



## **Independent University Bangladesh**

Department of Electrical and Electronics Engineering

### **Project on Digital Filter Design**

Name: Injamamul Haque Sourov

Id: 1820170

Course code: EEE 321L

Couse name: Digital Signal Processing Lab

Semester: Autumn 2020

Date: 08/01/2021

**Task:** To remove the noise and artifact from a contaminated ECG signal to produce a clean signal such as provided.

### Introduction:

A filter is a device that is used to select certain frequencies from a signal while rejecting others. Filters can be of digital (discrete) or analogue (continuous) in nature. However, the design of a digital filter is preceded by that of an analog filter. From the signal processing perspective, the digital filter is nothing but a Linear Time Invariant System (LTI) that can be completely defined by its impulse response  $h(n)$  or the system function  $H(z)$ , written as series of polynomial coefficients  $a$  (denominator, also represents the pole locations) and  $b$  (numerator, also represent the zero locations) that also defines the filter order.

### Filter type:

Both digital and analog filters can be of different types depending on the criteria of frequency selection (such as high pass, low pass, band pass and band stop filters) or the type of design (such as Butterworth, Chebyshev, and Elliptic).

Further classification of filters can be made depending on the duration of this impulse response, infinite impulse response (IIR) or finite impulse response (FIR). The table below differentiates between these two types.

Finite Impulse Response (FIR)	Infinite Impulse Response (IIR)
Finite in length (duration)	Infinite in length (duration)
Obtained from IIR using some sort of window function	Designed independently
Has linear phase response	Has non-linear phase response
System function contains $b$ (zeros) coefficients only	Both $a$ (poles) and $b$ (zeros) coefficients present in system function
Is modelled as an open loop system	Is modelled as a close loop system
Absence of poles mean the system is always stable	Position of poles on the right half plane will make the system unstable
Requires <i>relatively</i> higher order design (more coefficients)	Requires <i>relatively</i> lower order design (fewer coefficients)
Larger computational complexity	Smaller computational complexity

It should be noted that for the same slope of the filter (ideally infinite), FIR filters require much higher order (N) design than IIR filters. *For this particular application, I'll be using an IIR filter as it requires lower order and has no separate window design.*

### Methodology:

The IIR filter is designed using one of the two methods shown below;

- analog low pass filter design (s domain) → frequency band transformations (low pass to high pass etc.) → filter transformation (s-domain to z-domain)
- analog low pass filter design (s domain) → filter transformation (s-domain to z-domain) → frequency band transformations

For this project, I'll be using the latter method for the filter design.

An analog low pass filter (LPF) is usually specified via the parameters shown in the figure below;

## Filter Specification

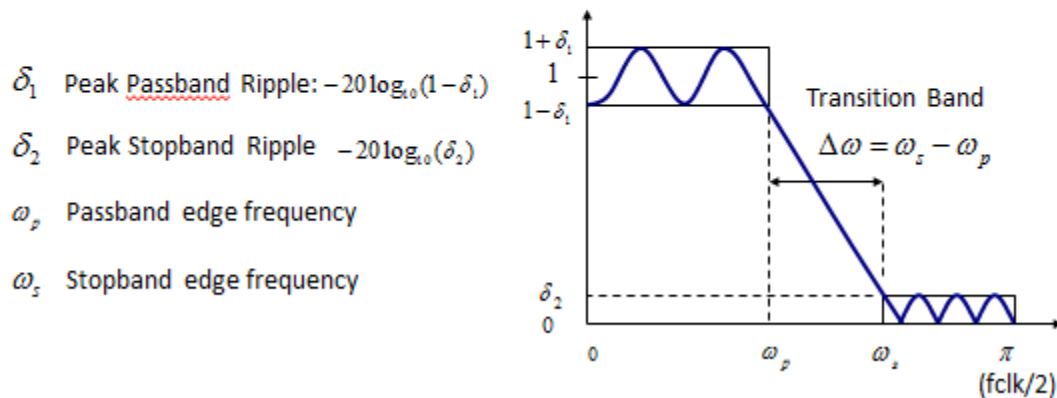


Figure: Analog Low Pass Filter specifications

Often the pass band ripple and the stop band attenuation are represented in the decibel (dB) scale are defined as;

$$R_p = -20 \log_{10} \frac{1-\delta_1}{1+\delta_1} > 0 (\approx 0)$$

$$A_s = -20 \log_{10} \frac{\delta_2}{1+\delta_1} > 0 (>> 1)$$

Ideally, the pass band ripple ( $R_p$ ) and the transition band width should be 0 and the stop band attenuation should be 1. This translates to flat pass and stop bands and an infinite transition slope (roll off). The figure below shows the frequency response of different IIR filters that are commonly used.

