Laboratory Report Cover Sheet

SRM Institute of Science and Technology College of Engineering and Technology Department of Electronics and Communication Engineering
18ECC204J DIGITAL SIGNAL PROCESSING **Fifth Semester, 2022-23 (Odd semester)**

Name :

Register No. :

Day / Session :

Venue :

Title of Experiment :

Date of Conduction :

Date of Submission :

Particulars	Max. Marks	Marks Obtained
Prelab and Post lab	10	
Lab Performance	10	
Simulation and results	10	
Total	30	

REPORT VERIFICATION

Staff Name :

Signature :

Experiment 11(a) Design of Digital Butterworth filter using Bilinear Transformation

Aim: To Design an digital Butterworth filter using Bilinear Transformation method

Software Requirement:SCILab

Theory: Butterworth filters are optimal in the sense of having a *maximally flat amplitude response*, as measured using a Taylor series expansion .The trivial filter H(Z)=1 has a perfectly flat amplitude response, but that's an all pass, not a low pass filter. Therefore, to constrain the optimization to the space of low pass filters, we need *constraints* on the design, such as H(1)=1 and H(-1)=0. That is, we may require the dc gain to be 1, and the gain at half the sampling rate to be 0.

It turns out Butterworth filters are much easier to design as *analog filters* which are then converted to digital filters. This means carrying out the design over the s plane instead of the z plane, where the s plane is the complex plane over which analog filter transfer functions are defined. The analog transfer function $H_a(s)$ is very much like the digital transfer function H(z), except that it is interpreted relative to the analog frequency axis $s = j\omega_a$ (the ``$j\omega$ axis'') instead of the digital frequency axis $z = e^{j\omega_d T}$ (the ``unit circle''). In particular, analog filter poles are stable if and only if they are all in the *left-half* of the s plane, *i.e.*, their real parts are *negative*.

Digital Butterworth Filter using Bilinear Transformation

The bilinear -transform is a mathematical transformation from the -domain to the -domain which preserves the frequency characteristics and is defined by:

$$s = \frac{2}{T}\frac{1 - z^{-1}}{1 + z^{-1}} \quad \text{where } T = \text{ sampling period}$$

The bilinear transformation gives a non-linear relationship between analog W_a frequency and digital frequency. W_d

$$\omega_a = \frac{2}{T}\tan\frac{\omega_d T}{2}$$

Algorithm:

Step 1: From the given specifications, find prewarping analog frequencies, using the formula

$$\omega_a = \frac{2}{T}\tan\frac{\omega_d T}{2}$$

Step2: Using the analog frequencies, find Ha(s) of the analog filter.

Step3:Select the sampling rate of the digital filter, Call it 'T' seconds per sample.

Step 4: Substitute

$$s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}} \quad \text{where } T = \text{ sampling period}$$

into the transfer function found in step 2.

Program:

Using Bilinear transformation, design a high pass filter ,monotonic in pass band with cutoff frequency of 1000 and down 10dB at 350 Hz.The sampling frequency is 5000Hz.

```
//To design digital Butterworth filter using bilinear transformation

clear all;
clc;
close;
ap=input('enter the value of ap in dB');
as=input('enter the value of as in dB Hz');
fp=input('enter the value of fp in Hz');
fs=input('enter the value of fs in Hz');
f=input('enter the value of f ');

T=1/f;
wp=2*%pi*fp;
ws=2*%pi*fs;
op=2/T*tan(wp*T/2);
os=2/T*tan(ws*T/2);
N=log(sqrt((10^(0.1*as)-1)/(10^(0.1*ap)-1)))/log(op/os);
disp(ceil(N));
s=%s;
HS=1/(s+1);
oc=op;
HS1=horner(HS,oc/s);
```

```
disp(HS1,'Normalized transfer function,H(s)=');
z=%z;
HZ=horner(HS,(2/T)*(z-1)/(z+1));
disp(HZ,'H(z)=');
```

Simulation Output:

nter the value of ap in dB 3

enter the value of as in dB Hz 10

enter the value of fp in Hz 1000

enter the value of fs in Hz 350

enter the value of f 5000

```
1.

     s
------------
7265.4253 +s

"Normalized transfer function,H(s)="

    1 +z
-------------
-9999 +10001z

"H(z)="
```

\

Experiment 11(b) Design of digital Butterworth Filter using Impulse Invariant Method

Aim: To design of digital Butterworth Filter using Impulse Invariant Method

Software Requirement: SCI Lab

Theory: **Design procedure using impulse invariance:**

Using the impulse invariance design procedure, the relation between frequency in the continuous-time and discrete-time domains is

$$\Omega_p = \omega_p / T$$

$$\Omega_s = \omega_s / T$$

Express the analog transfer function as the sum of single pole filters:

$$H(s) = \sum_{k=1}^{N} \frac{A_k}{(s - s_k)}$$

Compute the z-transform of the digital filter by using the formula

$$H(z) = \sum_{k=1}^{N} \frac{A_k}{\left(1 - e^{s_k T} z^{-1}\right)}$$

Algorithm:

Step 1: From the given specifications, find prewarping analog frequencies, using the formula

$$\Omega_p = \omega_p / T$$

$$\Omega_s = \omega_s / T$$

Step2: Using the analog frequencies, find Ha(s) of the analog filter.

Step3:Select the sampling rate of the digital filter, Call it 'T' seconds per sample.

Step4: Express the analog transfer function as the sum of single pole filters:

$$H(s) = \sum_{k=1}^{N} \frac{A_k}{(s - s_k)}$$

Step5:Compute the z-transform of the digital filter by using the formula

$$H(z) = \sum_{k=1}^{N} \frac{A_k}{\left(1 - e^{s_k T} z^{-1}\right)}$$

Program:

```
clear all;
clc ;
 close ;
 s=%s;
 T =1;
HS =(2) /(s ^2+3* s +2) ;
 elts = pfss (HS);
disp (elts , ' Fa c t o r i z e d HS=' );

//The p o l e s a s s o c i a t e d a r e 2 and 1
 p1 = -2;
 p2 = -1;

 z=%z;
 HZ =(2/(1 - %e ^( p2*T)*z^( -1))) -(2/(1 - %e ^( p1*T)*z^( -1)));

 disp (HZ , 'HZ=' );
```

Prelab Questions:

1. What are the properties that are maintained same in the transfer of analog filter into a digital filter?
2. What is meant by impulse invariant method of designing IIR filter?
3. What are the properties of the bilinear transformation?

Postlab Questions:

1. An analog filter has a transfer function H(s)= $\frac{10}{s^2+7s+10}$. Design a digital filter equivalent tom this using impulse invariant method for T=0.2 sec.
2. For the analog transfer fuction, H(s)= $\frac{2}{s^2+3s+2}$,determine H(z) using bilinear transformation if (a) T=1 second (b) T=0.1 second.