Chittagong University of Engineering & Technology

EEE-496

Digital Signal Processing Sessional

Design Low Pass, High Pass and Band Pass FIR filter.

Submitted by:
MD. Sayedul
ID: **1702079**
Section: **B**

Submitted to:
Dr. Muhammad Ahsan Ullah
Professor
Dept. of EEE

June 13, 2023

Contents

1 Objectives

The objectives of the experiment are:

1. To design Hamming window, Hanning window, Blackman, Berlett window using MATLAB.
2. To design low pass, high pass, band pass kernal.
3. To design low pass, high pass, band pass FIR filter.

2 Different type of window

2.1 Hamming window

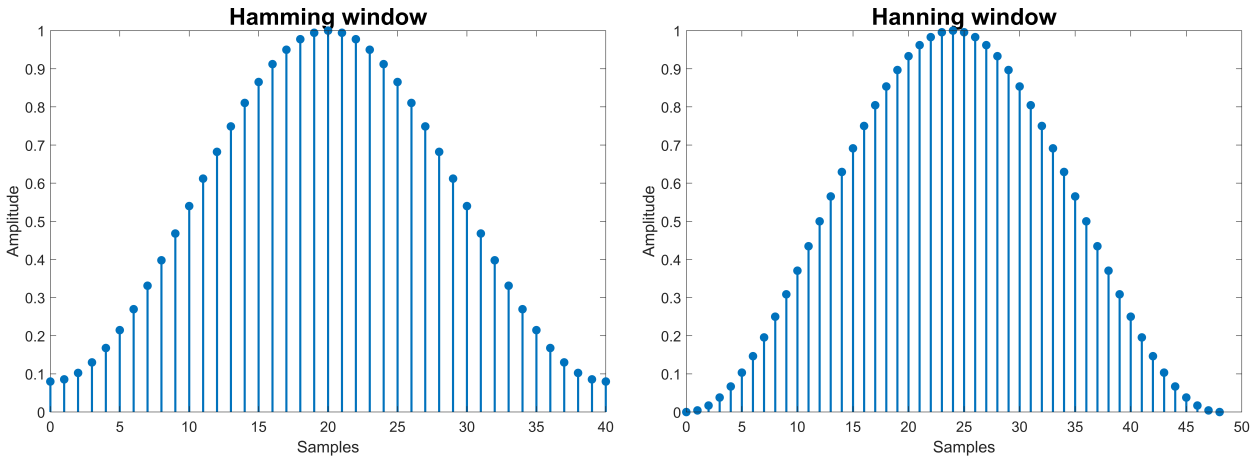
```
clc;
clear;
close all;
wp=0.4*pi;
ws=0.6*pi;
TranBand=ws-wp;
M=8*pi/TranBand;
n=0:1:M;
w=0.54-0.46*cos(2*n*pi/M);
stem(n, w, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("Hamming window", "FontSize", 16)
```

2.2 Hnning window

```
clc;
clear;
close all;
wp=0.5*pi;
ws=0.33*pi;
TranBand=wp-ws;
M=ceil(8*pi/TranBand);
```

```
n=0:1:M;
w=0.5-0.5*cos(2*n*pi/M);
stem(n, w, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("Hanning window", "FontSize", 16)
```

2.3 Figure



(a) (b)

Figure 1: (a) Hamming window. (b) Hanning window.

