EENG 479 : Digital Signal Processing (DSP)

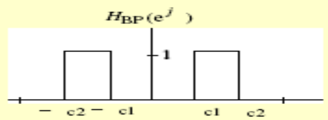
Lecture #14: FIR Design Using Matlab

Prof. Mohab A. Mangoud
Professor of Wireless Communications (Networks, IoT and AI)
University of Bahrain, College of Engineering
Department of Electrical and Electronics Engineering
P.O.Box 32038- Kingdom of Bahrain
mmangoud@uob.edu.bh
http://mangoud.com

FIR-BPF Design Example

[Ex 1] Plot the magnitude response of a linear phase FIR highpass filter using a rectangular Window truncation of ideal impulse response $h_{BP}[n]$ of the ideal BP filter to length N=2M+1 <u>for two different values of M</u>. show that the truncated filter exhibits oscillatory behavior on both sides of cutoff frequencies at 0.3 π and 0.7 π rad/sec.

- Ideal bandpass filter –

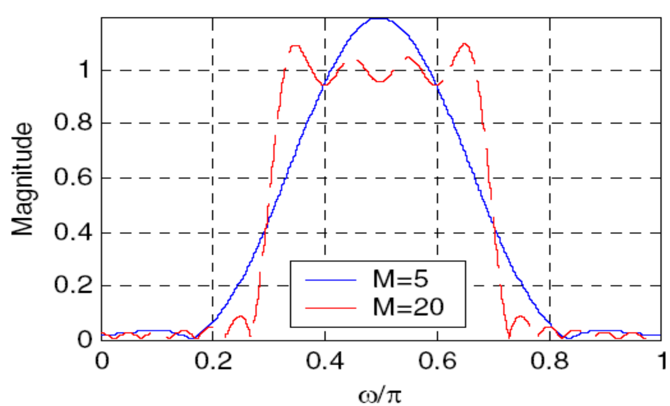


$$h_{BP}[n] = \begin{cases} \frac{\sin(\omega_{c2}n)}{\pi n} - \frac{\sin(\omega_{c1}n)}{\pi n}, & n \neq 0 \\ \frac{\omega_{c2}}{\pi} - \frac{\omega_{c1}}{\pi}, & n = 0 \end{cases}$$

The impulse response coefficients of the truncated FIR bandpass filter with cutoff frequencies at 0.7π and 0.3π can be generated using the following MATLAB statements:

```
n = -M:M;
num = 0.7*sinc(0.7*n) - 0.3*sinc(0.3*n);
```

The magnitude responses of the truncated FIR bandpass filter for two values of *M* are shown below:



FIR-LPF Design Example

[Ex 2] Using the windowed FS approach, design a linear phase FIR LPF with the following specifications:

Passband edge at 4 KHz,
stopband edge at 6 KHz,
maximum passband attenuation of 0.2 dB,
minimum stopband attenuation of 42 dB,
and sampling frequency of 18 KHz.

Use each of the following windows for the design: Hamming, Hann, and Blackman. Show the impulse response coefficients and plot the gain response of the designed filter for each case. Comment on your results. Do not use the M file fir1.

Passband edge at 2 khz,
stopband edge at 4 khz,
maximum passband attenuation of 0.2 dB,
minimum stopband attenuation of 42 dB,
and sampling frequency of 18 khz.

```
wp = 2*(2*pi)/18;
ws = 4*(2*pi)/18;
wc = (wp + ws)/2;    6pi/18
dw = ws - wp;
```

- Filter Design Steps -
  - (1) Set
    $$\omega_c = (\omega_p + \omega_s)/2$$
  - (2) Choose window based on specified α_s
  - (3) Estimate M using
    $$\Delta\omega \approx \frac{c}{M}$$

See table

Table 10.2: Properties of some fixed window functions.[4]

