Medipol EEE4210344 DSP

Spring 2020, PROJECT REPORT

by Dr. Tuncer Baykas, Istanbul Medipol University

Şeyma YILMAZ 62160010

Hilal ÇEVİK 62160014

BME

Şeyma YILMAZ 62160010
Hilal ÇEVİK 621600104

TABLE OF CONTENTS **Page**

Şeyma YILMAZ 62160010
Hilal ÇEVİK 621600104

1. ABSTRACT

The aim of this project is to use and compare different filter structures on AM type signal. The project includes software and hardware parts. Due to the current events, the software part of the project has been completed. Expected in the software part of the project, respectively; Finding AM type signal. Amplitude modulation is a form of modulation used for radio broadcasts for broadcast and two-way radio communication applications. It is used commonly because of to implement is simple. On the absence of AM type signal, the option offered in the project is to find the signal that contains noise. If signal is analog, converting the signal to digital is the second step. Then clear the signal using multiple filters. To discuss the effects of filters on the signal and to compare them.

2. INTRODUCTION

2.1 ECG Signal

The signal used in the project was chosen as the ECG signal. The ECG shows the plot of the bio-potential generated by the activity of the heart. PQRST should also be mentioned in this report because the ECG signal is selected. The P wave represents the depolarization of the left and right atrium. The QRS complex is a combination of three graphic deviations seen in a typical ECG. T wave follows the QRS complex and indicates ventricular repolarization.

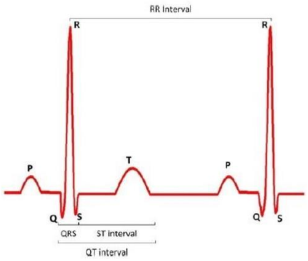

Figure 1 : ECG of a heart in normal sinus rhythm

2.2 AWGN ECG Signal

AM type ECG signal could not be reached in the available sources. Noise was added to the original ECG signal that was taken as ready. This matlab was made with the AWGN method. First data of signal was obtained. For a certain t time, it was lie on the x axis which is represented by time. Fs was chosen as 10 times of data length and it is observed for 10 second. Next step was adding noise. AWGN was added successfully by using awgn function. The important point was applying this noise base on the y axis. As it is observed in Figure 2, the original signal and noisy signal were plotted properly.

For this step,

```
val1=transpose(val);
t=0:10/length(val1):10;
t=t(1:length(val1))
figure, subplot(2,1,1);
plot(t,val1)
legend('Original Signal of ECG')
y = awgn(val1,10,'measured');
```

```
subplot(2,1,2);
plot(t,[val1 y])
legend('Signal with AWGN')
```

This code used. Data files in workspace. All files and codes are in DSP Project zip file.

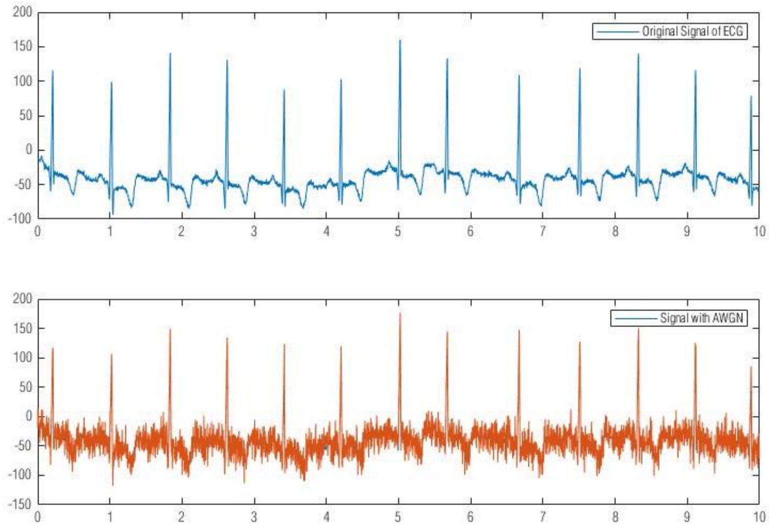

Figure 2 : Original and with AWGN Signal of ECG

2.3 Filter Structures

Selecting filters are required for the operation of the electronic circuit. The filter must be removed from a transmitted signal or a device or process that removes features. Most commonly, it should disable interfering signalsDigital filters are used for two general purposes. The first is the separation of the combined signals, the second is the restoration of somehow distorted signals. . Filters can be divided by their use, and how they are implemented. Digital filters can be implemented in two ways. By convolution (FIR) and by recursion (IIR) as in Figure 3.