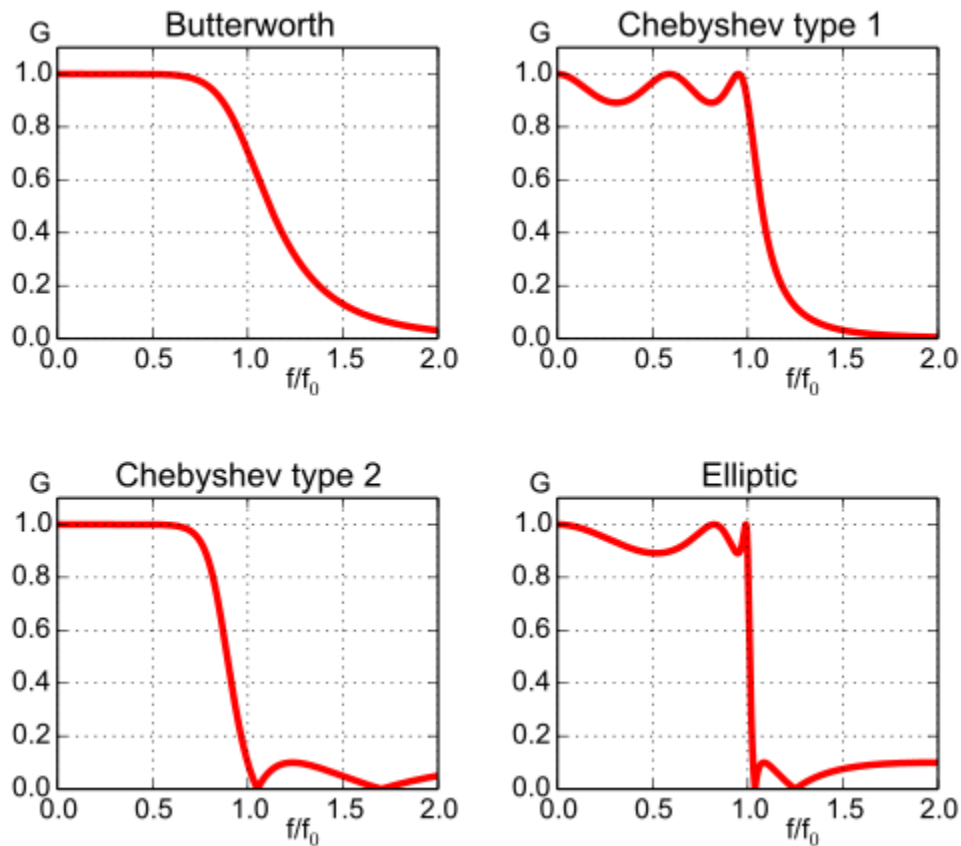


Figure: Frequency response of commonly used IIR filters

It can be seen that the different designs offer a tradeoff between the 3 parameters mentioned before (pass band ripple, stop band attenuation and roll-off). The table summarizes their characteristics;

Type	Pass-band	Stop-band	Roll-off	Step Response
Butterworth	Flat	Monotonic	Good	Good
Chebyshev	Rippled	Monotonic	Very Good	Poor
Inverse Chebyshev	Flat	Rippled	Very Good	Good
Elliptic	Rippled	Rippled	Best	Poor
Bessel	Flat	Monotonic	Poor	Best