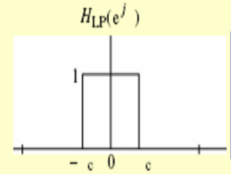
Type of Window	Main Lobe Width Δ_{ML}	Relative Sidelobe Level $A_{s\ell}$	Minimum Stopband Attenuation	Transition Bandwidth $\Delta\omega$
Rectangular	$4\pi/(2M+1)$	13.3 dB	20.9 dB	$0.92\pi/M$
Barlett	$4\pi/(M+1)$	26.5 dB	See text	See text
Hann	$8\pi/(2M+1)$	31.5 dB	43.9 dB	$3.11\pi/M$
Hamming	$8\pi/(2M+1)$	42.7 dB	54.5 dB	$3.32\pi/M$
Blackman	$12\pi/(2M+1)$	58.1 dB	75.3 dB	$5.56\pi/M$

sinc(x) = sin(pi*x)/(pi*x)

```
% Hamming
M = ceil(3.32*pi/dw);N = 2*M+1;n = -M:M;
num = (6/18)*sinc(6*n/18);
wh = hamming(N)';b = num.*wh;

figure(1);
k=0:2*M:stem(k,b);
title('Impulse Response Coefficients');
xlabel('Time index n'); ylabel('Amplitude');

figure(2);
[h, w] = freqz(b,1,512);
plot(w/pi, 20*log10(abs(h))); grid;
xlabel('\omega/\pi'); ylabel('Gain, in dB');
title('Lowpass filter designed using Hamming window');
axis([0 1 -80 10]);
```

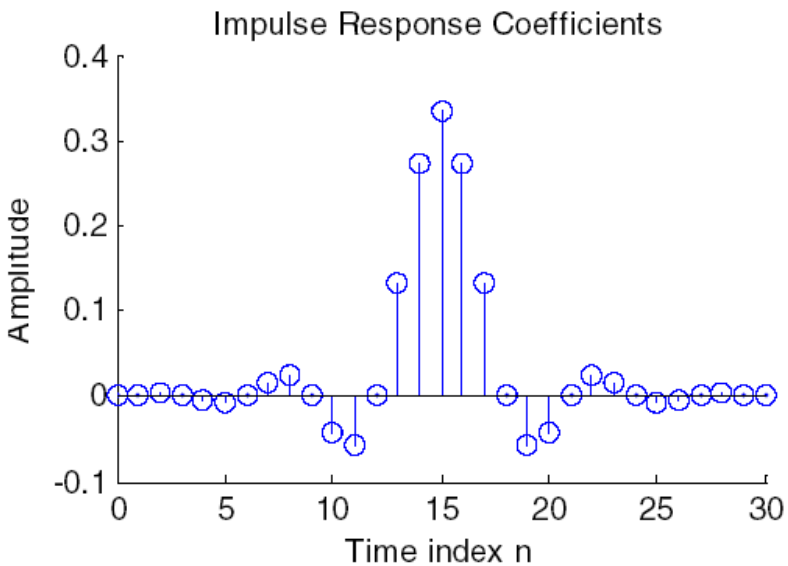
$$2f_c \frac{\sin(n\omega_c)}{n\omega_c}$$

