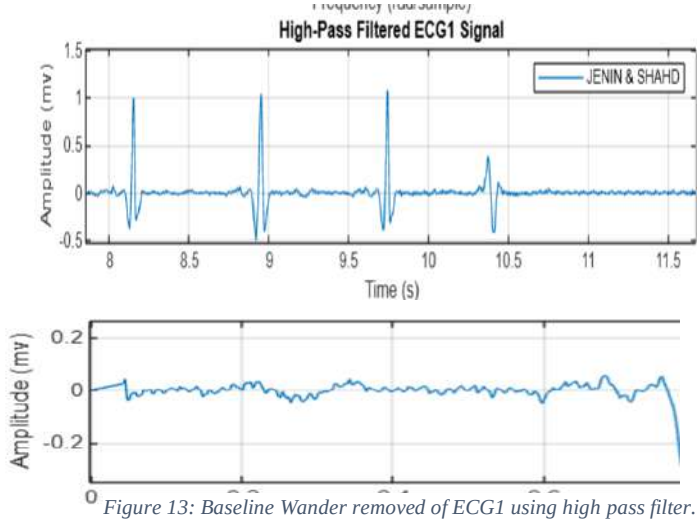

Figure 13: Baseline Wander removed of ECG1 using high pass filter.

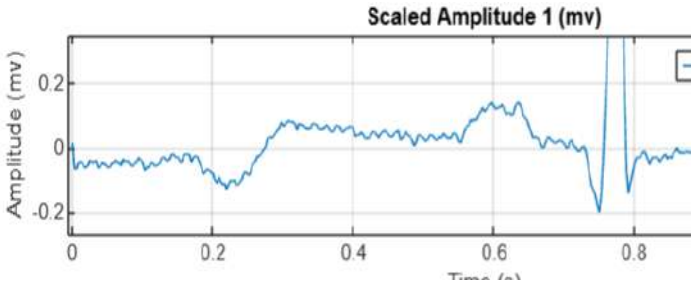

Figure 12:BaseLine Wander effect of original ECG1

ECG2 Signal:

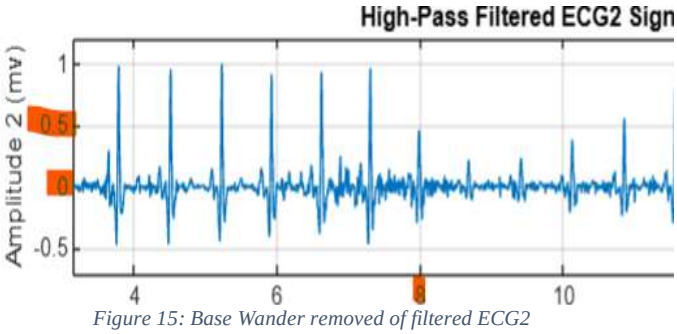

Figure 15: Base Wander removed of filtered ECG2

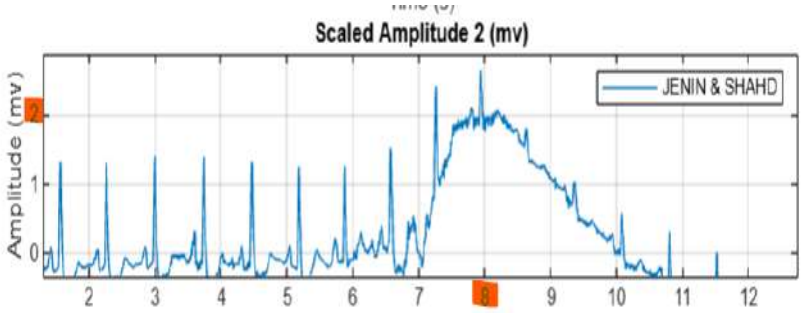

Figure 14: Base Wander effect of original ECG2

Secondly, by observing the signal in the frequency domain after it passes through the filter, compared to the signal before it passes through the filter, we can notice the change in low-frequency components below 50 Hz. Additionally, at higher frequencies, there is no significant change.

ECG1 Signal:

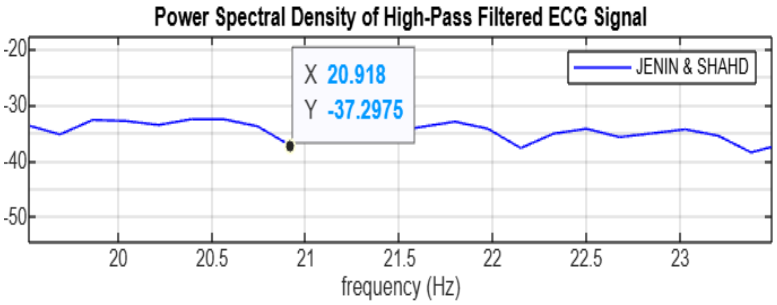

Figure 16: filtered ECG1 in frequency domain

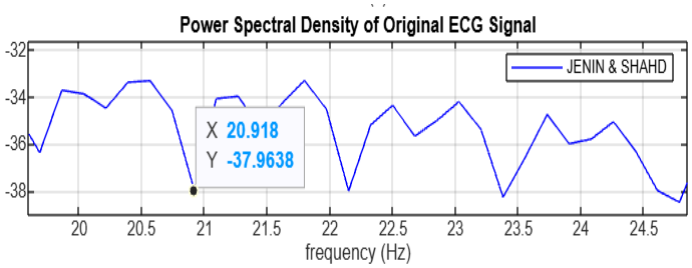

Figure 17: original ECG1 in frequency domain

ECG2 Signal:

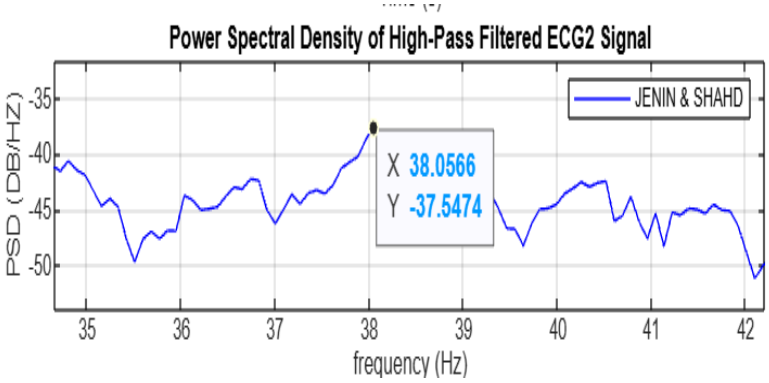

Figure 18: filtered ECG2 in frequency domain

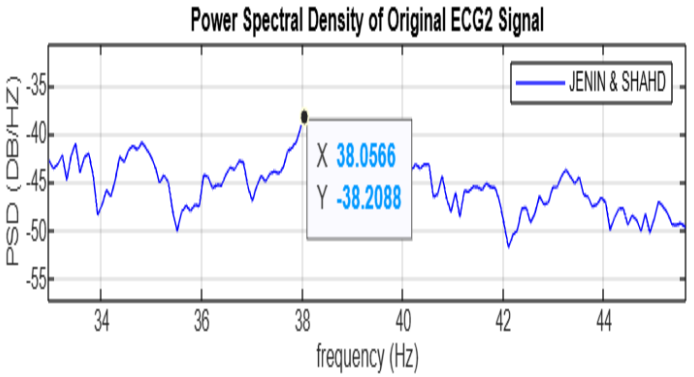

Figure 19: original ECG2 in frequency domain

The difference is clear, in Fig.16 and 17 at low frequencies the magnitude of the signal(y-axis) is less than the original and the signal appears smoother. Finally, there is no change at high frequencies as shown below.

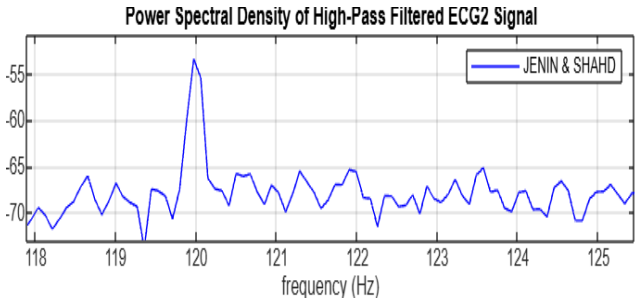

Figure 21: filtered ECG2 signal at high frequency

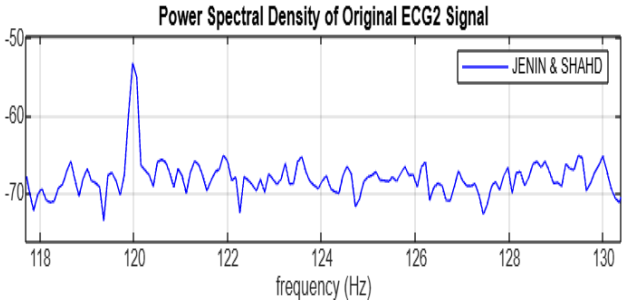

Figure 20: original ECG2 signal at high frequency

2.1.5 How to partially resolve the problems occurring at the start of the signal?

The problem at the beginning of the signal is baseline wander and it can be dealt with using different methods. The first one is a high-pass filter, which removes low-frequency components like baseline wander (we used it in the previous part). Another method is the Moving Average Filter, which smooths out the signal by averaging nearby data points. Polynomial Fitting fits a curve to the baseline, letting us estimate and subtract the baseline wander. Wavelet Transform separates the signal into parts, making it easier to remove baseline wander while keeping important features. Lastly, Adaptive Filtering, like the Least Mean Squares (LMS) filter, adjusts itself based on the signal to remove baseline wander effectively [4].

2.2 Low-pass filter

$$LP(n) = 2.LP(n-1) - LP(n-2) + X(n) - 2X(n-6) + X(n-12)$$

2.2.1 Calculate its transfer function.

$$\boldsymbol{LP(n) - 2 \cdot LP(n-1) + LP(n-2) = X(n) - 2 \cdot X(n-6) + X(n-12)}$$

$$\mathbf{LP(Z) - 2LP(Z)Z^{-1} + Z^{-2}LP(Z) = X(Z) - 2Z^{-6}X(Z) + Z^{-12}X(Z)}$$

$$\mathbf{LP(Z)(1 - 2Z^{-1} + Z^{-2}) = X(Z)(1 - 2Z^{-6} + Z^{-12})}$$

$$\mathbf{H(Z) = \frac{LP(Z)}{X(Z)} = \frac{1 + 2Z^{-6} + Z^{-12}}{1 - 2Z^{-1} + Z^{-2}}}$$

2.2.2 Find and plot the Frequency response.

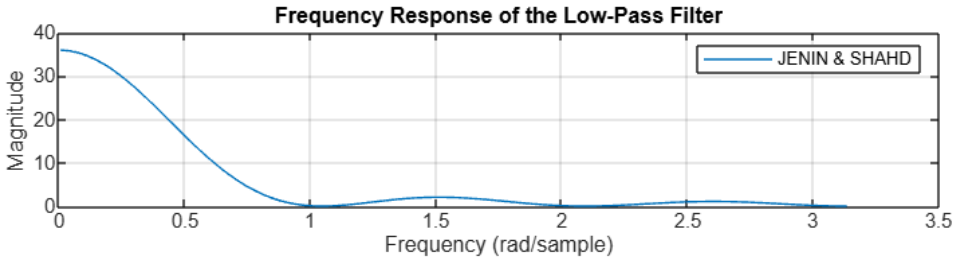

Figure 22:Frequency response of low pass Filter

Using the filter function in MATLAB with the filter coefficients, we obtained the following plot. (write the freq. function)

2.2.3 Which family does it belong to (FIR, IIR)?

The low-pass filter depicted in the zero-pole plot in Fig. 23 is a Finite Impulse Response (FIR) filter. The absence of poles indicates that this filter is not an Infinite Impulse Response (IIR) filter. FIR filters are characterized by having only zeros and no poles (except possibly at the origin), whereas IIR filters have both poles and zeros. Even though there is a pole at z=1, it is canceled by a zero, confirming that the filter is indeed FIR.