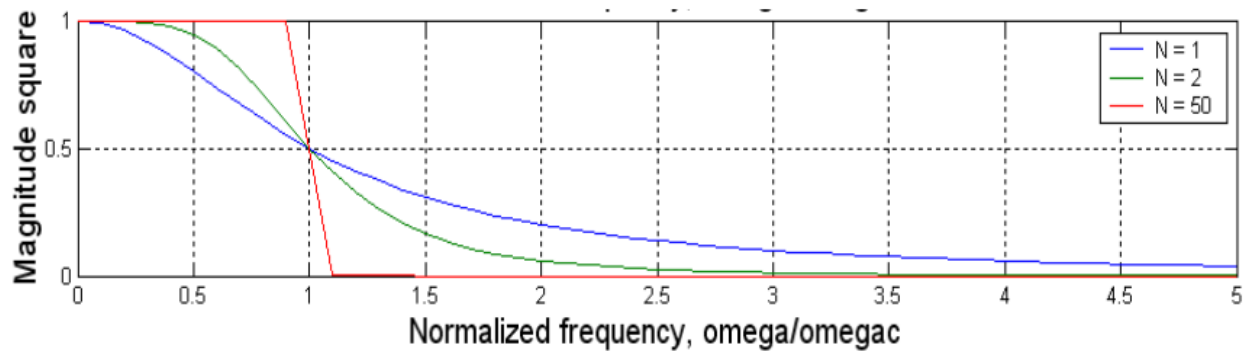
From the above table, it is seen that Butterworth filters provide a balance between the tradeoffs. Therefore, *for this project I'll be using the Butterworth filter as it provides a nearly flat pass and stop band along with a good roll-off and step response.*

### The Butterworth Low Pass Filter:

A Butterworth low pass filter is usually represented by its magnitude-square response shown below, where  $\Omega_c$  is the cutoff frequency and  $N$  is the order of the filter.

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \left(\frac{\Omega}{\Omega_c}\right)^{2N}}$$

The figure below shows the effect of filter order on the magnitude-square response. It is seen that the roll-off increases as the order is increased.



For the filter response in s-domain, the poles of the filter needs to be determined using either of the equations (depending on the filter order,  $N$ ).

$$p_k = \Omega_c e^{jk\pi/N}, \quad k = 0, 1, \dots, 2N-1 \quad \text{for odd } N$$

$$p_k = \Omega_c e^{j\left(\frac{\pi}{2N} + \frac{k\pi}{N}\right)}, \quad k = 0, 1, \dots, 2N-1 \quad \text{for even } N$$

Once the poles are known, only the left half planes poles are used to obtain the magnitude response of the Butterworth low pass filter using the equation.

$$H_a(s) = \frac{\Omega_c^N}{\prod_{\text{LHP poles}} (s - p_k)}$$

Therefore, it is seen that the Butterworth low pass filter is completely described by its LHP poles, cutoff frequency  $\Omega_c$  and the order  $N$ .

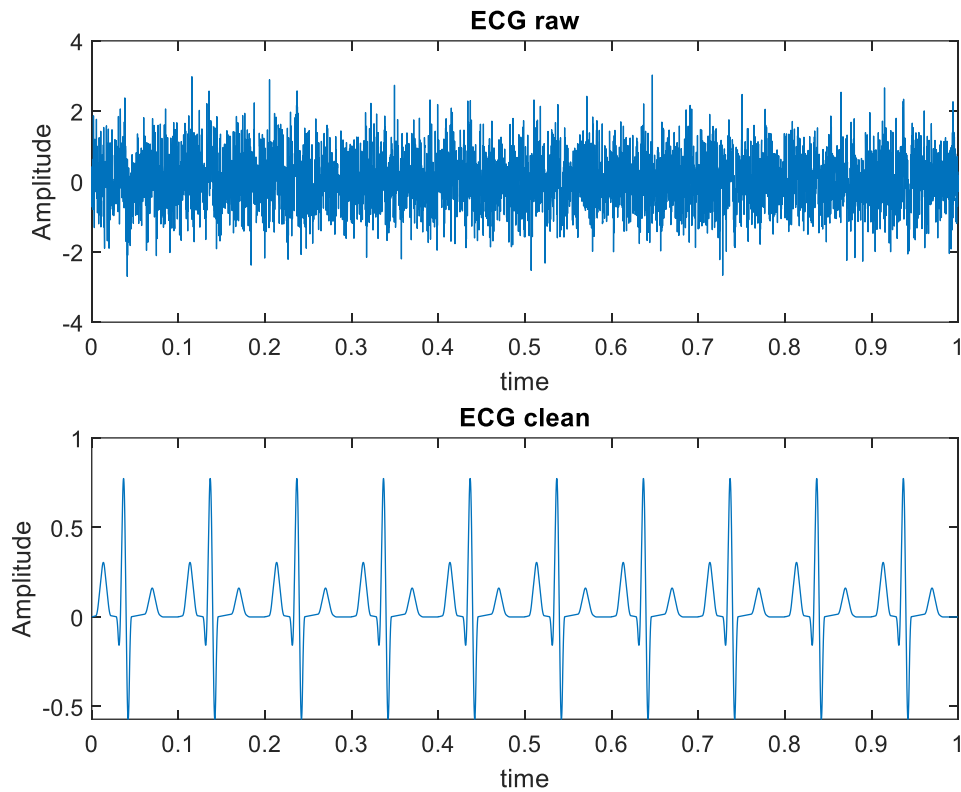
The MATLAB function *butter*( $N, Wn$ ) is used to obtain the system function  $H(z)$  coefficients (b and a) for an  $N$ th order low pass frequency with normalized cutoff frequency  $Wn$  (equal to  $f_c$  divided by half of sampling frequency). If  $Wn$  is given as array of two values and the additional 'stop' parameter is passed in, the coefficients of a band-stop filter with the specified passband and stopband cutoff frequencies can be obtained. (This can be used to design a notch filter to remove the 50Hz power line noise).

