A

Project Report on

“ IIR DIGITAL FILTER ”

Submitted By

Md. Mahfuzur Rahman (190639)

Md. Imran Hossain (190622)

Md. Mehedi Hasan (190611)

Md. Mofazzol Hossain (180639)

Md. Sharif Ahammed (190636)

Under The Guidance of

Dr. Md. Imran Hossain

Associate Professor

Department of Information and Communication Engineering

Pabna University of Science and Technology

Pabna, Bangladesh

In Partial Fulfillment of the Requirement of the Subject in

Digital Signal Processing

LAB- IIR DIGITAL FILTER USING BUTTERWORTH METHOD Using MATLAB

❖Introduction

This laboratory studies the use of Butterworth method and bilinear transformation to design IIR filters. Filters are an essential tool in our complex world of mixed signals — both electronic and otherwise. IIR (infinite impulse response) filters are generally chosen for applications where linear phase is not too important and memory is limited. They have been widely deployed in audio equalization, biomedical sensor signal processing, IOT smart sensors and high-speed telecommunication/RF applications.

❖Objective

➢ Educational

1. Learn the four general filter types: High-pass, Low-pass, Band pass, and Band stop.

2. Learn to alter filter type by changing the specifications.

3. Learn magnitude and phase spectrum of IIR filters.

4. Design simple filter.

➢ Experimental

1. Calculate and measure order and cutoff frequency for IIR filters.

2. Design simple IIR low-pass filter.

3. Generate and interpret spectrum plots for filters.

❖Prelab Preparation:

1. Study the Background section of this Laboratory exercise.
2. Study textbook regarding origin and meaning of IIR filters.

❖Equipment needed

1. Personal computer
2. MATLAB software.

5 Background

The IIR filter can realize both the poles and zeroes of a system because it has a rational transfer function, described by polynomials in z in both the numerator and the denominato.The transfer functoin of IIR filter of order N is given by

H[z] = ∑ M k=0 bkz −k ∑ N k=1 akz −k

The difference equation for such a system is described by the following:

y(n) = ∑ M k=0 bkx(n − k) − ∑ N k=1 aky(n − k)

M and N are order of the two polynomials bk and ak are the filter coefficients.

These filter coefficients are generated using MATLAB. IIR filters can be expanded as infinite impulse response filters. In designing IIR filters, cutoff frequencies of the filters should be mentioned. The order of the filter can be estimated using butter worth polynomial. That’s why the filters are named as butter worth filters. Filter coefficients can be found and the response can be plotted.

