Scilab Manual for Digital Signal Processing Lab by Dr Veena Hegde Instrumentation Engineering B.M.S College Of Engineering[1]

Solutions provided by
Dr Veena Hegde
Instrumentation Engineering
B.m.s College Of Engineering

November 25, 2026

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Scilab Manual and Scilab codes written in it can be downloaded from the "Migrated Labs" section at the website http://scilab.in

Contents

List of Scilab Solutions 3

1 Design and Testing of a Digital Butterworth Low pass filter with cutoff of 5 KHz to filter an input (.wav) file 5

2 Design a digital Butterworth low pass filter to band limit a sine wave up to 4 KHz, by considering input as 'tone' 9

3 Illustrate the working of a digital low pass filter by taking audio data as input and allow frequency up to 2 KHz 13

4 Design and test the working of a High Pass FIR filter using Hamming window by taking a high frequency signal as input 17

5 Design and test the working of a Butterworth band pass filter by giving a time domain input signal 20

6 Design and test band pass, Butterworth filter for given specifications which includes Edge frequencies, Ripple and Attenuation 24

List of Experiments

Solution 1.1	DSP Lab Migration	5
Solution 2.2	DSP Lab Migration	9
Solution 3.3	DSP Lab Migration	13
Solution 4.4	DSP Lab Migration	17
Solution 5.5	DSP lab migration	20
Solution 6.6	DSP lab migration	24

List of Figures

1.1 DSP Lab Migration 7
1.2 DSP Lab Migration 8

2.1 DSP Lab Migration 11
2.2 DSP Lab Migration 12

3.1 DSP Lab Migration 15
3.2 DSP Lab Migration 16

4.1 DSP Lab Migration 18
4.2 DSP Lab Migration 18

5.1 DSP lab migration 21
5.2 DSP lab migration 23

6.1 DSP lab migration 25
6.2 DSP lab migration 26

Experiment: 1

Design and Testing of a Digital Butterworth Low pass filter with cutoff of 5 KHz to filter an input (.wav) file

Scilab code Solution 1.1 DSP Lab Migration

```
1
2 // Exp 1: Design Low Pass Filter as per the given
     specification and test the working by taking an
     input sound signal.
3 // Enter cutoff freq in Hz fc = 5
4
5 // Version : Scilab 5.2.2
6 // Operating Syatem : Ubuntu 16.04 LTS
7
8 clc;
     //clear console
9 clear;
10 xdel(winsid());
11 fc=input('Enter cutoff freq in Hz fc =')
     //Cutoff frequency
```

```
12 fs=11025;
13 n=11;
     //Filter order
14 Fp=2*fc/fs;
15 [Hz]=iir(n,'lp','butt',[Fp/2,0],[0,0])
16 [p,z,g]=iir(n,'lp','butt',[Fp/2,0],[0,0])
     //Filter design
17 [Hw,w]=frmag(Hz,256);
18 figure(1)
19 subplot(2,1,1)
20 plot(2*w,abs(Hw));
21 xlabel('Normalized Digital frequency w->')
22 ylabel('magnitude');
23 title('Magnitude response of IIR filter')
24 xgrid(1)
25 subplot(2,1,2)
26 plot(2*w*fs,abs(Hw));
27 xlabel('Analog Frequency in Hz f ---->')
28 ylabel('Magnitude |H(w)|=')
29 title('Magnitude Response of IIR LPF')
30 xgrid(1)
31
32 [y,Fs]=wavread("meow.wav")
     //Reading input sound signal
33 figure(2)
34 subplot(2,1,1)
35 plot(y)
36 title('Input signal waveform');
37 xlabel('Frequency-->');
38 ylabel('Magnitude-->');
39 playsnd(y)
40
41 outlo=filter(z,abs(p),y);                    //
     Passing accquired signal through desired filter
42 subplot(2,1,2)
43 plot(outlo)
44 title('Output signal waveform after filtering')
45 xlabel('Frequency-->');
```