Finally, to determine the optimum order of filter, the system can be simulated using different values of  $N$  along with the known cutoff frequency  $\Omega_c$  (around 80Hz, from ECG bandwidth). Using these values of  $N$  and  $\Omega_c$  the filter specification parameters pass band ripple  $R_p$  and stop band attenuation  $A_s$  can be found using the following formulas;

$$-10 \log_{10} \left( \frac{1}{1 + \left( \frac{\Omega_p}{\Omega_c} \right)^{2N}} \right) = R_p \qquad -10 \log_{10} \left( \frac{1}{1 + \left( \frac{\Omega_s}{\Omega_c} \right)^{2N}} \right) = A_s$$

### Simulation:

The *time domain representation* of the supplied raw and clean ECG signals was first observed as shown below;



The *dc offset* was then removed from the raw signal by subtracting the mean value of the signal from the signal itself. The dc offset value was found to be 0.0425.

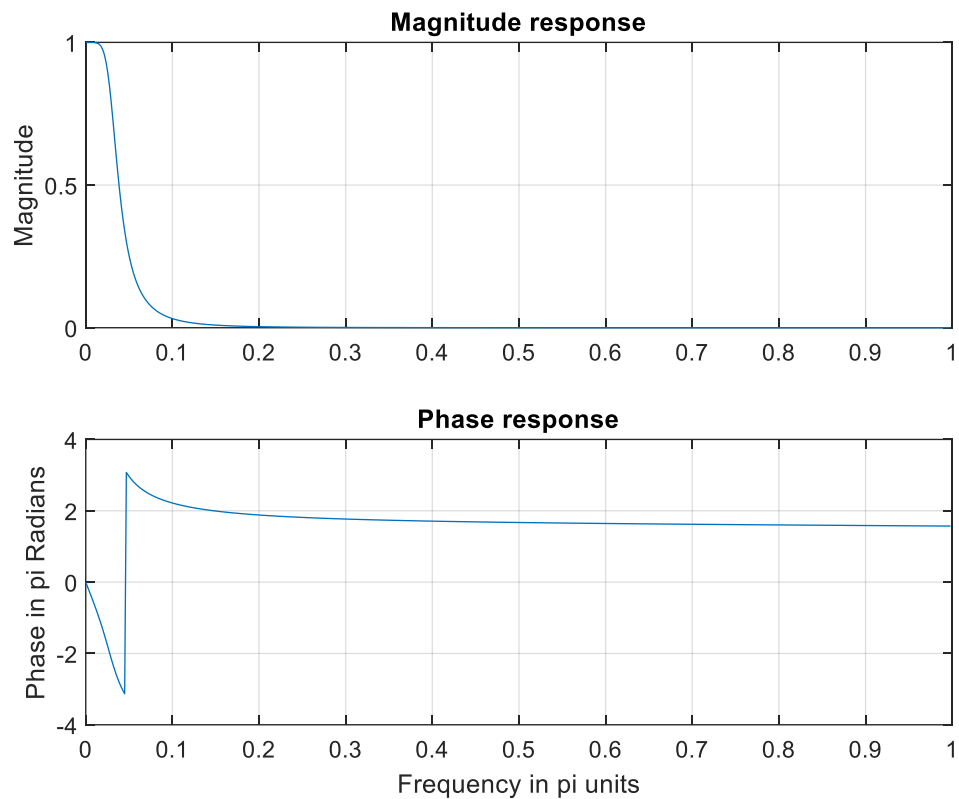
In addition, a notch filter was used to remove the 50Hz power line noise.