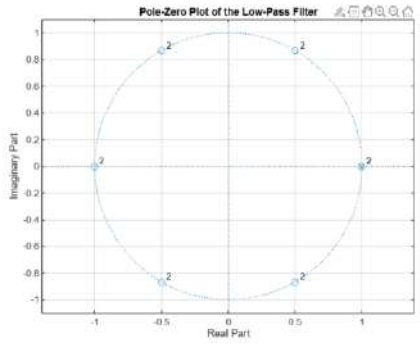

Figure 23:pole - zero plot of the low pass filter

2.2.4 Apply this filter to the ECG1 and then ECG2 signal.

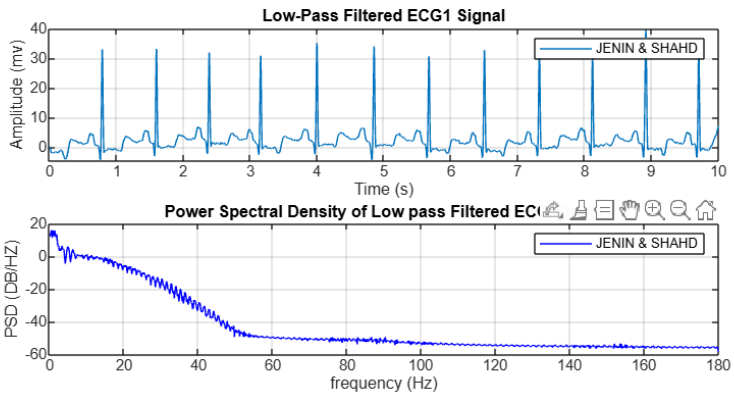

Figure 25: Low pass filtered ECG1 signal

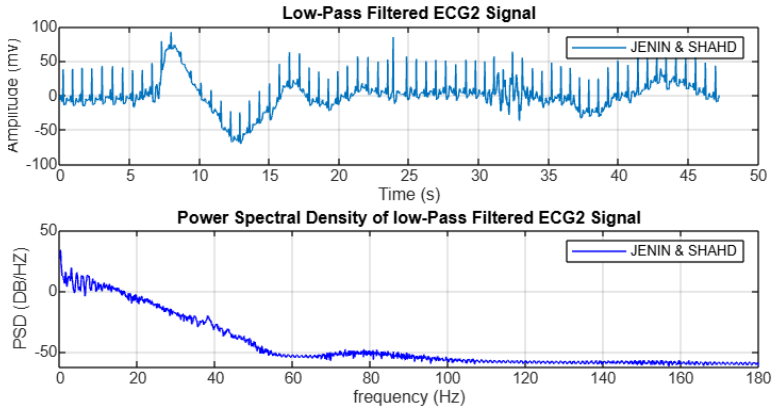

Figure 24: Low pass filtered ECG2 signal

The first thing we can notice in the signal after it passes through the low-pass filter is the cancelation of high frequencies in the frequency domain in both signals as shown in the second subplot above and described in detail below when I zoom in.

ECG1 Signal:

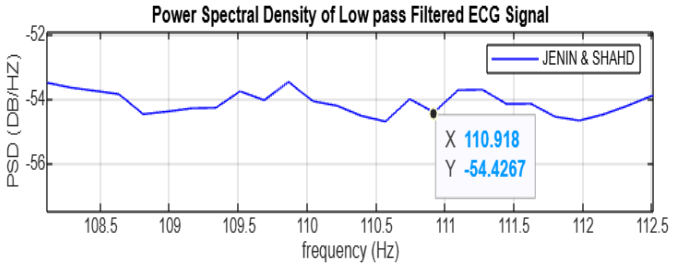

Figure 27: Filtered ECG2 signal at high frequency

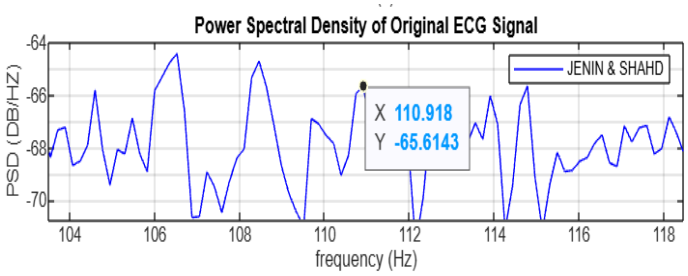

Figure 26:original ECG1 signal at high frequency

ECG2 Signal:

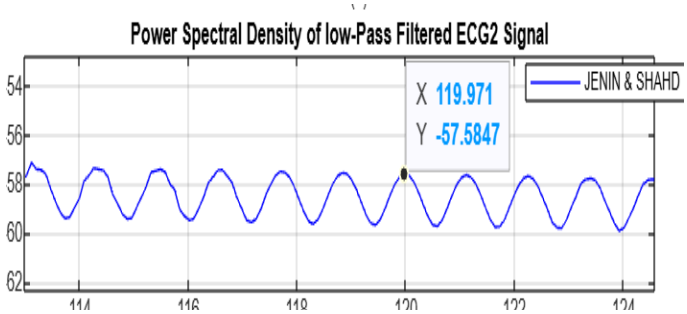

Figure 28:Filtered ECG2 signal at high frequency

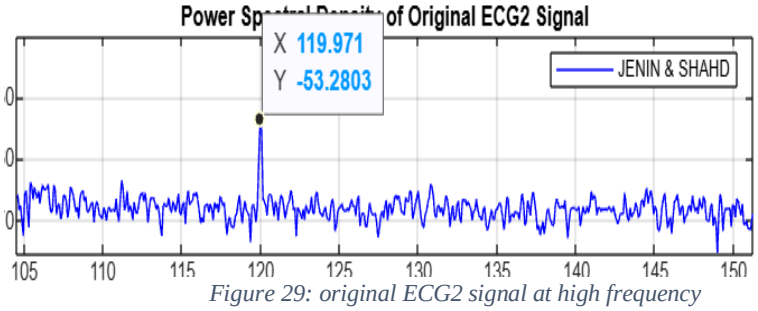

Figure 29: original ECG2 signal at high frequency

The difference is clear, in Fig.27 and 28 at high frequencies the magnitude of the signal(y-axis) is less than the original and the signal appears smoother.

Finally, there is no effect at low frequency, the effect of the low frequencies (baseline wander) is still very clear but it become more smoother as described in detail below

ECG1 Signal:

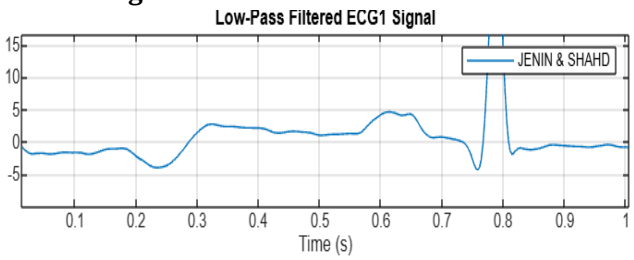

Figure 30:Baseline Wander of ECG1 using low pass filter.

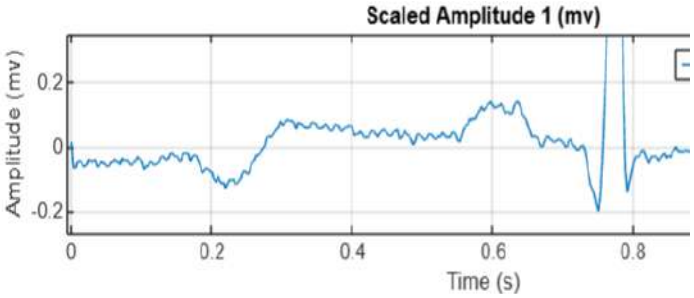

Figure 31:Baseline Wander effect of original ECG1

ECG2 Signal:

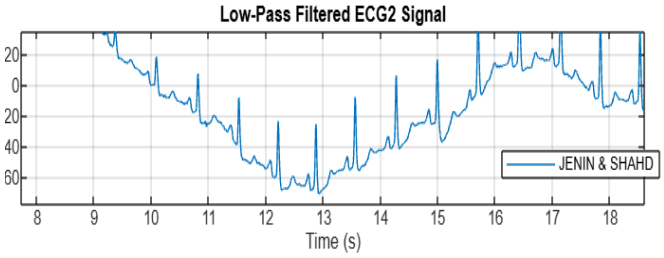

Figure 33:Baseline Wander removed of ECG2 using low pass filter.

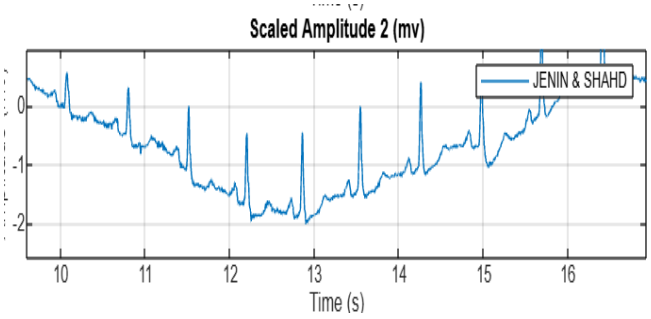

Figure 32: Baseline Wander effect of original ECG2

2.3 Combined Filtering

As shown in below figures there is no difference if we reversed the filters. While the filtered signals become smoother, and the effect of both the high and low frequency is canceled as shown, The high-pass filter reduces low-frequency noise, while the low-pass filter attenuates high-frequency noise, resulting in a cleaner, smoother signal. This combined filtering approach ensures that both high and low-frequency noise are effectively removed, maintaining the essential features of the signal.