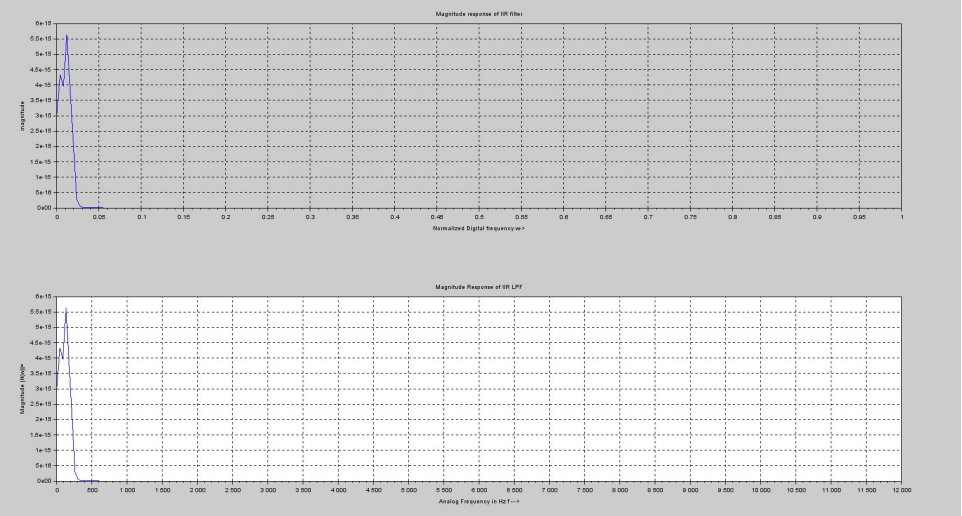

Figure 1.1: DSP Lab Migration

```
46 ylabel('Magnitude-->');
```

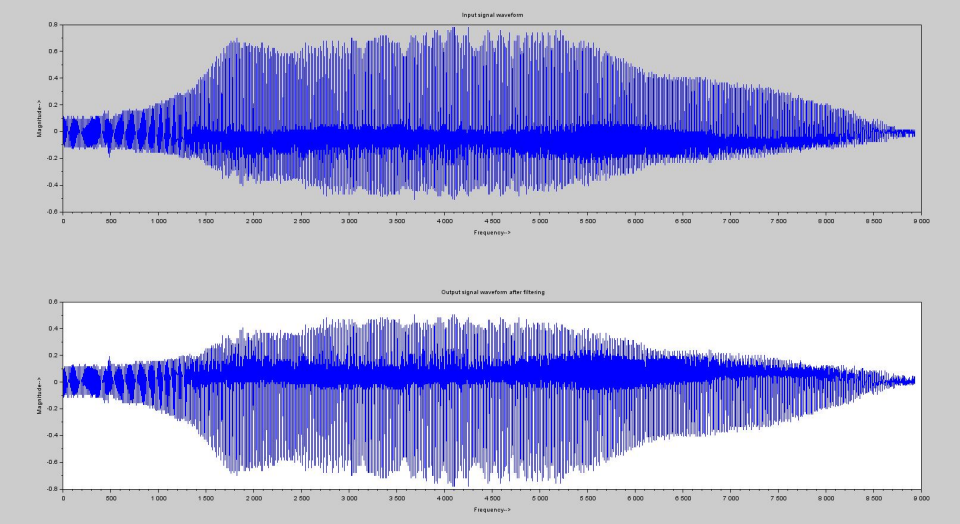

Figure 1.2: DSP Lab Migration

Experiment: 2

Design a digital Butterworth low pass filter to band limit a sine wave up to 4 KHz, by considering input as 'tone'

Scilab code Solution 2.2 DSP Lab Migration

```
1 //Program to design a Butterworth Low pass filter to
     Band limit a sine wave up to 4kHz . Taking the
    input as a     tone     .
2 // Version : Scilab 5.2.2
3 // Operating Syatem : Ubuntu 16.04 LTS
4 clc;
5 clear;
6 xdel(winsid());
7 Fc =4000;                                    // Cut-
    off frequency
8 Fs =44100;                                   //
    Sampling frequency
9 N =8 ;                                       // Order
10 Fp = 2*Fc/Fs;                               //Pass
    band edge frequency
```

```
11 [Hz]=iir(N,'lp','butt',[Fp/2,0],[0,0])
12 [p,z,g]=iir(N,'lp','butt',[Fp/2,0],[0,0])    //
      digital IIR Butterworth Filter
13 [Hw,w] = frmag(Hz ,256);
14
15 // Plotting the filter design
16 figure(1)
17 plot(2*w,abs(Hw));
18 xlabel("Digital Frequency  Normalized (w)")
19 ylabel("Magnitude")
20 title("Magnitude Response of Butterworth filter ")
21 xgrid(1)
22
23 [y,Fs]=wavread("tone1k.wav")                    //
      Reading the y
24 figure(2)
25 subplot(2,1,1)
26 plot(y)
27 title('Input signal waveform before filtering');
28 xlabel('Frequency');
29 ylabel('Magnitude');
30 playsnd(y)
31 L=length(y)
32
33 outlow=filter(z,abs(p),y);
34 subplot(2,1,2)
35 plot(outlow)
36 title('Output signal waveform after filtering')
37 xlabel('Frequency');
38 ylabel('Magnitude');
39
40 playsnd(outlow)
41
42 Da = fft(y,-1);
43 Pyy = (1/L)*(abs(Da).^2); //Peridogram Estimate
```

