|  |  |
| --- | --- |
| **Experiment Number** | **2** |
| **Date of Experiment** |  |
| **Date of Submission** |  |
| **Name** |  |
| **Roll Number** |  |
| **Section** | ECS-01 |

**Aim of The Experiment :-**

Analysis of a given signal in time and frequency domain using Discrete Fourier Transform (DFT)

using MATLAB.

**Equipment and Software Required:-**

The Equipment and Software required are as follows:

 MATLAB R2017 / 2019 / LabVIEW

 Kaggle ECG Database

**Code:**

% In this experiment we are doing signal processing on ecg signal

clc

close all;

clear all;

x=csvread("'ecg.1.csv'");

% y=x(1,:)

% z=x(:,1)

% w=x(1:5,1:3)

figure(1),

plot(x)

% subplot(122),

% stem(x(1:200));

xlabel('index','FontSize',12,'Fontweight','bold','Color','r');

ylabel('Amplitude','FontSize',12,'Fontweight','bold','Color','b');

xlim([1 3600]);

ylim([900 1225]);

title('ECG Plot(2204029)'),grid();

Fs = 1000; % Sampling frequency is 1000 Hz

t =(0:length(x)-1)/Fs; % Creates a time vector

% Calculate the PSD of the ECG signal

[pxx,freqs] = pwelch(x',[],[],[],Fs);

% Find cutoff frequency where power is reduced to half

half\_power = max(pxx)/2;

cutoff\_index = find(pxx>=half\_power, 1, 'last');

cutoff\_frequency = freqs(cutoff\_index);

% Print the cutoff frequency to the console

disp(['Cutoff frequency: ', num2str(cutoff\_frequency), ' Hz']);% Designing a low-pass filter with the cutoff frequency

[b, a] = butter(6, cutoff\_frequency/(Fs/2));

% Apply the filter to the ECG signal

filtered\_ECG = filter(b, a, x');

% Plotting filtered ECG signal

figure;

plot(t, filtered\_ECG);

title('Filtered ECG Signal');

xlabel('Time (s)','FontSize',12,'Fontweight','bold','Color','r');

ylabel('Amplitude','FontSize',12,'Fontweight','bold','Color','b');

title('ECG Filtered Plot(2204029)'),grid();

% Applying DFT on the filtered signal

N = length(filtered\_ECG);

f = (0:N-1)\*(Fs/N);

ECG\_FFT = fft(filtered\_ECG);

% Plotting the magnitude spectrum

figure;

stem(f(1:50), abs(ECG\_FFT(1:50)));

title('Magnitude Spectrum of Filtered ECG Signal');

xlabel('Frequency (Hz)','FontSize',12,'Fontweight','bold','Color','r');

ylabel('Magnitude','FontSize',12,'Fontweight','bold','Color','b');

title('Magnitude spectrum Plot(2204029)'),grid();

% Signal analysis through PSD

[Pxx, freqs] = pwelch(filtered\_ECG, [], [], [], Fs);

% Plotting PSD

figure;

plot(freqs, 10\*log10(Pxx));

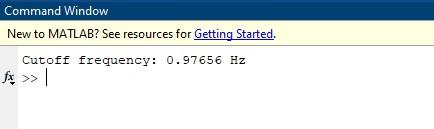
title('Power Spectral Density of Filtered ECG Signal');

xlabel('Frequency (Hz)','FontSize',12,'Fontweight','bold','Color','r');

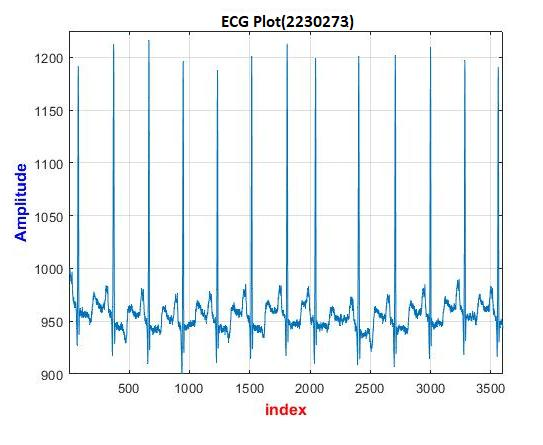
ylabel('Power/Frequency (dB/Hz)','FontSize',12,'Fontweight','bold','Color','b');

title('PSD Plot(2204029)'),grid();