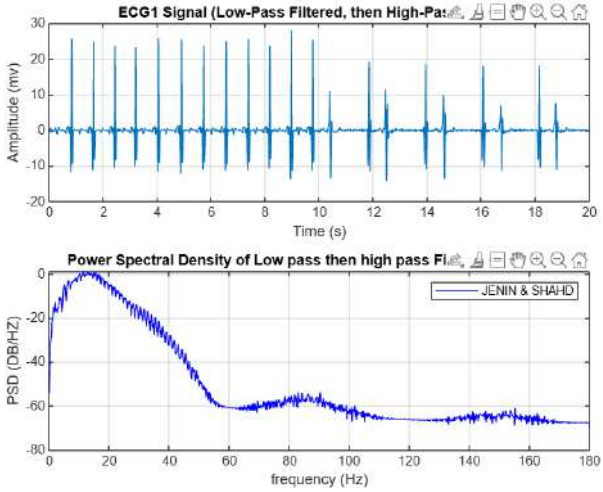

Figure 35 :Low pass then High pass Filtering of ECG1

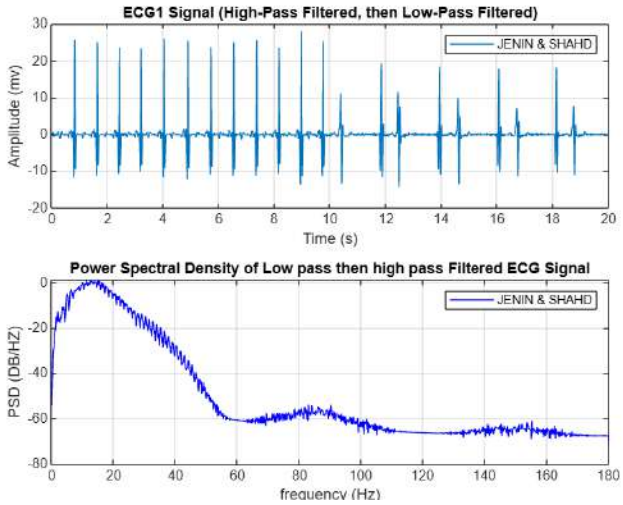

Figure 34: High pass then Low pass filtering of ECG1

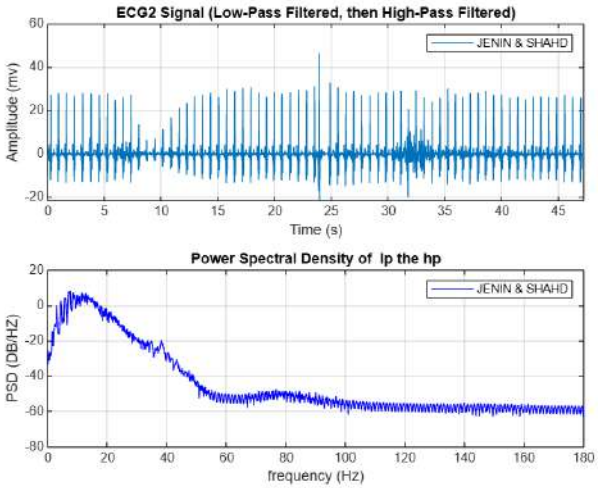

Figure 36:Low pass then High pass Filtering of ECG2

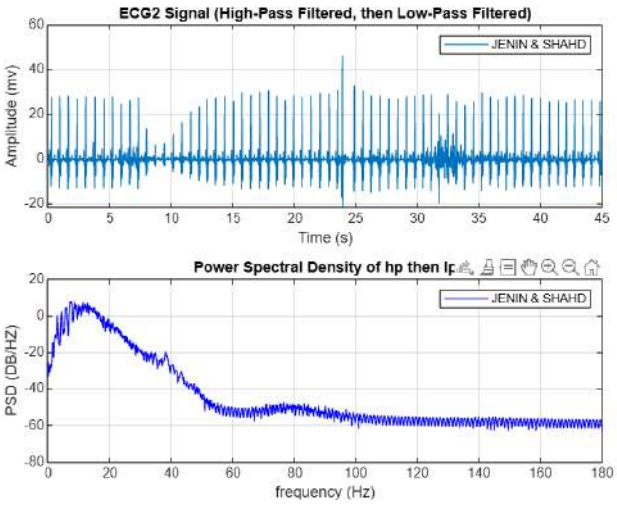

Figure 37:High pass then Low pass filtering of ECG2

The filtered signal has reduced baseline wander and high-frequency noise, providing a cleaner ECG signal.

ECG2

Baseline wander (Low frequency effect) is removed:

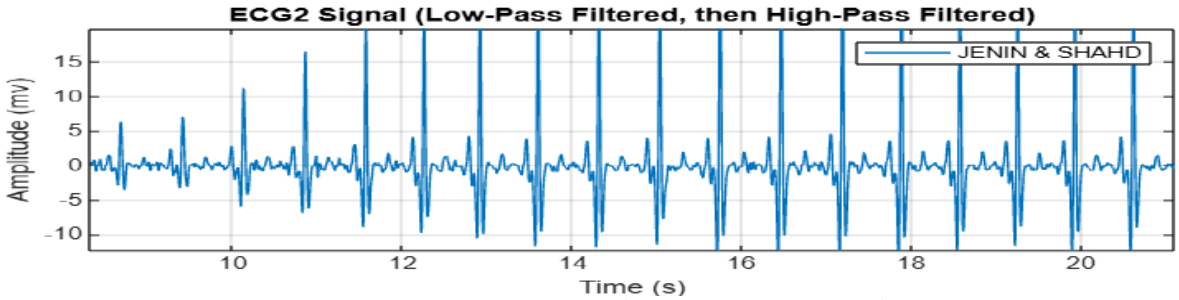

Figure 38 :Baseline correction of ECG2 using LP then HP filter

High frequency effect is removed and the signal smoother.

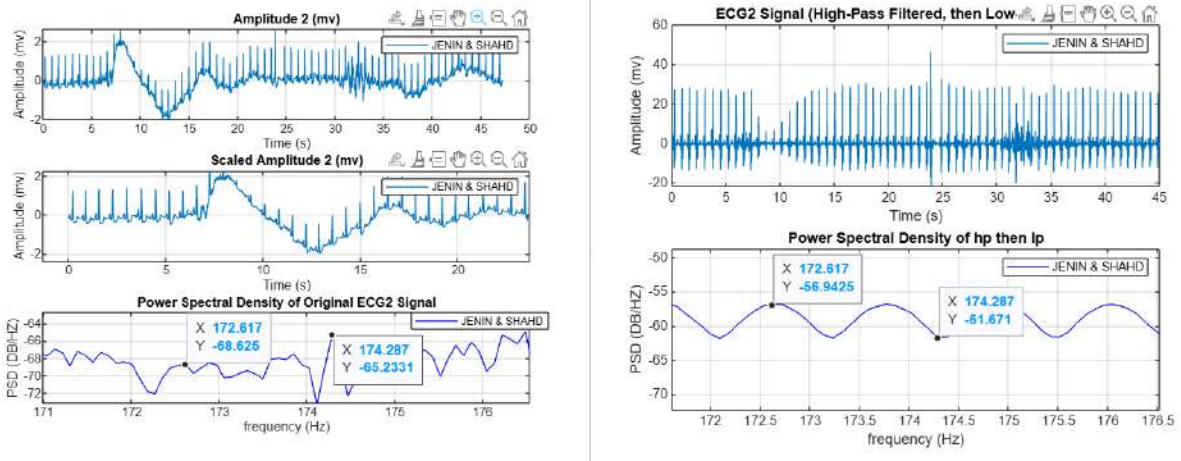

Figure 39:High frequency effect correction of ECG2

ECG1

Baseline wander (Low frequency effect) is removed:

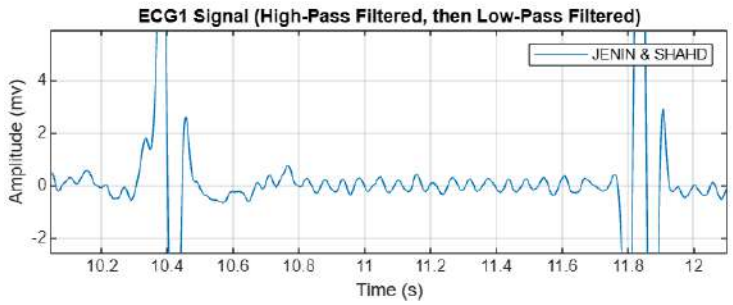

Figure 40: Baseline correction of ECG1 using HP then LP filter

High frequency effect is removed.

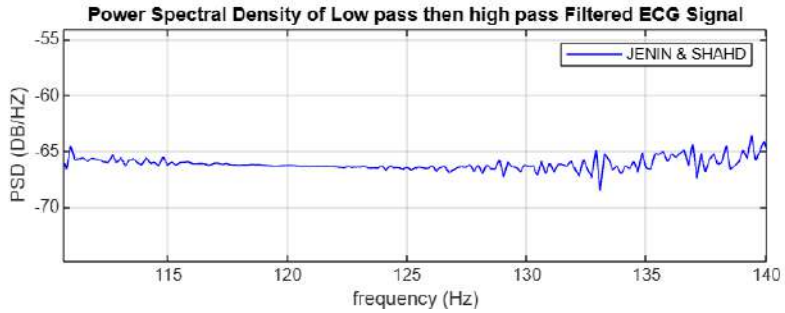

Figure 41: High frequency effect correction of ECG1

As show below the signal become more smoother with clearly visible P, QRS, and T waves. Almost the same shape as Fig2.

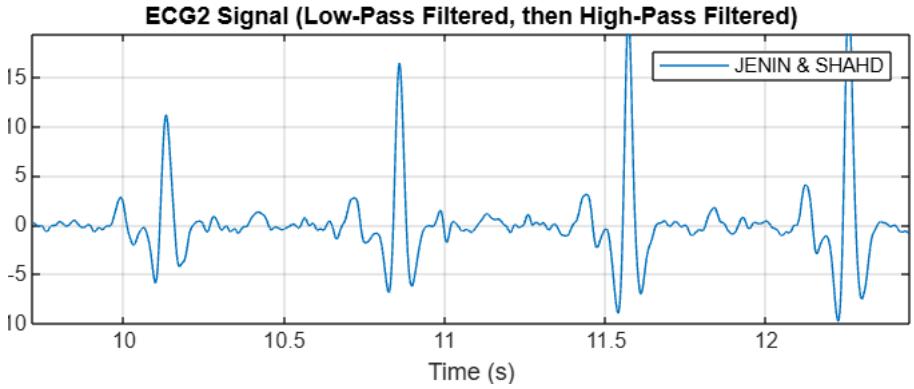

Figure 42: Final ECG2 signal

Conclusion

In this project, we focused on the preprocessing of ECG signals to improve their readability and facilitate automatic analysis. We employed filtering techniques, specifically high-pass and low-pass filters, to eliminate baseline wander and high-frequency noise, respectively. The high-pass filter successfully removed low-frequency interference, centering the ECG signals around zero. And the low-pass filter attenuates high-frequency noise, We visualized the signals in both the time and frequency domains to analyze the effects of the filters. The power spectral density plots confirmed the removal of low and high frequency components. This preprocessing step is crucial for enhancing the quality of ECG signals, making them more suitable for further clinical analysis and automated processing. The results demonstrated the effectiveness of our filtering approach in cleaning the ECG data, thereby paving the way for more accurate and reliable ECG interpretation.

References

[1] https://www.gehealthcare.com/insights/article/a-guide-to-ecg-signal-filtering#:~:text=All%20of%20these%20documents%20remain,%2C%20and%20anti-aliasing%20filters.
[Accessed 7 June 2024, 3:17]

[2] https://en.wikipedia.org/wiki/Electrocardiography
[Accessed 7 June 2024, 4:00]

[3] https://academy.theortusgroup.com/en-gb/ecg-filtering-that-can-help-save-lives#:~:text=For%20diagnostic%20interpretation%2C%20the%20ECG,pass%20filter%20with%20150%20Hz[Accessed 7 June 2024, 4:12]

[4] https://www.researchgate.net/publication/220902643_Comparison_of_different_approaches_for_removal_of_Baseline_wander_from_ECG_signal[Accessed 7 June 2024, 7:00]