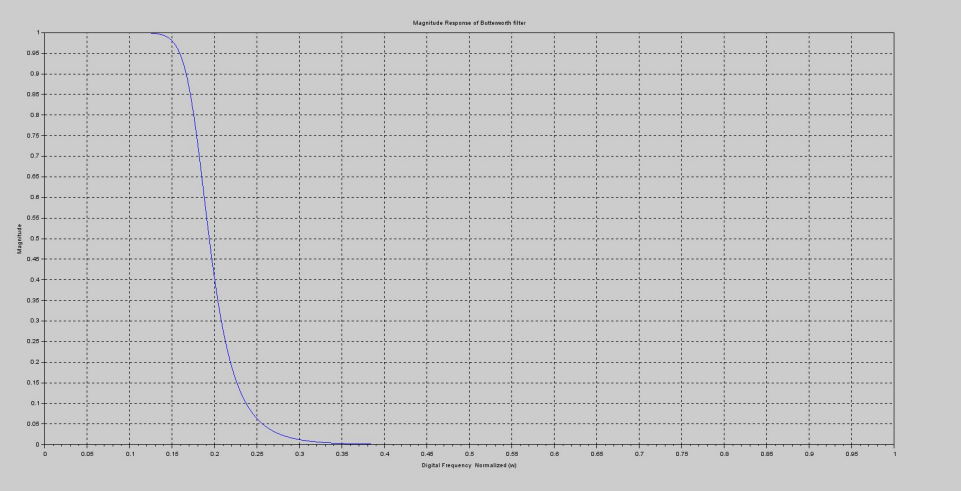

Figure 2.1: DSP Lab Migration

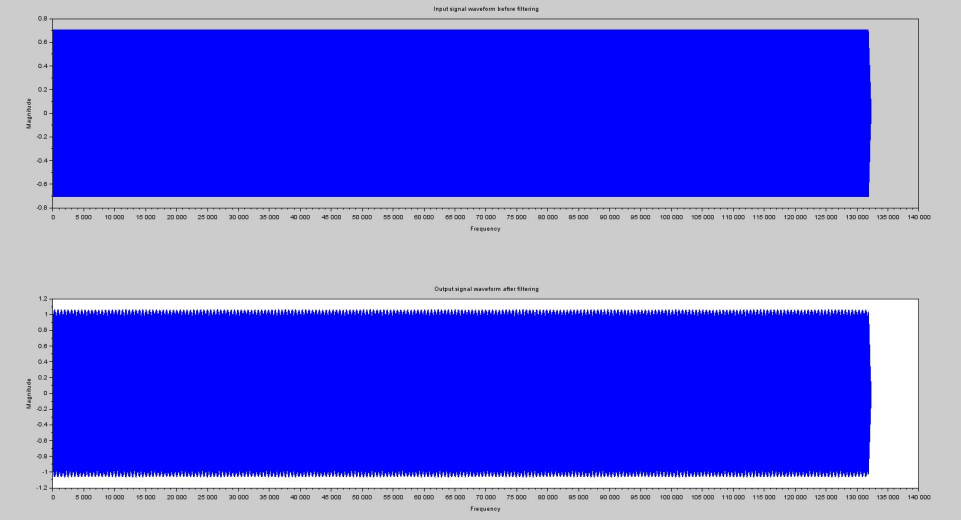

Figure 2.2: DSP Lab Migration

Experiment: 3

Illustrate the working of a digital low pass filter by taking audio data as input and allow frequency up to 2 KHz