FILTER USED FOR:	FILTER IMPLEMENTED BY: Convolution *Finite Impulse Response (FIR)*	Recursion *Infinite Impulse Response (IIR)*
Time Domain *(smoothing, DC removal)*	Moving average (Ch. 15)	Single pole (Ch. 19)
Frequency Domain *(separating frequencies)*	Windowed-sinc (Ch. 16)	Chebyshev (Ch. 20)
Custom *(Deconvolution)*	FIR custom (Ch. 17)	Iterative design (Ch. 26)

Figure 3 : Filter Classification

Moving Average Filter (Smoothing Filter)

The moving average is the most widely used filter in DSP for easier understanding and use, and because it is a digital filter. It is the duty to reduce noise. This makes it the first filter for time domain coded signals.

$$y[i] = \frac{1}{M}\sum_{j=0}^{M-1} x[i+j]$$

Formula 1 : Equation of the moving average filter.

Butterworth Filter

The Butterworth filters are one of the most commonly used digital filters for analyzing motion and for audio circuits. Butterworth 's digital implementations and other filters are also based on either the bilinear transform method or the matched Z-transform method, two different approaches to discrete an analog filter design. The matched Z-transform method is similar to the impulse invariance method for all-pole filters such as the Butterworth.

$$H_{(j\omega)} = \frac{1}{\sqrt{1+\varepsilon^2\left(\frac{\omega}{\omega_p}\right)^{2n}}}$$

Formula 2 : Butterworth Filter Design with a Low Pass Butterworth

Chebyshev Filter

Chebyshev filters are used to separate one band of frequencies from another. The designing filters is based on a technique called the z-transform.

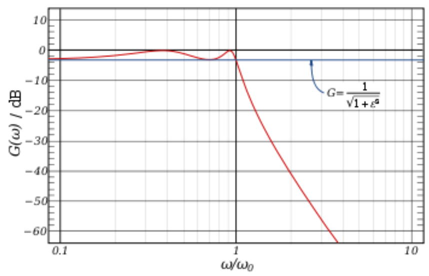

Figure 4 : The frequency response of a fourth-order type I Chebyshev low-pass filter

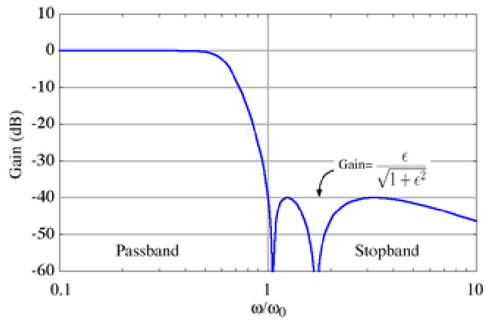

Figure 5 : The frequency response of a fifth-order type II Chebyshev low-pass filter

2.4 Applying Filter Structures to Signal on Matlab

1- Moving Average Filter (Smoothing Filter)

```
val1=transpose(val);
t=0:10/length(val1):10;
t=t(1:length(val1))
figure, subplot(3,1,1);
plot(t,val1)
legend('Original Signal of ECG')
y = awgn(val1,10,'measured');
subplot(3,1,2);
plot(t,[val1 y])
legend('Signal with AWGN')

%   Implementation of Moving Average Filter or Smoothing Filter
% Algorithm:
%                X(n-1) + X(n) + X(n+1)
%    Y(n) =      ----------------------
%                          3
%               here, X(n) = noise effected signal
%                     Y(n) = Smoothed or averaged signal
% therefore,
t1= 0:0.001:1;
[m,n] = size(t);
y2 = zeros(m,n);
for i = 2:(n-1)
    y2(i) = (y(i-1) + y(i) + y(i+1))/3;
end
subplot(3,1,3);
plot(t,y2); title('Smoothed Signal');
```

