

**ECE 2111 Signals and Systems
Spring 2005, UMD**

Experiment 9: Low Pass and High Pass Digital Filters

In this experiment you will learn to design Low Pass and High Pass Digital Filters using Matlab.

1.- Signal + High Frequency Noise

Using Matlab you will generate a 1Hz and a 10 Hz sinusoidal signals. You will add those two signals to generate a signal+highfrequency noise. To do this type the following commands.

```
% Duration 2 sec. Assume a sampling rate fs = 100 Hz, ie Ts= 0.01
t = 0:0.01:2;
y1= sin(2*pi*t);      % 1Hz signal
y10 = sin (20*pi*t); % 10Hz signal
% add y1 and y10 to generate the signal + high-frequency noise
yt = y1 + 0.3*y10
% plot the signals
subplot(3,1,1), plot (t,y1); title('signal')
subplot(3,1,2), plot (t,y10); title('high frequency signal')
subplot(3,1,3), plot (t,yt); title('signal + high-frequency noise')
```

2.- Low Pass Filter

2.1. Butterworth filters. Using the Matlab function: *butter*, design a 2nd, 4th, and 8th order low pass filters. For your design choose the cutoff frequency $f_c = 5$ Hz. Assume that the sampling rate for the input signals is $f_s = 100$ Hz.

Note the in the function *butter*, we have

$$[b,a] = \text{butter}(N, wc)$$

where N is the order of the filter, and wc is the cutoff frequency, normalized by πf_s .

This means that $wc=1$ implies $f_c=f_s$.

Then for $f_c = 5$ Hz, we have that $wc = 5(2\pi)/\pi f_s = 10/100 = 0.1$

For each of the designed filters compute and plot the step response of the filter, and its frequency response. Compare the plots and describe the differences. Comment about advantages and disadvantages (analyze the transition band in the frequency response, and the overshoot in the time domain)

2.2. Chebyshev type I . Using the Matlab function: *cheby1*, design a 2nd, 4th, and 8th order low pass filters. For your design choose the passband edge frequency $f_p = 5$ Hz.

Select the bandpass ripple $R_p = 0.5\text{db}$. Assume that the sampling rate for the input signal is $f_s = 100\text{Hz}$. w_p is also normalized by πf_s

[b,a] = cheby1(N, Rp, wp)

For each of the designed filters compute and plot the step response of the filter, and its frequency response. Compare the plots and describe the differences. Comment about advantages and disadvantages (analyze the transition band in the frequency response, and the overshoot in the time domain)

2.3. Chebyshev type II. Using the Matlab function: *cheby2*, design a 2nd, 4th, and 8th order low pass filters. For your design choose the stopband frequency $f_n = 5\text{ Hz}$. Assume that the sampling rate for the input signal is $f_s = 100\text{Hz}$. w_n is also normalized by πf_s . Select $R_s = 10\text{dB}$

[b,a] = cheby2(N, Rs, wn)

For each of the designed filters compute and plot the step response of the filter, and its frequency response. Compare the plots and describe the differences. Comment about advantages and disadvantages (analyze the transition band in the frequency response, and the overshoot in the time domain)

3.- High Pass Filter

3.1. Butterworth filters. Using the Matlab function: *butter*, design a 2nd, 4th, and 8th order high pass filters. For your design choose the cutoff frequency $f_c = 5\text{ Hz}$. Assume that the sampling rate for the input signals is $f_s = 100\text{Hz}$.

Note that in the function *butter*, we have

[b,a] = butter(N, wc, 'high')

where N is the order of the filter, and w_c is the cutoff frequency, normalized by πf_s .

This means that $w_c = 1$ implies $f_c = f_s$.