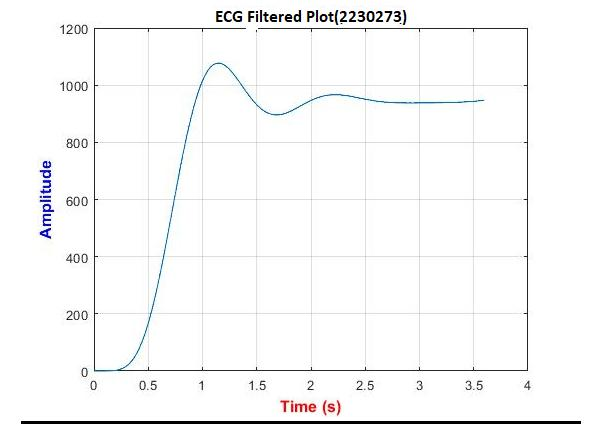
**Block diagram:**



**ECG PLOT**

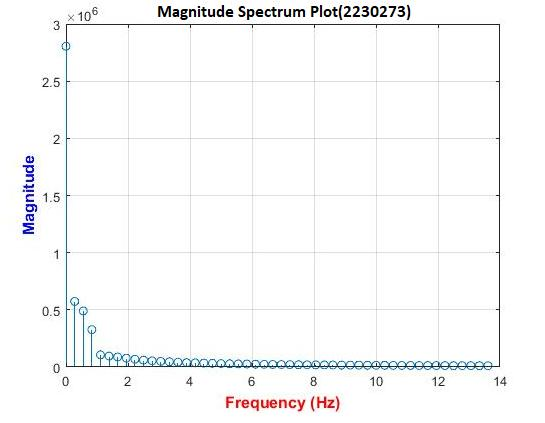


**ECG FILTERED PLOT**

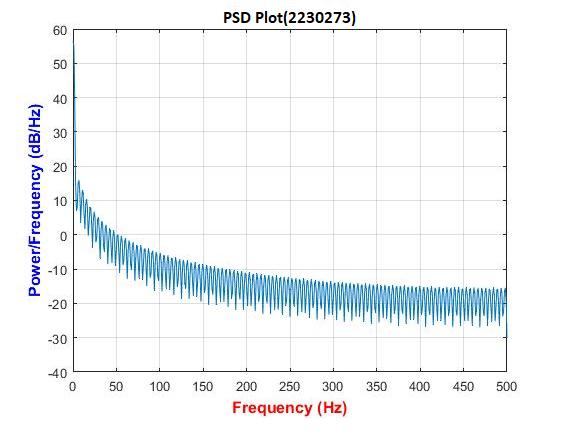


\

x10^6 **MAGNITUDE SPECTRUM PLOT**



**PSD FILTERED PLOT**



**Discussion or Inference of the experiment:**

The Kaggle ECG dataset is commonly used for tasks related to electrocardiogram

analysis. What specific inference or analysis are you looking to perform on this

dataset? Whether it's heartbeat classification, arrhythmia detection, or any other task,

please provide more details so I can assist you effectively.It includes recordings of

electrical activity of the heart and is often used for tasks like arrhythmia detection,

heartbeat classification, and more. Researchers and data scientists use this dataset todevelop and evaluate machine learning models for various cardiac-related

applications.

**Conslusion:**

In conclusion, this experiment taught us the fundamental of MATLABwith the

generation of different signals such as unit step, impulse signal, sine/cosine function,

exponential function and ramp function.

Analyzing a time signal in both time and frequency domains using the discrete

Fourier transform (DFT) in MATLAB allows for a comprehensive understanding of

the signal's characteristics. By examining the signal's time-domain behavior, you can

identify features such as amplitude, frequency, and duration. Transitioning to the

frequency domain through the DFT reveals the signal's frequency components and

their respective magnitudes. This insight is invaluable for detecting dominant

frequencies, harmonics, and noise. Ultimately, this dual-domain analysis aids in

making informed decisions about filtering, signal processing, and system behavior,

enhancing your ability to extract meaningful information from the given time signal.