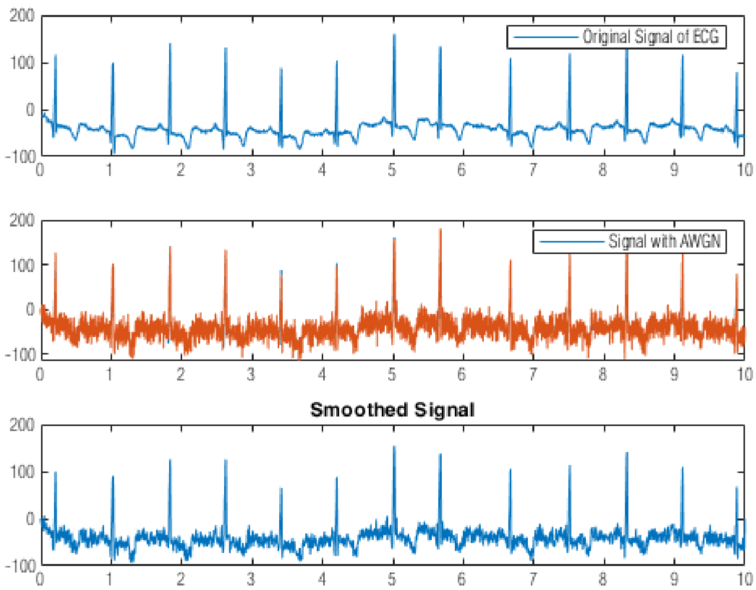

Figure 6 : Moving Average Filter or Smoothing Filter

Şeyma YILMAZ 62160010
Hilal ÇEVİK 621600104

2- Butterworth Filter

```
val1=transpose(val);
t=0:10/length(val1):10;
t=t(1:length(val1))
figure, subplot(3,1,1);
plot(t,val1)
legend('Original Signal of ECG')
y = awgn(val1,10,'measured');
subplot(3,1,2);
plot(t,[val1 y])
legend('Signal with AWGN')

Rp = 0.3; %ripple
Rs = 40;
Wp = [0.4];
Ws = [0.6];

%Butterworth Filter

FilterType = 'Butterworth Filter';
[N, Wpe] = buttord(Wp, Ws, Rp, Rs);

[b,a] = butter(N, Wpe);
snr = 22;
x1_n = awgn(y,snr);
y2 = filter(b,a,x1_n);
subplot(3,1,3);
plot(y2); title('Butterworth Filter');
```

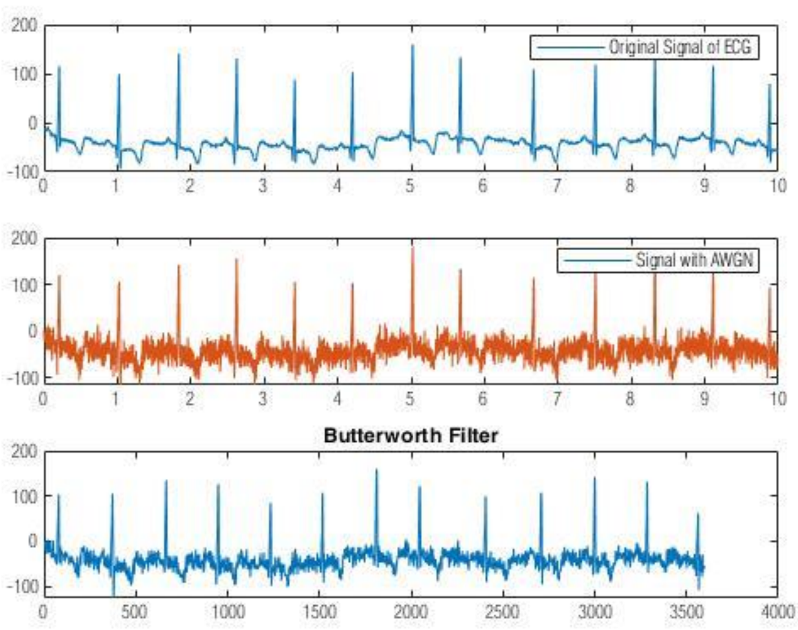

Figure 7 : Butterworth Filter

Şeyma YILMAZ 62160010
Hilal ÇEVİK 621600104

3- Chebyshev Type 1 Filter

```
val1=transpose(val);
t=0:10/length(val1):10;
t=t(1:length(val1))
figure, subplot(3,1,1);
plot(t,val1)
legend('Original Signal of ECG')
y = awgn(val1,10,'measured');
subplot(3,1,2);
plot(t,[val1 y])
legend('Signal with AWGN')

Rp = 0.3;
Rs = 40;
Wp = [0.4];
Ws = [0.6];

% Chebyshev Type 1 Filter

FilterType = 'Chebyshev Type Filter';
[N1, Wpe1] = cheb1ord(Wp, Ws, Rp, Rs);

[b,a] = cheby1(N1, Rp, Wpe1, 'low');
snr = 22;
x1_n = awgn(y,snr);
y2 = filter(b,a,x1_n);
subplot(3,1,3);
plot(y2); title('Chebyshev Type 1 Filter');
```