Then for $f_c = 5\text{Hz}$, we have that $w_c = 5(2\pi)/\pi f_s = 10/100 = 0.1$

For each of the designed filters compute and plot the step response of the filter, and its frequency response. Compare the plots and describe the differences. Comment about advantages and disadvantages (analyze the transition band in the frequency response, and the overshoot in the time domain)

3.2. Chebyshev type I. Using the Matlab function: *cheby1*, design a 2nd, 4th, and 8th order high pass filters. For your design choose the passband edge frequency $f_p = 5\text{ Hz}$. Select the bandpass ripple $R_p = 0.5\text{db}$. Assume that the sampling rate for the input signal is $f_s = 100\text{Hz}$. w_p is also normalized by πf_s

[b,a]=cheby1(N, Rp, wp, 'high')

For each of the designed filters compute and plot the step response of the filter, and its frequency response. Compare the plots and describe the differences. Comment about advantages and disadvantages (analyze the transition band in the frequency response, and the overshoot in the time domain)

3.3. Chebyshev type II . Using the Matlab function: *cheby2*, design a 2nd, 4th, and 8th order high pass filters. For your design choose the stopband frequency $f_c = 5$ Hz. Assume that the sampling rate for the input signal is $f_s = 100$ Hz. ω_n is also normalized by πf_s . Select $R_s = 10$ dB

[b,a]=cheby2(N, Rs, wn, 'high')

For each of the designed filters compute and plot the step response of the filter, and its frequency response. Compare the plots and describe the differences. Comment about advantages and disadvantages (analyze the transition band in the frequency response, and the overshoot in the time domain)

4.- Filtering a Signal

In this part you will apply the signal generated in part 1: signal+high frequency to your filters designed in parts 2 and 3. To do this you will use the Matlab function: *filter(b,a,yt)*, where *yt* is the signal+highfrequency noise generated in part 1.

4.1. Apply the signal *yt* to each of your low pass filters designed in part 2. Compare the result of each filter.

4.2. Apply the signal *yt* generated in part 1, to the high pass filters designed in part 3. Compare the result of each filter.

5.- Matlab Code Example

```
% Duration 2 sec. Assume a sampling rate fs = 100 Hz, Ts = 0.01
t = 0:0.01:2;
y1 = sin(2*pi*t);      % 1 Hz signal
y10 = sin(20*pi*t);   % 10 Hz signal

% Add y1 and y10 to generate the signal + high-frequency noise
yt = y1 + 0.3*y10;

%Plot the signals
figure(1)
subplot(3,1,1),plot(t,y1);title('Signal')
subplot(3,1,2),plot(t,y10);title('High frequency signal')
subplot(3,1,3),plot(t,yt);title('Signal + High-frequency noise')

% 2nd order
[num,den]=butter(2,0.1);
printsys(num,den,'z')
[y1,w1]=freqz(num,den,100);
y1=abs(y1);
yout1 = filter(num,den,yt);

% 4th order
[num2,den2]=butter(4,0.1);
printsys(num2,den2,'z')
[y2,w2]=freqz(num2,den2,100);
y2=abs(y2);
yout2 = filter(num2,den2,yt);

% 8th order
[num3,den3]=butter(8,0.1);
printsys(num3,den3,'z')
[y3,w3]=freqz(num3,den3,100);
y3=abs(y3);
yout3 = filter(num3,den3,yt);

% 16th order
[num4,den4]=butter(16,0.1);
printsys(num4,den4,'z')
[y4,w4]=freqz(num4,den4,100);
y4=abs(y4);
yout4 = filter(num4,den4,yt);

w1 = w1 * 100;
w2 = w2 * 100;
w3 = w3 * 100;
w4 = w4 * 100;
order = 2;
type = 'butterworth';

% Step response for the 2nd Order Filter
figure(2)
stepz(num,den)
title(sprintf('%g POLE %s LPF ',order,type))
order = 4;
```

```

% Step response for the 4th Order Filter
figure(3)
stepz(num2,den2)
title(sprintf('%g POLE %s LPF ',order,type))
order = 8;

% Step response for the 8th Order Filter
figure(4)
stepz(num3,den3)
title(sprintf('%g POLE %s LPF ',order,type))
order = 16;

% Step response for the 16th Order Filter
figure(5)
stepz(num4,den4)
title(sprintf('%g POLE %s LPF ',order,type))

% Frequency Response for all the filters
figure(6)
plot(w1,y1,'b',w2,y2,'c',w3,y3,'g',w4,y4,'m')

% Output of the filters after inserting the signal
figure(7)
subplot(4,1,1),plot(yout1);title('Output of the signal after the
2nd order filter')
subplot(4,1,2),plot(yout2);title('Output of the signal after the
4th order filter')
subplot(4,1,3),plot(yout3);title('Output of the signal after the
8th order filter')
subplot(4,1,4),plot(yout4);title('Output of the signal after the
16th order filter')

```