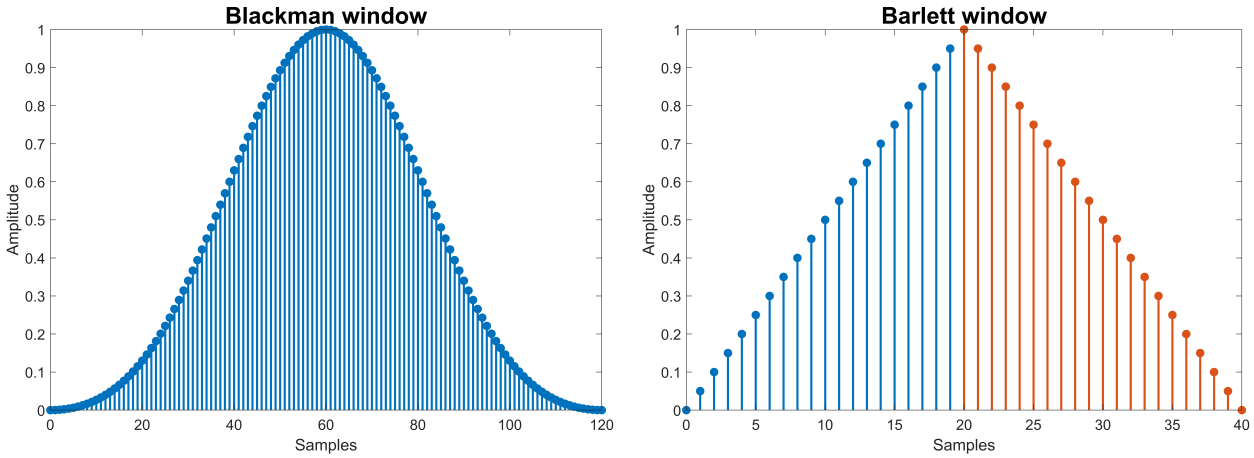
2.4 Blackman window

```
clc;
clear;
close all;
M=120;
n=0:1:M;
w=0.42-0.5*cos(2*n*pi/M)+0.08*cos(4*n*pi/M);
stem(n, w, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("Blackman window", "FontSize", 16)
```

2.5 Barlett window

```
clc;
clear;
close all;
M=40
n=0:1:M;
w1=2*n(1:M/2)/M;
w2=2-(2*n(M/2+1:end)/M);
stem(n(1:M/2), w1, "filled", "linewidth", 2)
hold on
stem(n(M/2+1:end), w2, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("Barlett window", "FontSize", 16)
```

2.6 Figure



(a) (b)

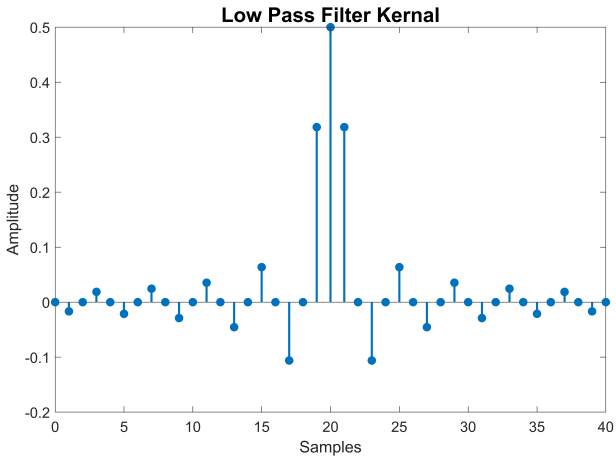
Figure 2: (a) Blackman window. (b) Barlett window.

3 Different type of kernal

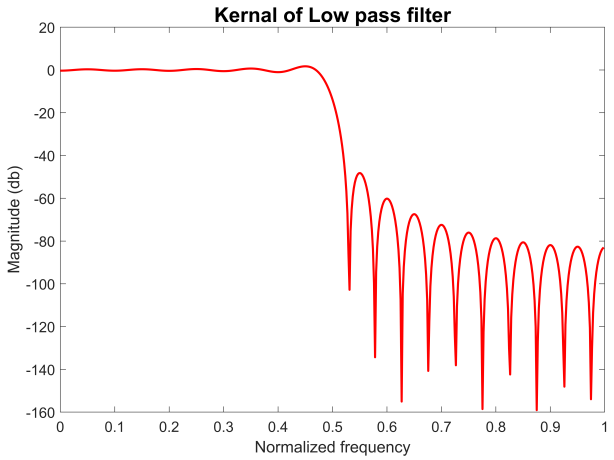
3.1 Low pass filter kernal

```
clc;
clear;
```

```
close all;
wp=0.4*pi;
ws=0.6*pi;
wc=(wp+ws)/2;
TranBand=ws-wp;
M=8*pi/TranBand;
n=0:1:M;
h_d_lpf=(wc/pi)*sinc(wc*(n-M/2)/pi);
figure
stem(n, h_d_lpf, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("Low Pass Filter Kernal", "FontSize", 14)
figure
%freqz(h_d_lpf)
[h, w]=freqz(h_d_lpf);
plot(w/pi, 20*log(abs(h)), "color", "r", "linewidth
  ", 2)
xlabel("Normalized frequency")
ylabel("Magnitude (db)")
title("Kernal of Low pass filter", "FontSize", 14)
```



(a)