Scilab code Solution 3.3 DSP Lab Migration

```
1 // Design a filter using Butterworth polynomial for
     the following specifications:
2 //1.    Order : 7
3 //2.    Cut-off frequency : 2.5 kHz
4
5 // Version : Scilab 5.2.2
6 // Operating Syatem : Ubuntu 16.04 LTS
7
8 clc;
9 clear;
10 xdel(winsid());
11 Fc =2500;

     // Cut-off frequency
12 Fs =44100;
```

```
    // Sampling frequency
13 N =7 ;

    // Order of the filter
14 Fp = 2*Fc/Fs;

    //Pass band edge frequency
15 [Hz]=iir(N,'lp','butt',[Fp/2,0],[0,0])
16 [p,z,g]=iir(N,'lp','butt',[Fp/2,0],[0,0])
17 [Hw,w] = frmag(Hz ,256);
18
19 figure(1)
20 plot(2*w,abs(Hw));
21 xlabel("Digital Frequency  Normalized (w)")
22 ylabel("Magnitude")
23 title("Magnitude Response of Butterworth filter ")
24 xgrid(1)
25
26 [y,Fs]=wavread("tone1k.wav")

    // Reading the .wav file
27
28 outlow=filter(z,abs(p),y);
29
30 psd1=pspect(100,200,'re',y)
31 figure(2)
32 subplot(2,1,1)
33 plot(psd1)

    //Plotting power spectral density of input
34 title('Input signal power spectral density')
35 xlabel('Frequency');
36 ylabel('Magnitude');
37
38 psd2=pspect(100,200,'re',outlow)
39 subplot(2,1,2)
40 plot(psd2)
```

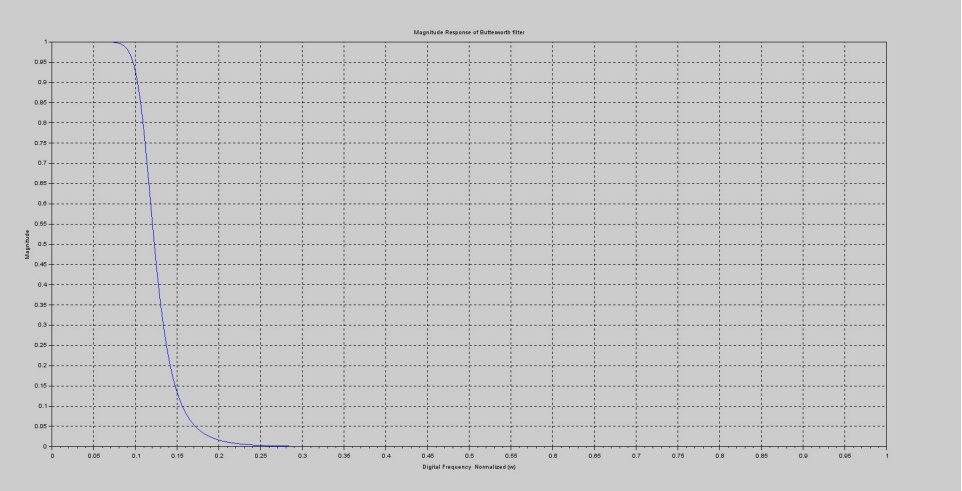

Figure 3.1: DSP Lab Migration

```
    //Plotting power spectral density of output
41 title('Filtered signal power spectral density')
42 xlabel('Frequency');
43 ylabel('Magnitude');
44
45 playsnd(outlow)
```