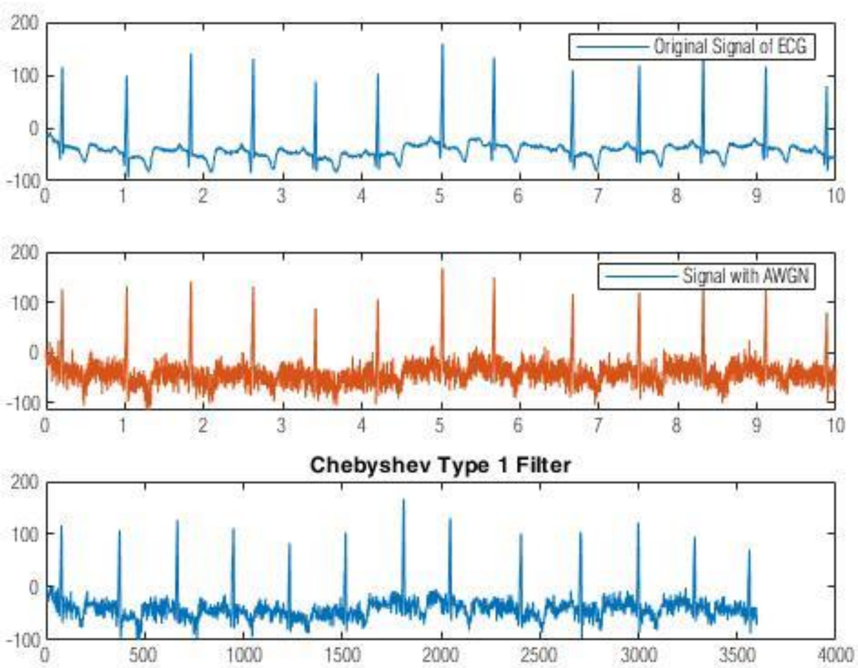

Figure 8 : Chebyshev Type 1 Filter

4- Chebyshev Type 2 Filter

```
val1=transpose(val);
t=0:10/length(val1):10;
t=t(1:length(val1))
figure, subplot(3,1,1);
plot(t,val1)
legend('Original Signal of ECG')
y = awgn(val1,10,'measured');
subplot(3,1,2);
plot(t,[val1 y])
legend('Signal with AWGN')

Rp = 0.3;
Rs = 40;
Wp = [0.4];
Ws = [0.6];

% Chebyshev Type 2 Filter

FilterType = 'Chebyshev Type Filter';
[N1, Wpe1] = cheb2ord(Wp, Ws, Rp, Rs);

[b,a] = cheby2(N1, Rp, Wpe1, 'low');
snr = 22;
x1_n = awgn(y,snr);
y2 = filter(b,a,x1_n);
subplot(3,1,3);
plot(y2); title('Chebyshev Type 2 Filter');
```