Appendix:

```
% Define the filename and sheet name
filename = 'Data_ECG_raw.xlsx';
fs =360;
ts= 1/fs;

% Read the table from the specified sheet
data1 = readtable(filename, 'Sheet', 'ECG1');
data2 = readtable(filename, 'Sheet', 'ECG2');

% Remove unnecessary columns from data1
data1(:, {'Var5', 'Var6', 'Var7', 'Var8'}) = [];

% Access specific columns
sample_number1 = data1.('sampleNumber');
temps1 = data1.('temps_s_');
amplitude1 = data1.('amplitude_mv_');  % First amplitude column


% Insert the real-time column
sampling_frequency = 360;  % 360 Hz
real_time1 = sample_number1 / sampling_frequency;
% Check for NaN values in amplitude1
nan_indices = isnan(amplitude1);
if any(nan_indices)
    % Interpolate or remove NaN values
    amplitude1 = fillmissing(amplitude1, 'linear');
end
% Append the real-time column to the data table
data1.real_time = real_time1;
disp(head(data1));
%rescle
gain=200;

% Create the first figure
figure;

% Plot the original ECG signal
subplot(3, 1, 1);
%plot(real_time1(1:2*fs),amplitude1(1:2*fs));
plot(data1.real_time, amplitude1, 'LineWidth', 0.5);
title('Amplitude 1 (mv)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
legend('JENIN & SHAHD');
xlim([0 22.5]);
grid on;
%%%
amplitude1 = amplitude1/gain;


% Plot the original ECG signal
subplot(3, 1, 2);
%plot(real_time1(1:2*fs),amplitude1(1:2*fs));
plot(data1.real_time, amplitude1, 'LineWidth', 0.5);
title('Scaled Amplitude 1 (mv)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
xlim([0 10]);
legend('JENIN & SHAHD');
grid on;
%%%


%plot the signal in frequency domain
subplot(3, 1, 3);
[pxx,f] = pwelch(amplitude1 ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of Original ECG Signal');
legend('JENIN & SHAHD');
grid on;

%%%

% Coefficients of the high-pass filter
b_hp = [-1/32, zeros(1, 15), 1, -1, zeros(1, 14), 1/32]; % Numerator
a_hp = [1, -1]; % Denominator


% Frequency response
[h_hp, w_hp] = freqz(b_hp, a_hp, 1024, sampling_frequency); % 1024 points, 360 Hz sampling frequency

figure;

% Plot frequency response in radians per sample
subplot(3, 1, 1);
plot(w_hp, abs(h_hp));
title('Frequency Response of the High-Pass Filter');
xlabel('Frequency (rad/sample)');
ylabel('Magnitude');
legend('JENIN & SHAHD');
grid on;

% Apply the high-pass filter to the amplitude_mv signal
hp_amplitude1 = filter(b_hp, a_hp , amplitude1);
% Plot the filtered ECG signal
subplot(3, 1, 2);
plot(real_time1, hp_amplitude1, 'LineWidth', 0.5);
%plot(real_time1(1:2*fs),hp_amplitude1(1:2*fs));
title('High-Pass Filtered ECG1 Signal');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
xlim([0 20]);
legend('JENIN & SHAHD');
grid on;


%plot the signal in frequency domain
subplot(3, 1, 3);
[pxx,f] = pwelch(hp_amplitude1 ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of High-Pass Filtered ECG Signal');
legend('JENIN & SHAHD');
grid on;



% Create the second figure
figure;


% Define the low-pass filter coefficients
b_lp = [1, 0, 0, 0, 0, 0, -2, 0, 0, 0, 0, 0, 1];
a_lp = [1, -2, 1];

% Compute the frequency response of the low-pass filter
[h_lp, w_lp] = freqz(b_lp, a_lp);


% Plot frequency response of the low-pass filter
subplot(3, 1, 1);
plot(w_lp,abs(h_lp));
```

```
title('Frequency Response of the Low-Pass Filter');
xlabel('Frequency (rad/sample)');
ylabel('Magnitude');
legend('JENIN & SHAHD');
grid on;

lp_amplitude1 = filter(b_lp, a_lp,amplitude1);

subplot(3, 1, 2);
plot(real_time1, lp_amplitude1, 'LineWidth', 0.5);
%plot(real_time1(1:2*fs),lp_amplitude1(1:2*fs));
title('Low-Pass Filtered ECG1 Signal');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
xlim([0 10]);
legend('JENIN & SHAHD');
grid on;


%plot the signal in frequency domain
subplot(3, 1, 3);
[pxx,f] = pwelch(lp_amplitude1 ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of Low pass Filtered ECG Signal');
legend('JENIN & SHAHD');
grid on;



figure;

% Apply the low-pass filter to the high-pass filtered ECG signals
ecg1_hp_lp = filter(b_lp, a_lp, hp_amplitude1);

% Plot the high-pass and low-pass filtered ECG signal
subplot(2, 1, 1);
% plot(real_time1(1:2*fs),ecg1_hp_lp(1:2*fs));

plot(real_time1, ecg1_hp_lp, 'LineWidth', 0.5);
title('ECG1 Signal (High-Pass Filtered, then Low-Pass Filtered)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
legend('JENIN & SHAHD');
xlim([0 20]);
grid on;



%plot the signal in frequency domain
subplot(2, 1, 2);
[pxx,f] = pwelch(ecg1_hp_lp ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of Low pass then high pass Filtered ECG Signal');
legend('JENIN & SHAHD');
grid on;



%%%%%%%%%%%

figure;


% Apply the filters in reverse order: low-pass then high-pass
ecg1_lp_hp = filter(b_hp, a_hp,lp_amplitude1);


% Plot the low-pass then high-pass filtered signals
subplot(2, 1, 1);
%plot(real_time1(1:2*fs),ecg1_lp_hp(1:2*fs));
plot(real_time1, ecg1_lp_hp, 'LineWidth', 0.5);
title('ECG1 Signal (Low-Pass Filtered, then High-Pass Filtered)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
xlim([0 20]);
grid on;

%plot the signal in frequency domain
subplot(2, 1, 2);
[pxx,f] = pwelch(ecg1_lp_hp ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of Low pass then high pass Filtered ECG Signal');