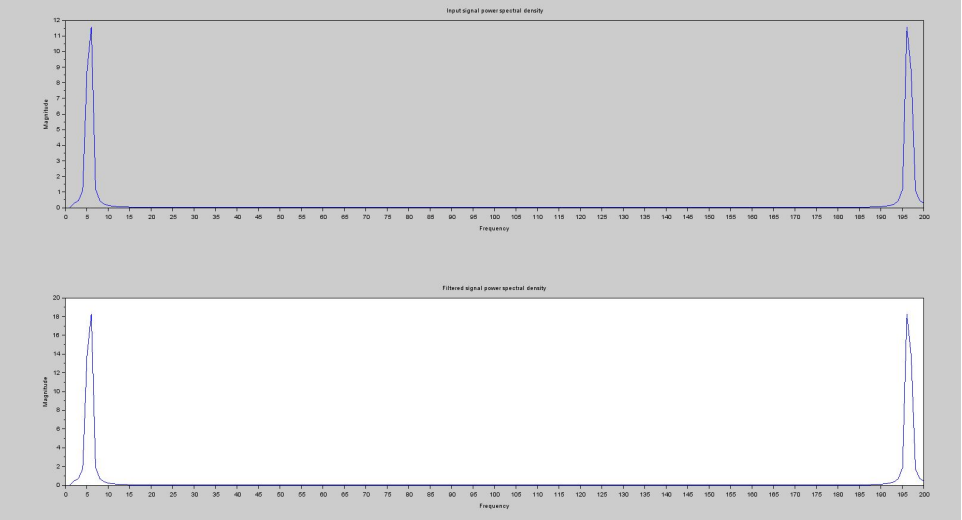

Figure 3.2: DSP Lab Migration

Experiment: 4

Design and test the working of a High Pass FIR filter using Hamming window by taking a high frequency signal as input

Scilab code Solution 4.4 DSP Lab Migration

```
1 // Version : Scilab 5.2.2
2 // Operating Syatem : Ubuntu 16.04 LTS
3
4 clc;          clear;       xdel(winsid());
5 fc=20000;     fs=44100;    M=63;
                                                    //
     Filter order
6 wc=2*fc/fs;
7 [wft,wfm,fr]=wfir('hp',M,[wc/2,0],'hm',[0,0]);
     //Fir Filter
8 figure(1)
```