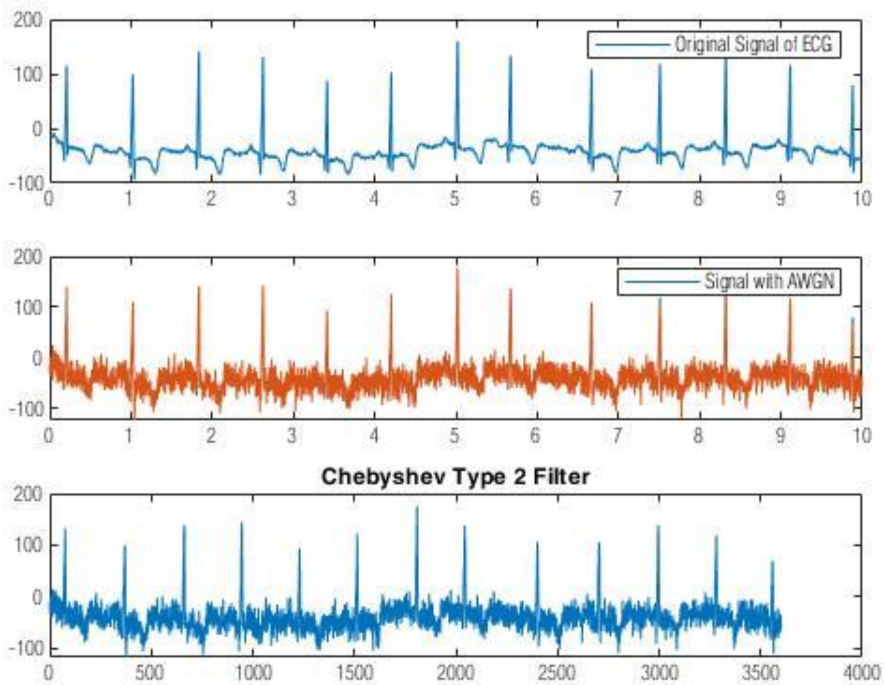

Figure 9 : Chebyshev Type 2 Filter

3. DISCUSSION

3.1 About the Working of Filters

We were responsible for the software part of the project. Different filter structures were observed in Matlab environment. But the desired path could not be followed in the project. Transfer function coefficient was the main part of the project. We had to think that we were adding using the structures in the parallel filter structures. Cascade and serial could also be but parallel filter structers are more suitable for FPC. In this way, too many branches can be made. These parts could not be done in the project.

The application of the Moving Average Filter (or Smoothing Filter) was clearly observed. The filter worked. It cleans the noise ECG signal as much as possible and output it.This was the best working filter.

The application of the Butterworth Filter was clearly observed. The filter worked. It cleans the noise ECG signal as much as possible and output it. It does not show clear result.

The application of the Chebyshev Type 1 Filter was clearly observed. The filter worked. It cleans the noise ECG signal as much as possible and output it. It does not show clear result too.

The application of the Chebyshev Type 2 Filter was clearly observed. The filter worked. It cleans the noise ECG signal as much as possible and output it. It does not show clear result too.

3.2 Comparing Filter Structures

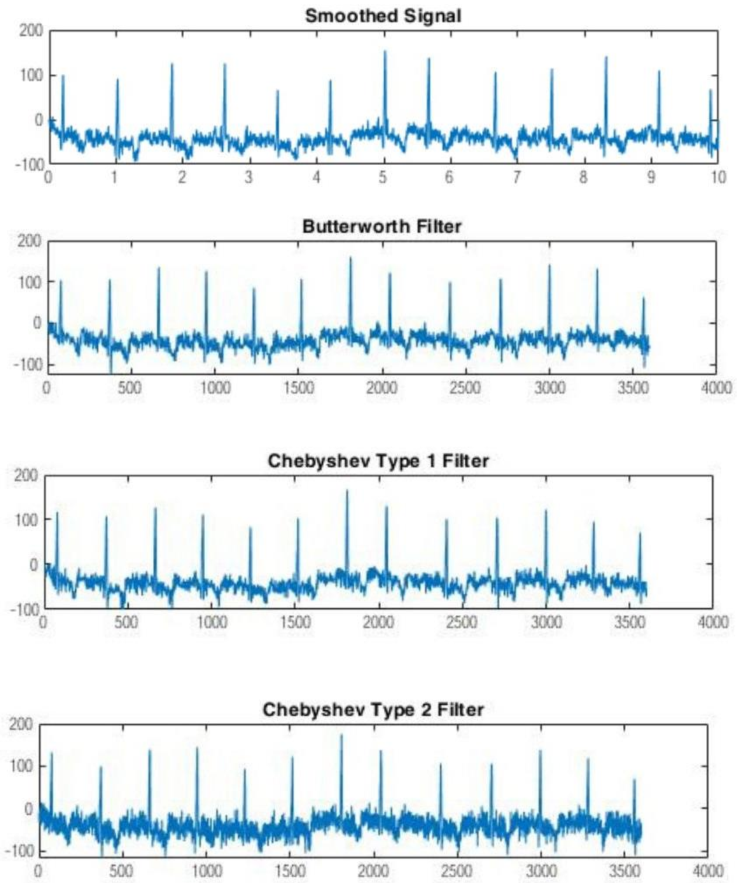

Figure 10 : All Filters

If their efficiency is compared;

Moving Average> Butterworth Filter> Chebyshev Type 1 Filter> Chebyshev Type 2 Filter

4. CONCLUSION

The purpose of this project is to use and compare different filter structures on AM type signal. The software part of the project has been completed. Noise was added on the signal because the AM type ECG signal was not found. This signal was cleared using four different filters. The effects of filters on the signal are discussed and compared. When the yield comparison was made as a result of the project, it was determined that the best filter was Moving Average.