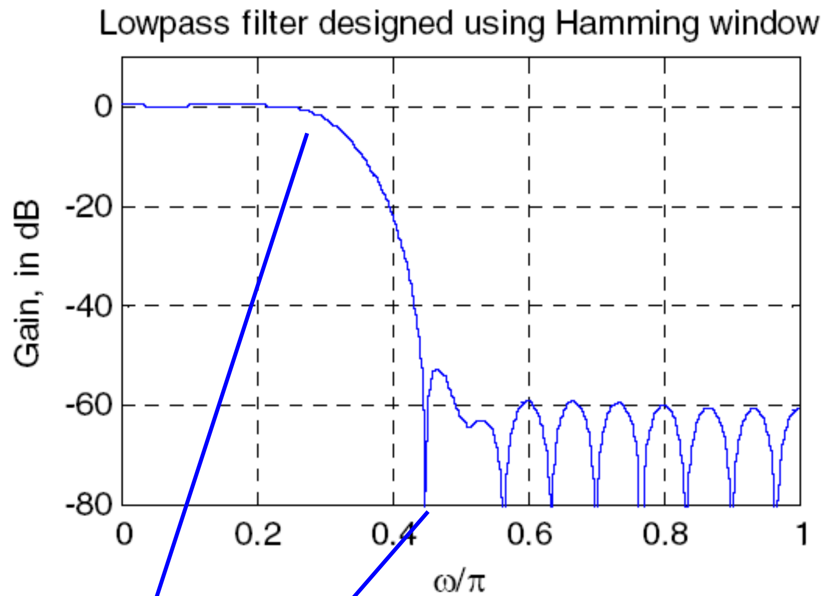
- Ideal lowpass filter -

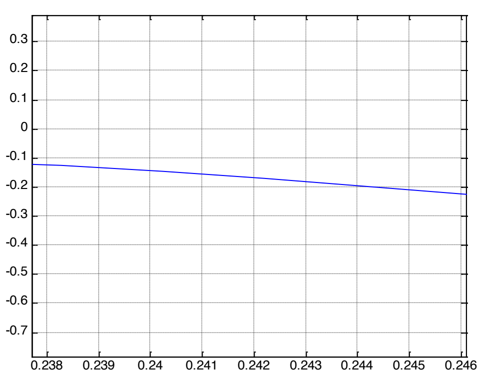


$$h_{LP}[n] = \frac{\sin \omega_c n}{\pi n}, \quad -\infty \le n \le \infty$$

Lowpass filter design using Hamming window: $N = 31$



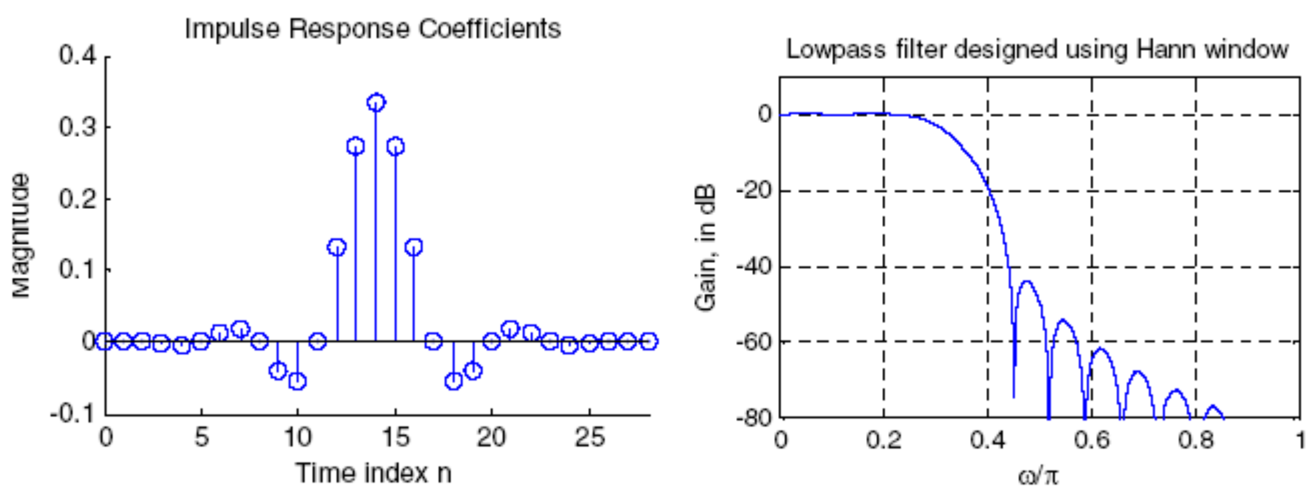





```
>> wp/pi

ans =

    0.2222

>> ws/pi

ans =

    0.4444
```

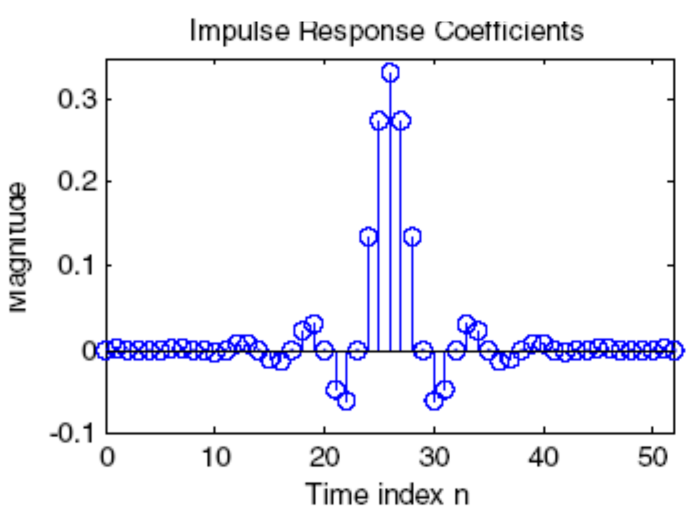
```
% Hann
M = ceil(3.11*pi/dw);N = 2*M+1;n = -M:M;
num = (6/18)*sinc(6*n/18);
wh = hann(N)';b = num.*wh;

figure(3);
k=0:2*M:stem(k,b);
title('Impulse Response Coefficients');
xlabel('Time index n'); ylabel('Amplitude');

figure(4);
[h, w] = freqz(b,1,512);
plot(w/pi, 20*log10(abs(h)));grid;
xlabel('\omega/\pi');ylabel('Gain, in dB');

title('Lowpass filter designed using Hann window');
axis([0 1 -80 10]);
```