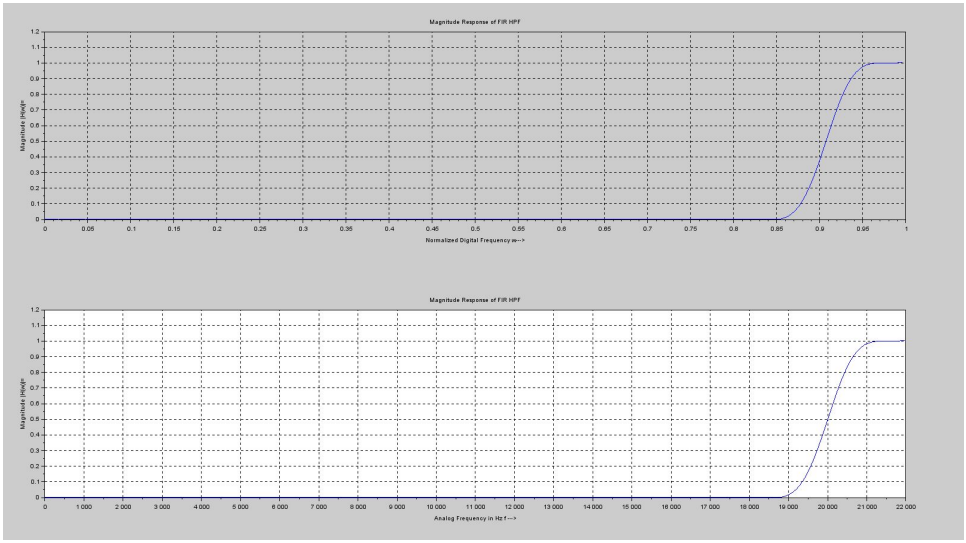

Figure 4.1: DSP Lab Migration

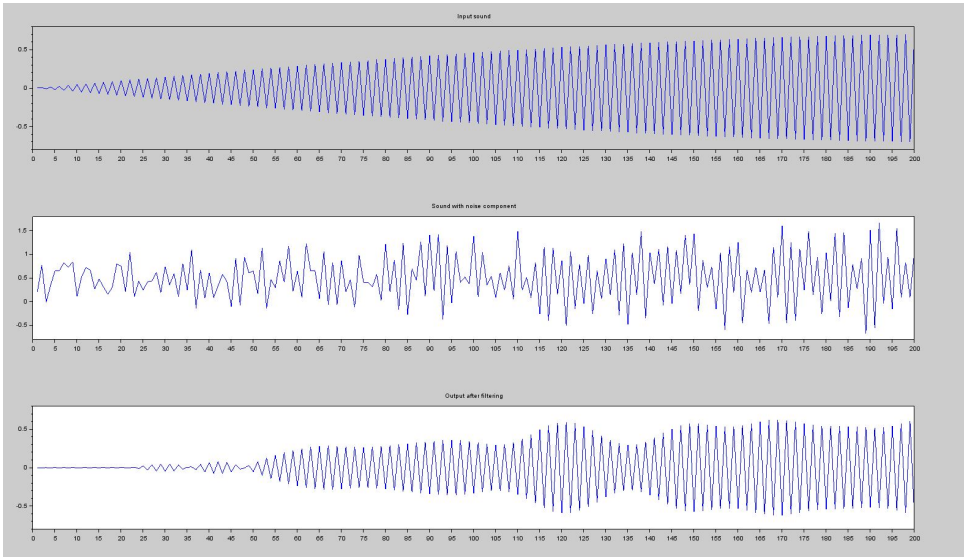

Figure 4.2: DSP Lab Migration

```
subplot(2,1,1);           plot(2*fr,wfm);
xlabel('Normalized Digital Frequency w---->')
ylabel('Magnitude |H(w)|=')
title('Magnitude Response of FIR HPF');      xgrid
   (1)
subplot(2,1,2);           plot(fr*fs,wfm);
xlabel('Analog Frequency in Hz f ---->')
ylabel('Magnitude |H(w)|=')
title('Magnitude Response of FIR HPF');      xgrid
   (1)

[d,Fs]=wavread("22000.wav")
playsnd(d,Fs)       //single tone high frequency
   sound wave
L = length(d);      a=1+nextpow2(L);        N=2*(2^a
   );
noise = rand(1,L);           data = d+noise;
   playsnd(data);
outhi = filter(wft,1,data);  playsnd(outhi);

figure(2)
subplot(3,1,1);      plot(d(1:200));
title('Input sound');
subplot(3,1,2);      plot(data(1:200));
title('Sound with noise component');
subplot(3,1,3);      plot(outhi(1:200));
title('Output after filtering');
```

Experiment: 5

Design and test the working of a Butterworth band pass filter by giving a time domain input signal

Scilab code Solution 5.5 DSP lab migration

```
1  // Version : Scilab 5.2.2
2  // Operating Syatem : Ubuntu 16.04 LTS
3  //Assume fp=1000 hz and fs=5000 hz
4  //Use the formula to convert the pass band ripple
       and To convert stop band attenuation in dB(A in
       dB)=-20*log10(Ap or As)
5  // assume kp=1.93 dB and ks=13.97 dB
6
7
8  clc;
9  clear;
10 xdel(winsid());
11 fp= input( " Enter the pass band edge (Hz) = ");
```