By simulating the system for different values of  $N$ , it was decided that a 3<sup>rd</sup> order filter suits this application well. The cutoff-frequency  $\Omega_c$  of the analog LPF is chosen as 80Hz as per the bandwidth of ECG signal. The table below compares the magnitude squared coherence, correlation coefficient and the root mean squared error of the filtered signal with the clean signal, which was then used to determine the optimal order.

Filter Order $N$	RMSE of ECG_raw w.r.t ECG_clean	mscohere of ECG Filtered w.r.t ECG_clean	Corrcoef of ECG Filtered w.r.t ECG_clean	RMSE of ECG Filtered w.r.t ECG_clean
1	0.7567	0.1615	0.8145	0.2076
2		0.1878	0.9642	0.2332
3		0.1785	0.9980	0.2594
4		0.0919	0.9800	0.2695
5		0.1701	0.9816	0.2682
6		0.1976	0.9900	0.2637

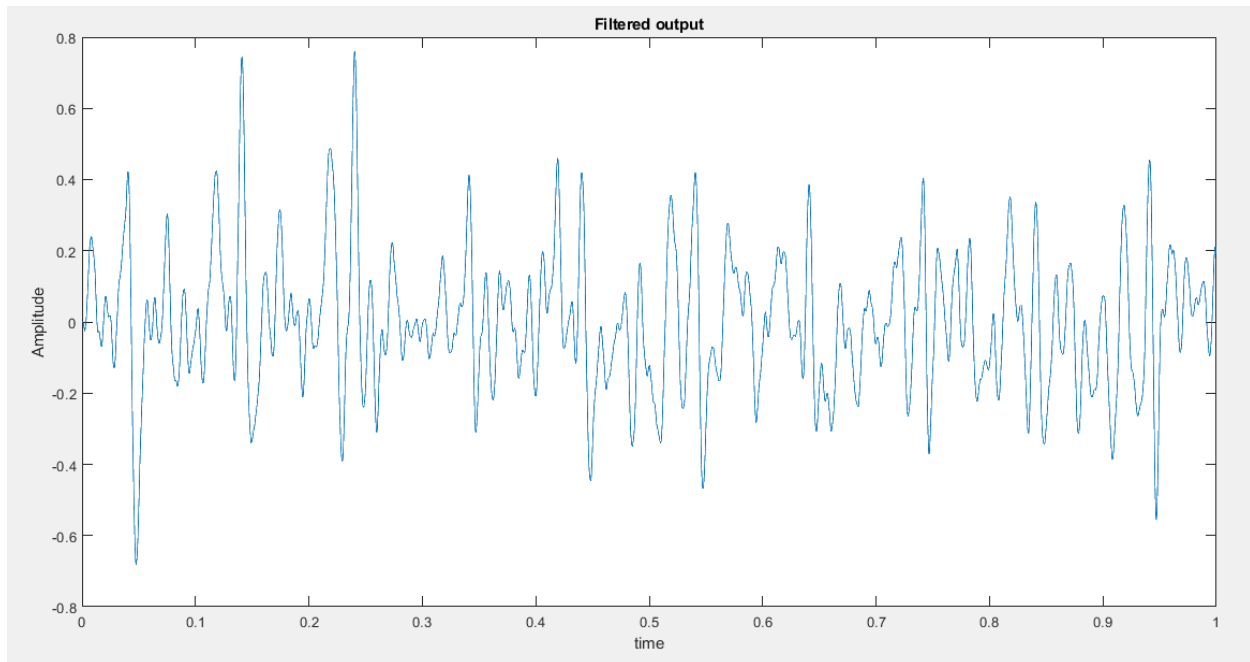
7		0.1678	0.9932	0.2613
8		0.1310	0.9920	0.2619
9		0.1238	0.9890	0.2635
10		0.1403	0.9884	0.2637