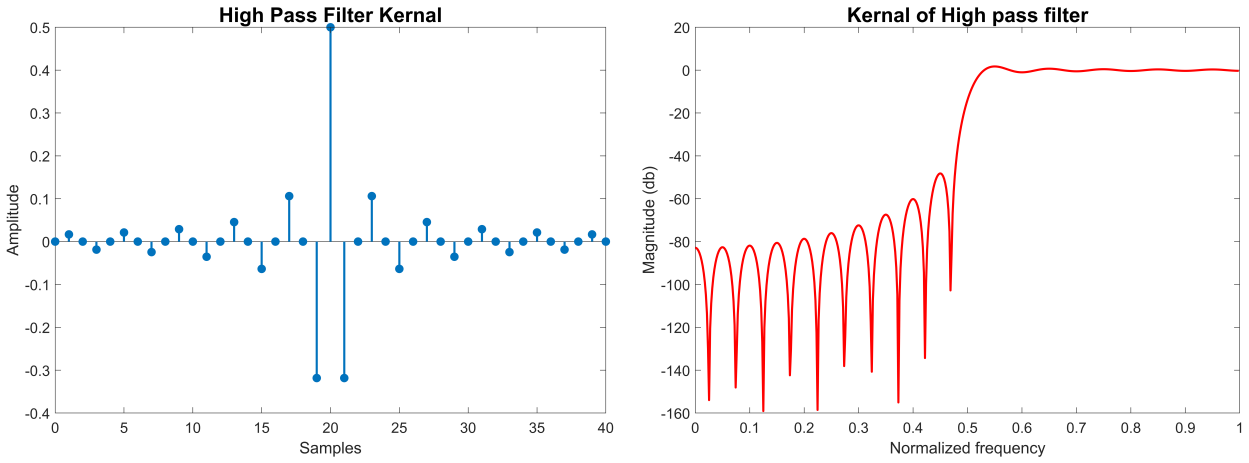


(b)

Figure 3: (a) Kernal of low pass filter in time domain. (b) Kernal of low pass filter in frequency domain.

3.2 High pass filter kernal

```
clc;
clear;
close all;
wp=0.4*pi;
ws=0.6*pi;
wc=(wp+ws)/2;
TranBand=ws-wp;
M=8*pi/TranBand;
n=0:1:M;
h_d_hpf=sinc(n-M/2)-(wc/pi)*sinc(wc*(n-M/2)/pi);
figure
stem(n, h_d_hpf, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("High Pass Filter Kernal", "FontSize", 14)
figure
%freqz(h_d_lpf)
[h, w]=freqz(h_d_hpf);
plot(w/pi, 20*log(abs(h)), "color", "r", "linewidth
  ", 2)
xlabel("Normalized frequency")
ylabel("Magnitude (db)")
title("Kernal of High pass filter", "FontSize", 14)
```



(a) (b)

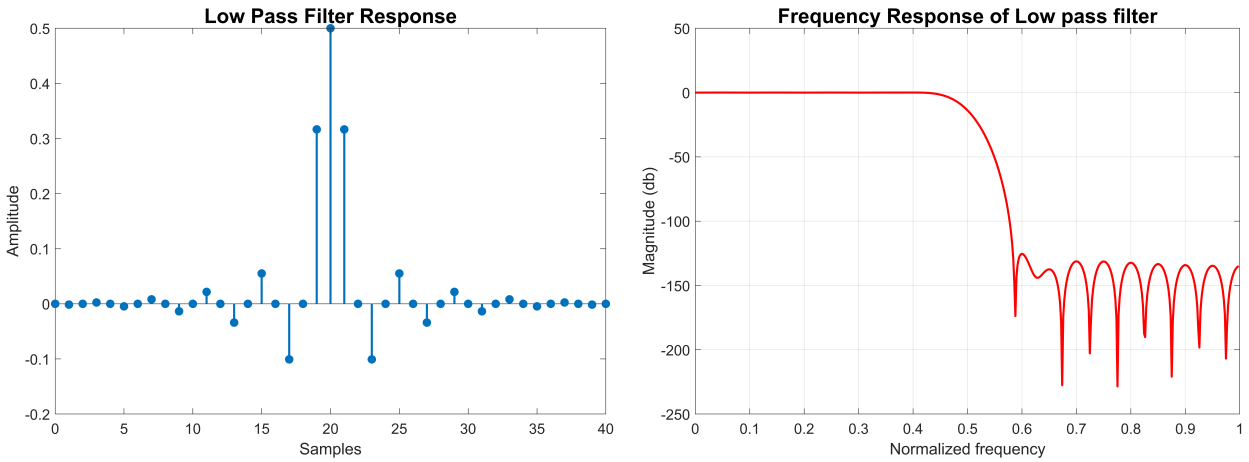
Figure 4: (a) Kernal of high pass filter in time domain. (b) Kernal of high pass filter in frequency domain.