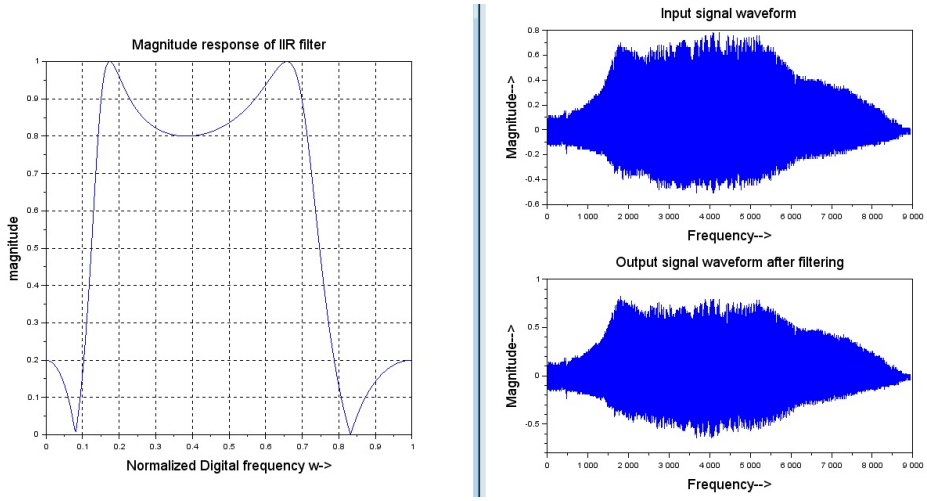

Figure 5.1: DSP lab migration

```
fs= input( "Enter the stop band edge (Hz) = ");
kp= 1.93                //assume the pass band ripple(
    dB)
ks= 13.97               //assume the stop band
    attenuation(dB)
Fsf=44000;              //sampling frequency
//Converting to digital frequency
Fp1=2*3.14*fp/Fsf;
Fs1=2*3.14*fs/Fsf;

// Digital filter specifications ( rad/samples )
N = log10(sqrt((10^(0.1*ks)-1)/(10^(0.1*kp)-1)))/
    log10(Fs1/Fp1);  //Order of the filter
N = ceil(N);             //rounded to nearest integer
disp(N," IIR Filter order N =");

oc = 0.5*((Fp1*Fsf)/((10^(0.1*kp)-1)^(1/(2*N))) + (
    Fs1*Fsf)/((10^(0.1*ks)-1)^(1/(2*N))) );  //Cut
    off Frequency
disp(oc, "Cutoff Frequency in rad/ seconds OC =")
[Hz]=iir(N,'bp','ellip',[Fp1/2,Fs1/2],[0.2,0.200])
```

```
      //the sum of last matrix [0.2,0.200] must be
    less than 1
[p,z,g]=iir(N,'bp','ellip',[Fp1/2,Fs1
    /2],[0.2,0.200])
[Hw,w]=frmag(Hz,256);
figure(1)
plot(2*w,abs(Hw));
xlabel('Normalized Digital frequency w->')
ylabel('magnitude');
title('Magnitude response of IIR filter')
xgrid(1)
[y,Fs]=wavread("H:\DSP SCILAB\Final\meow.wav")    //
    Reading input .wav signal file path of the .wav
    file must be changed
figure(2)
subplot(2,1,1)
plot(y)
title('Input signal waveform');
xlabel('Frequency-->');
ylabel('Magnitude-->');
playsnd(y)
outlo=filter(abs(z),abs(p),y);      //Passing
    accquired signal through desired filter
subplot(2,1,2)
plot(outlo)
title('Output signal waveform after filtering')
xlabel('Frequency-->');
ylabel('Magnitude-->');

N =length(y);                                        //
    Power spectral density of the Input signal
Y = fft(y,-1);
Pxx = (1/N)*(abs(Y).^2); //Peridogram Estimate
figure(3)
plot2d3('gnn',[1:N],Pxx)
title('Input signal power spectral density')
xlabel('Analog Frequency in Hz f ---->')
ylabel('Magnitude |H(w)|=')
```

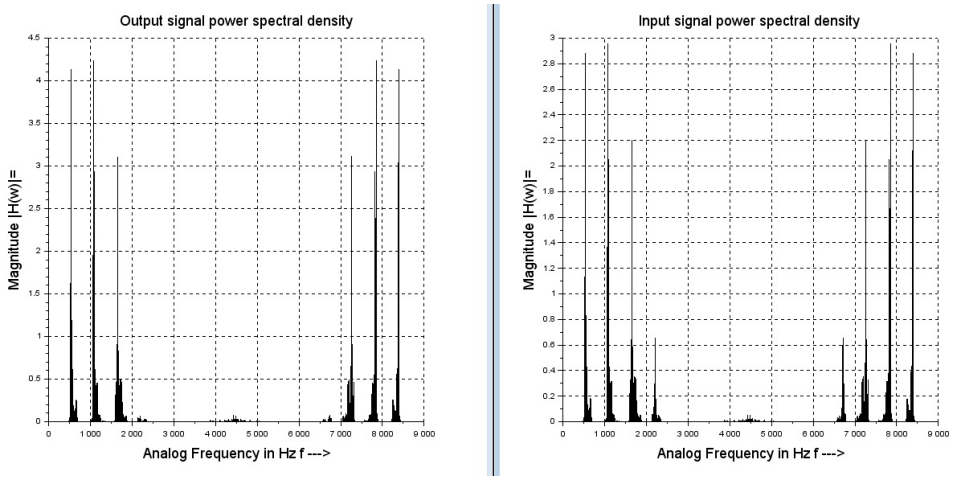

Figure 5.2: DSP lab migration

```
59 xgrid(1)
60 playsnd(outlo)
61
62 N=length(outlo)                                          ////
     Power spectral density of the Ouput signal
63 OL = fft(outlo,-1);
64 Fxx = (1/N)*(abs(OL).^2); //Peridogram Estimate
65 figure(4)
66 plot2d3('gnn',Fxx)
67 title('Output signal power spectral density')
68 xlabel('Analog Frequency in Hz f ---->')
69 ylabel('Magnitude |H(w)|=')
70 xgrid(1)
```