legend('JENIN & SHAHD');
grid on;

%%%%%%%%%%%
% Calculate poles and zeros of the high-pass filter
z_hp = roots(b_hp); % Zeros
p_hp = roots(a_hp); % Poles

% Plot the pole-zero plot for the high-pass filter
figure;
zplane(z_hp, p_hp);
title('Pole-Zero Plot of the High-Pass Filter');
xlabel('Real Part');
ylabel('Imaginary Part');
grid on;

% Calculate poles and zeros of the low-pass filter
z_lp = roots(b_lp); % Zeros
p_lp = roots(a_lp); % Poles

% Plot the pole-zero plot for the low-pass filter
figure;
zplane(z_lp, p_lp);
title('Pole-Zero Plot of the Low-Pass Filter');
xlabel('Real Part');
ylabel('Imaginary Part');
legend('JENIN & SHAHD');
grid on;


%%%%%%ECG2**************************************************


% Access specific columns
sample_number2 = data2.('n');
temps2 = data2.('temps_s_');
amplitude2 = data2.('amplitude_mv_');  % First amplitude column


% Insert the real-time column
real_time2 = sample_number2 / sampling_frequency;
% Check for NaN values in amplitude1
nan_indices = isnan(amplitude2);
if any(nan_indices)
    % Interpolate or remove NaN values
    amplitude2 = fillmissing(amplitude2, 'linear');
end
% Append the real-time column to the data table
data2.real_time = real_time2;

% Display the first few rows to verify the real-time column
disp(head(data2));


figure;

% Plot the original ECG signal
subplot(3, 1, 1);
 plot(data2.real_time, amplitude2, 'LineWidth', 0.5);
title('Amplitude 2 (mv)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
legend('JENIN & SHAHD');
grid on;



amplitude2 = amplitude2/gain;

% Plot the original ECG signal
subplot(3, 1, 2);
plot(data2.real_time, amplitude2, 'LineWidth', 0.5);
% plot(real_time1(1:2*fs),amplitude2(1:2*fs));
title('Scaled Amplitude 2 (mv)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
legend('JENIN & SHAHD');
xlim([0 25]);
grid on;


%plot the signal in frequency domain
subplot(3, 1, 3);
[pxx,f] = pwelch(amplitude2 ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of Original ECG2 Signal');
legend('JENIN & SHAHD');
grid on;


%%%%%%%%%%%%%%%
figure;
```

```
% Apply the high-pass filter to the amplitude2_mv signal
hp_amplitude2 = filter(b_hp, a_hp,amplitude2);


% Plot frequency response in radians per sample
subplot(3, 1, 1);
plot(w_hp, abs(h_hp));
title('Frequency Response of the High-Pass Filter');
xlabel('Frequency (rad/sample)');
ylabel('Magnitude');
legend('JENIN & SHAHD');
grid on;


% Plot the filtered ECG signal
subplot(3, 1, 2);
% plot(real_time2(1:2*fs),hp_amplitude2(1:2*fs));
plot(real_time2, hp_amplitude2, 'LineWidth', 0.5);
title('High-Pass Filtered ECG2 Signal');
xlabel('Time (s)');
ylabel('Amplitude 2 (mv)');
legend('JENIN & SHAHD');
grid on;

%plot the signal in frequency domain
subplot(3, 1, 3);
[pxx,f] = pwelch(hp_amplitude2 ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of High-Pass Filtered ECG2 Signal');
legend('JENIN & SHAHD');
grid on;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

figure;

% Plot frequency response of the low-pass filter
subplot(3, 1, 1);
plot(w_lp, h_lp);
title('Frequency Response of the Low-Pass Filter');
xlabel('Frequency (rad/sample)');
ylabel('Magnitude');
legend('JENIN & SHAHD');
grid on;

lp_amplitude2 = filter(b_lp, a_lp,amplitude2);

subplot(3, 1, 2);
% plot(real_time2(1:2*fs),lp_amplitude2(1:2*fs));
plot(real_time2, lp_amplitude2, 'LineWidth', 0.5);
title('Low-Pass Filtered ECG2 Signal');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
legend('JENIN & SHAHD');
grid on;

%plot the signal in frequency domain
subplot(3, 1, 3);
[pxx,f] = pwelch(lp_amplitude2 ,[],[],[],fs);
plot(f,10*log10(pxx),'b');
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of low-Pass Filtered ECG2 Signal');
legend('JENIN & SHAHD');
grid on;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure;

% Apply the low-pass filter to the high-pass filtered ECG signals
ecg2_hp_lp = filter(b_lp, a_lp, hp_amplitude2);

% Plot the high-pass and low-pass filtered ECG signal
subplot(2, 1, 1);
plot(real_time2, ecg2_hp_lp, 'LineWidth', 0.5);
title('ECG2 Signal (High-Pass Filtered, then Low-Pass Filtered)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
legend('JENIN & SHAHD');
xlim([0 45]);
grid on;


%plot the signal in frequency domain
subplot(2, 1, 2);
[pxx,f] = pwelch(ecg2_hp_lp ,[],[],[],fs);
plot(f,10*log10(pxx),"b");
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of hp then lp');
legend('JENIN & SHAHD');
grid on;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure;

% Apply the filters in reverse order: low-pass then high-pass
ecg2_lp_hp = filter(b_hp, a_hp,lp_amplitude2);

% Plot the low-pass then high-pass filtered signals
subplot(2, 1, 1);
plot(real_time2, ecg2_lp_hp, 'LineWidth', 0.5);
%plot(real_time1(1:2*fs),ecg2_lp_hp(1:2*fs));
title('ECG2 Signal (Low-Pass Filtered, then High-Pass Filtered)');
xlabel('Time (s)');
ylabel('Amplitude (mv)');
xlim([0 45]);
legend('JENIN & SHAHD');
grid on;

%plot the signal in frequency domain
subplot(2, 1, 2);
[pxx,f] = pwelch(ecg2_lp_hp ,[],[],[],fs);
plot(f,10*log10(pxx),"b");
xlabel('frequency (Hz)');
ylabel('PSD (DB/HZ)');
title('Power Spectral Density of  lp the hp');
legend('JENIN & SHAHD');
grid on;
```