❖Procedure

1. Click on the MATLAB Icon on the desktop.

2. MATLAB window opens.

3. Click on the 'FILE' Menu on menu bar.

4. Click on NEW M-File from the file Menu.

5. An editor window open, start typing commands.

6. Now SAVE the file in directory.

7. Then Click on DEBUG from Menu bar and Click Run.

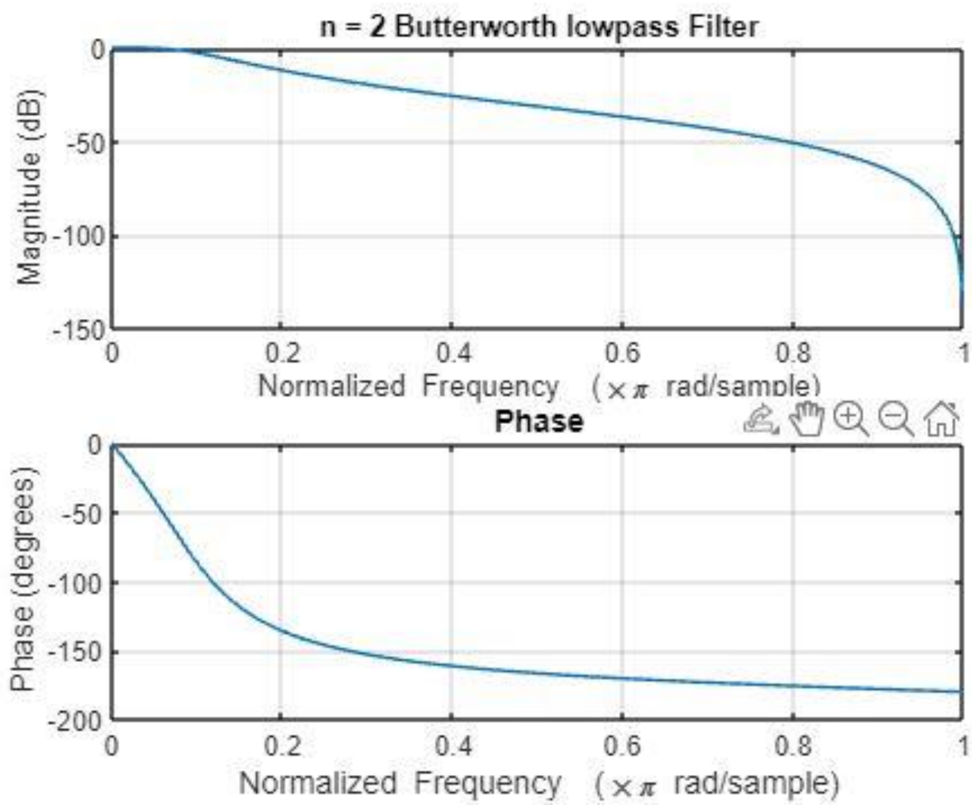
❖Program

% Design a digital lowpass Butterworth filter with pass cut-off frequencies 200 hz and 500 hz respectively.The pass band and stop band attenuations are -5db and -12db respectively. The sampling frequency is 5000hz.

```
clc;
 clear all;
 close all;
% pass band cut off frequency
 fp = 200;
% stop band cut off frequency
 fs = 500;
 % sampling frequency
Fs=5000;
 % corresponding digital frequencies
 Wp=2*fp/Fs;
 Ws=2*fs/Fs;
 % pass band ripple
Rp = 5;
 % stop band ripple
Rs = 12;
 % Butterworth filter order and cutoff frequency
 [n,Wn] = buttord(Wp,Ws,Rp,Rs);
```

```
% butter(n,Wn) returns the transfer function
coefficients of an nth-order lowpass digital
Butterworth filter with normalized cutoff frequency Wn.
[b,a] = butter(n,Wn);
 % freqz - Frequency response of digital filter
freqz(b,a)
% sprintf is used to format data into string or
character vector
 title(sprintf('n = %d Butterworth lowpass Filter',n))
```

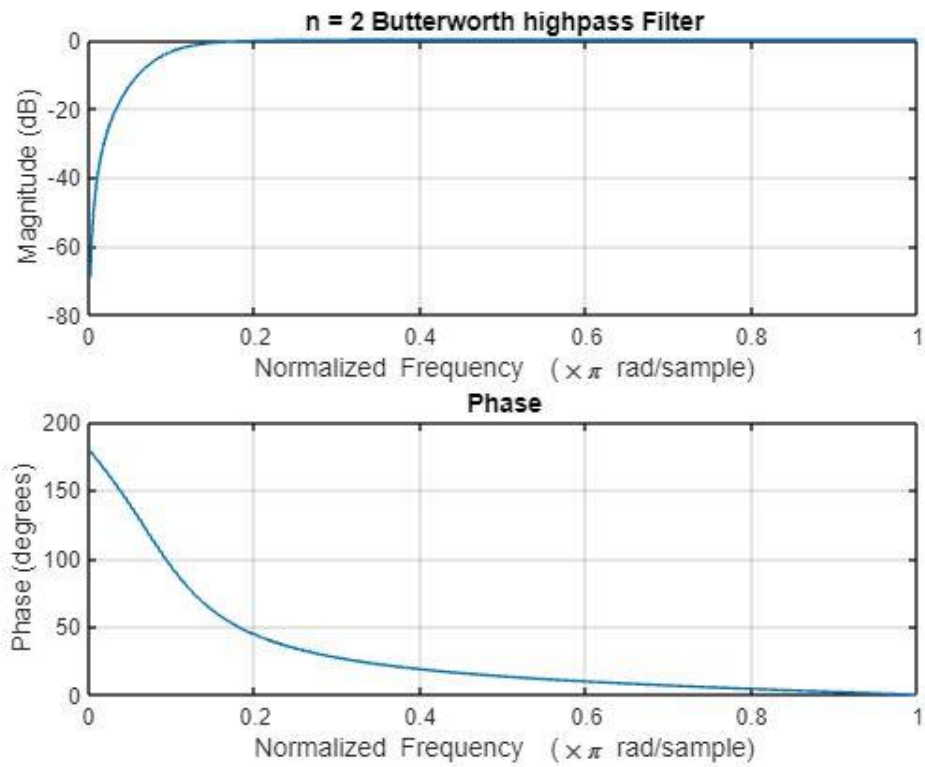
❖Output and waveforms



% Design a digital highpass Butterworth filter with pass cut-off frequencies 200 hz and 500 hz respectively.The pass band and stop band attenuations are -5db and -12db repectively. The sampling frequency is 5000hz.

```
clc;
 clear all;
close all;
% pass band cut off frequency
 fp = 200;
 % stop band cut off frequency
 fs = 500;
 % sampling frequency
Fs=5000;
% corresponding digital frequencies
 Wp=2*fp/Fs; Ws=2*fs/Fs;
 % pass band ripple
 Rp = 5;
 % stop band ripple
 Rs = 12;
 % Butterworth filter order and cutoff frequency
[n,Wn] = buttord(Wp,Ws,Rp,Rs);
 % butter(n,Wn) returns the transfer function coefficients
of an nth-order highpass digital Butterworth filter with
normalized cutoff frequency Wn.
 [b,a] = butter(n,Wn,'high');
% freqz - Frequency response of digital filter
 freqz(b,a)
% sprintf is used to format data into string or character
vector
 title(sprintf('n = %d Butterworth highpass Filter',n));
```