Experiment: 6

Design and test band pass, Butterworth filter for given specifications which includes Edge frequencies, Ripple and Attenuation

Scilab code Solution 6.6 DSP lab migration

```
1 // Version : Scilab 5.2.2
2 // Operating Syatem : Ubuntu 16.04 LTS
3
4 clc;
5 clear;
6 xdel(winsid());
7 fp= input( " Enter the pass band edge (Hz) = ");
8 fs= input( "Enter the stop band edge (Hz) = ");
9 kp= input( "Enter the pass band attenuation (dB) =")
```

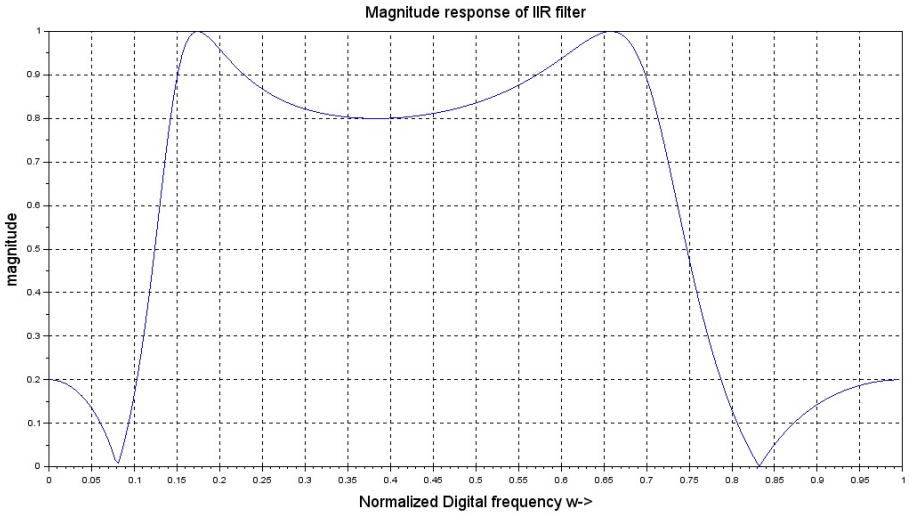

Figure 6.1: DSP lab migration

```
      ;
10 ks= input( "Enter the stop band attenuation (dB) = "
   ) ;
11 //kp= 1.93              //assume the pass band ripple
   (dB)
12 //ks= 13.97             //assume the stop band
   attenuation(dB)
13 Fsf=44000;          //sampling frequency
14 //Converting to digital frequency
15 Fp1=2*3.14*fp/Fsf;
16 Fs1=2*3.14*fs/Fsf;
17
18 // Digital filter specifications ( rad/samples )
19 N = log10(sqrt((10^(0.1*ks)-1)/(10^(0.1*kp)-1)))/
   log10(Fs1/Fp1);  //Order of the filter
20 N = ceil(N);           //rounded to nearest integer
21 disp(N," IIR Filter order N =");
22
23 oc = 0.5*((Fp1*Fsf)/((10^(0.1*kp)-1)^(1/(2*N))) + (
   Fs1*Fsf)/((10^(0.1*ks)-1)^(1/(2*N))) );  //Cut
```

```
 Enter the pass band edge (Hz) =
1000


Enter the stop band edge (Hz) =
5000

Enter the pass band attenuation (dB) =
2

Enter the stop band attenuation (dB) =
14


  IIR Filter order N =

   2.

 Cutoff Frequency in rad/ seconds OC =
   10675.066
```

Figure 6.2: DSP lab migration

```
    off Frequency
24 disp(oc, "Cutoff Frequency in rad/ seconds OC =")
25 [Hz]=iir(N,'bp','ellip',[Fp1/2,Fs1/2],[0.2,0.200])
       //the sum of last matrix [0.2,0.200] must be
    less than 1
26 [p,z,g]=iir(N,'bp','ellip',[Fp1/2,Fs1
    /2],[0.2,0.200])
27 [Hw,w]=frmag(Hz,256);
28 figure(1)
29 plot(2*w,abs(Hw));
30 xlabel('Normalized Digital frequency w->')
31 ylabel('magnitude');
32 title('Magnitude response of IIR filter')
33 xgrid(1)
```