Then b and a coefficients were obtained from the 3<sup>rd</sup> order Butterworth filter as specified above. The magnitude and the phase response of the system function is shown below;



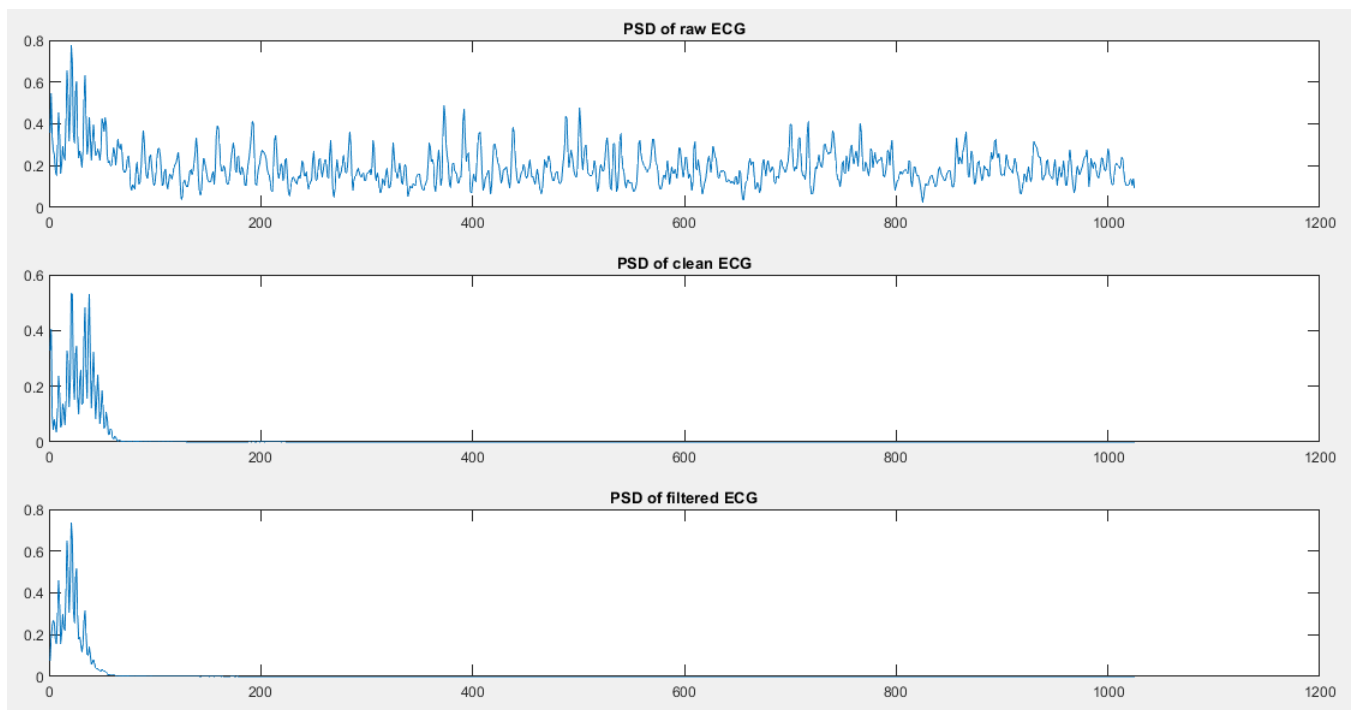
The raw signal with the noises removed was then filtered using b, a coefficients obtained from the butter function and then plotted.





It is seen that the filtered output does not exactly correspond but was somewhat close to the clean ECG signal.

To analyze the frequency spectrum, the power spectrum density was found using the pwelch function. The PSD of the three signals is shown below. It can be seen that the PSD of the filtered signal is again close to that of the clean ECG.