❖Output and waveforms

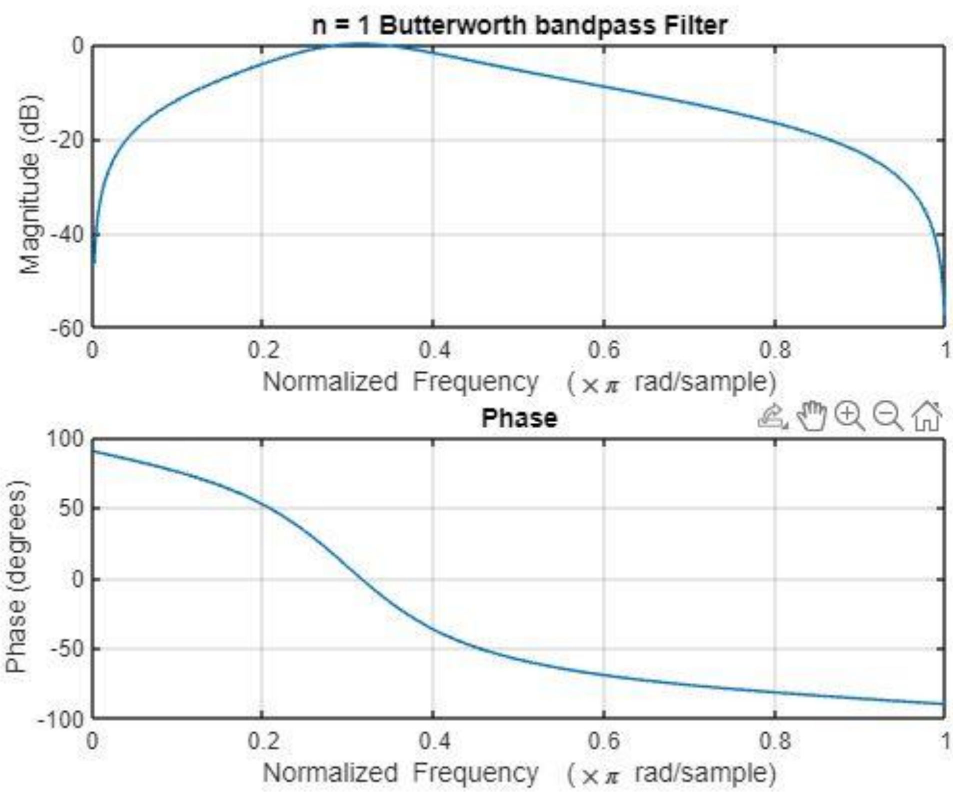


%Design a Butterworth digital band pass filter with pass band and stop band attenuation -3db and -10db respectively. The pass band cut-off frequencies are 600hz and 1000hz, and stop band frequencies are 300 hz and 1600 hz. The sampling frequency is 5000 hz.

```
clc;
 clear all;
close all;
% pass band cut off frequency
 fp = [600 1000];
% stop band cut off frequency
fs = [300 1600];
 % sampling frequency
 Fs=5000;
 % corresponding digital frequencies
```

```
% pass band ripple
 Rp = 3;
 Rs=10;
 Wp=2*fp/Fs;
 Ws=2*fs/Fs;
% pass band ripple Rp = 3;
 % stop band ripple Rs = 10;
% Butterworth filter order and cutoff frequency
 [n,Wn] = buttord(Wp,Ws,Rp,Rs);
% butter(n,Wn) returns the transfer function coefficients of
an nth-order bandpass digital Butterworth filter with
normalized cutoff frequency Wn.
 [b,a] = butter(n,Wn, 'bandpass');
 % freqz - Frequency response of digital filter freqz(b,a)
% sprintf is used to format data into string or character
vector
title(sprintf('n = %d Butterworth bandpass Filter',n))
```

❖Output and waveforms



%Design a Butterworth digital band stop filter with pass band and stop band attenuation -3db and -10db respectively. The pass band cut-off frequencies are 600hz and 1000hz, and stop band frequencies are 300 hz and 1600 hz. The sampling frequency is 5000 hz.

```
clc;
clear all;
close all;
% pass band cut off frequency fp = [600 1000];
% stop band cut off frequency
 fp = [600 1000];
 fs = [300 1600];
 % sampling frequency
 Fs=5000;
 % corresponding digital frequencies
 %Wp=2*fp/Fs;
 Wp=2*fp/Fs;
 Ws=2*fs/Fs;
 % pass band ripple
 Rp = 3;
 Rs = 10;
 % stop band ripple Rs = 10;
% Butterworth filter order and cutoff frequency
 [n,Wn] = buttord(Wp,Ws,Rp,Rs);
 % butter(n,Wn) returns the transfer function coefficients
of an nth-order bandstop digital Butterworth filter with
normalized cutoff frequency Wn.
 [b,a] = butter(n,Wn,'stop');
 % freqz - Frequency response of digital filter freqz(b,a)
% sprintf is used to format data into string or character
vector
 title(sprintf('n = %d Butterworth bandstop Filter',n));
```

❖Output and waveforms