4 Different type of filter response

4.1 Low pass filter response

```
clc;
clear;
close all;
wp=0.4*pi;
ws=0.6*pi;
wc=(wp+ws)/2;
TranBand=ws-wp;
M=8*pi/TranBand;
n=0:1:M;
w=0.54-0.46*cos(2*n*pi/M);
h_d_lpf=(wc/pi)*sinc(wc*(n-M/2)/pi);
h_n=w.*h_d_lpf;
figure
stem(n, h_n, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("Low Pass Filter Response", "FontSize", 14)
figure
```

```
%freqz(h_d_lpf)
[h, w]=freqz(h_n);
plot(w/pi, 20*log(abs(h)), "color", "r", "linewidth
   ", 2)
grid on;
xlabel("Normalized frequency")
ylabel("Magnitude (db)")
title("Frequency Response of Low pass filter", "
   FontSize", 14)
```



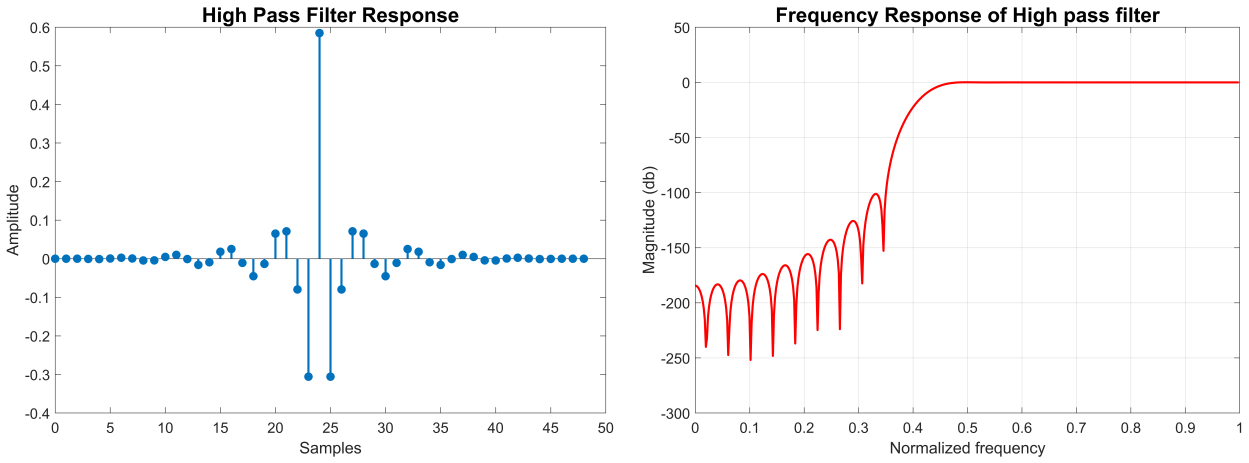
(a) (b)

Figure 5: (a) Response of low pass filter in time domain. (b) Response of low pass filter in frequency domain.

4.2 High pass filter response

```
clc;
clear;
close all;
wp=0.5*pi;
ws=0.33*pi;
wc=(wp+ws)/2;
TranBand=wp-ws;
M=ceil(8*pi/TranBand);
n=0:1:M;
```

```
w=0.5-0.5*cos(2*n*pi/M);
h_d_hpf=sinc(n-M/2)-(wc/pi)*sinc(wc*(n-M/2)/pi);
h_n=w.*h_d_hpf;
figure
stem(n, h_n, "filled", "linewidth", 2)
xlabel("Samples")
ylabel("Amplitude")
title("High Pass Filter Response", "FontSize", 14)
figure
%freqz(h_d_lpf)
[h, w]=freqz(h_n);
plot(w/pi, 20*log(abs(h)), "color", "r", "linewidth
  ", 2)
grid on;
xlabel("Normalized frequency")
ylabel("Magnitude (db)")
title("Frequency Response of High pass filter", "
  FontSize", 14)
```



(a) (b)

Figure 6: (a) Response of high pass filter in time domain. (b) Response of high pass filter in frequency domain.