Performance evaluation of the final designed filter compare to clean signal:

Order N	RMSE of Raw	RMSE of filtered	mscohere of filtered	Corrcoef of filtered
3	0.7567	0.2594	0.1785	0.9980

### Complete MATLAB code:

```
% time domain parameters
n = 0:4999;      % 5000 samples
Fs = 5000;      % sampling freq
t = n/Fs;

% Plot raw and clean ecg
figure(1);
subplot(2,1,1); plot(t, ECG_raw);
title('ECG raw');
ylabel('Amplitude');
xlabel('time');
subplot(2,1,2); plot(t, ECG_clean);
title('ECG clean');
ylabel('Amplitude');
xlabel('time');

% remove any dc offset
m = mean(ECG_raw);
x = ECG_raw - m; % signal without dc offset

% 50hz notch filter (fc/(fs/2))
[b,a] = butter(3,[0.0196 0.0204],'stop'); % Bandstop = 49Hz-51Hz, 6th Order
z = filter(b,a,x);

% butterworth filter
fc = 80; % cutoff frequency
fs = 5000; % sampling frequency
N = 3; % 3rd order filter
Wn = fc/(fs/2); % freq in rad/sample
[b,a] = butter(N, Wn);

% magnitude and phase
[H,W]= freqz(b, a);
magH = abs(H);
angH = angle(H);
% plot mangnitue and phase response
figure(2);
subplot(2,1,1); plot(W/pi, magH); grid
title('Magnitue response');
ylabel('Magnitue');
subplot(2,1,2); plot(W/pi, angH); grid
title('Phase response');
xlabel('Frequency in pi units');
ylabel('Phase in pi Radians');
```

```

%filter
clean = filter(b,a,z);
%plot filtered output
figure (3)
plot(t,clean);
title('Filtered output');
ylabel('Amplitude');
xlabel('time');

% PSD
PSD_r= pwelch(ECG_raw);
PSD_c= pwelch(ECG_clean);
PSD= pwelch(clean);
% plot PSD
figure (4);
subplot(3,1,1); plot(PSD_r); title('PSD of raw ECG');
subplot(3,1,2); plot(PSD_c); title('PSD of clean ECG');
subplot(3,1,3); plot(PSD); title('PSD of filtered ECG');

%performance
msc= mean(mscohere(clean,ECG_clean));
corr= det(corrcoef(clean,ECG_clean));
rmse= sqrt(mean((ECG_clean - clean).^2));
rmse2= sqrt(mean((ECG_clean - ECG_raw).^2));

```

## Discussion:

The order of the filter was assumed and then simulated on MATLAB as explained. It was found that as N was increased the filtered signal more closely resembled the clean signal however for N greater than 12 the filtered output was completely off and the magnitude and phase responses were very far from the expected. The design parameters  $R_p$ ,  $A_s$  were not obtained directly however they could be calculated using the formulas given provided that the pass and stop band frequencies are approximated close to the cutoff frequency.

## Conclusion:

The noise from the raw ECG signal was somewhat successfully removed from the signal. The power density spectrum of the clean signal closely resembled that of the filtered output. Various performance parameters such as root mean square error and correlation coefficient were calculated and tabulated to quantitatively compare the clean signal with the filtered output. The RMS error of the filtered output was significantly lower than the raw signal and the correlation coefficient close to 1 suggests that the clean signal and the filtered output closely resembled each other.

## References:

- i. Milchevski A., Gusev M. (2018) Performance Evaluation of FIR and IIR Filtering of ECG Signals. In: Stojanov G., Kulakov A. (eds) ICT Innovations 2016. ICT Innovations 2016. Advances in Intelligent Systems and Computing, vol 665. Springer, Cham.
- ii. <https://www.mathworks.com/help/signal/ref/butter.html#d122e8874>
- iii. [http://ocw.utm.my/file.php/134/M12\\_-\\_IIR\\_Filter\\_Design.pdf](http://ocw.utm.my/file.php/134/M12_-_IIR_Filter_Design.pdf)
- iv. <https://www.mathworks.com/help/signal/ref/periodogram.html#d122e110165>
- v. [http://www.brainkart.com/article/Difference-Between-FIR-Filter-and-IIR-Filter\\_13039/](http://www.brainkart.com/article/Difference-Between-FIR-Filter-and-IIR-Filter_13039/)
- vi. <https://www.mathworks.com/help/signal/ref/butter.html>