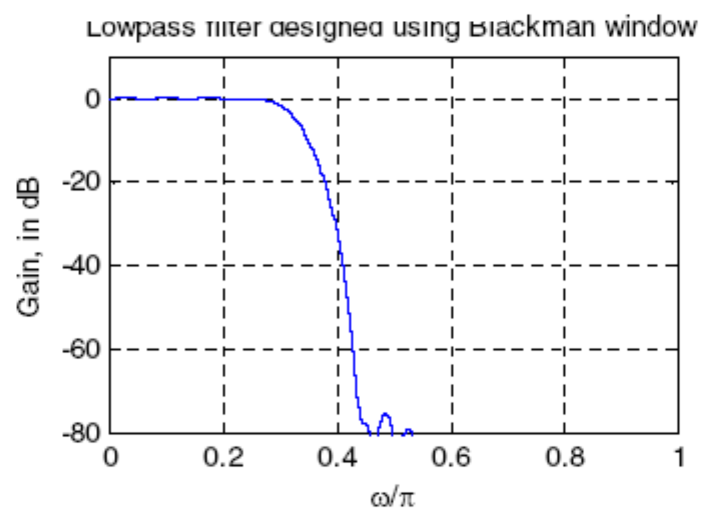
Lowpass filter design using Hann window: $N = 29$



```
% Blackman
M = ceil(5.56*pi/dw);N = 2*M+1;n = -M:M;
num = (6/18)*sinc(6*n/18);
wh = blackman(N)';b = num.*wh;

figure(5);
k=0:2*M:stem(k,b);
title('Impulse Response Coefficients');
xlabel('Time index n'); ylabel('Amplitude');
figure(6);
[h, w] = freqz(b,1,512);
plot(w/pi, 20*log10(abs(h)));grid;
xlabel('\omega/\pi');ylabel('Gain, in dB');
title('Lowpass filter designed using Blackman window');
axis([0 1 -80 10]);
```

Lowpass filter design using Blackman window: $N = 53$





Comments: The Hann window method results in using the lowest filter order. All filters meet the requirements of the specifications.

Adjustable Window Functions

- **Kaiser Window** -

The most commonly used

$$w[n] = \frac{I_0\{\beta\sqrt{1-(n/M)^2}\}}{I_0(\beta)}, \quad -M \le n \le M$$

where β is an adjustable parameter and $I_0(u)$ is the modified zeroth-order Bessel function of the first kind:

$$I_0(u) = 1 + \sum_{r=1}^{\infty}\left[\frac{(u/2)^r}{r!}\right]^2$$

N=2M

- Note $I_0(u) > 0$ for $u > 0$
- In practice $I_0(u) \cong 1 + \sum_{r=1}^{20}\left[\frac{(u/2)^r}{r!}\right]^2$

17

- β controls the minimum stopband attenuation of the windowed filter response
- β is estimated using

$$\beta = \begin{cases} 0.1102(\alpha_s - 8.7), & \text{for } \alpha_s > 50 \\ 0.5842(\alpha_s - 21)^{0.4} + 0.07886(\alpha_s - 21), & \text{for } 21 \le \alpha_s \le 50 \\ 0, & \text{for } \alpha_s < 21 \end{cases}$$

- Filter order is estimated using

$$N = \frac{\alpha_s - 8}{2.285(\Delta\omega)}$$

where $\Delta\omega$ is the normalized transition bandwidth

Note: it provides no control over passband ripples

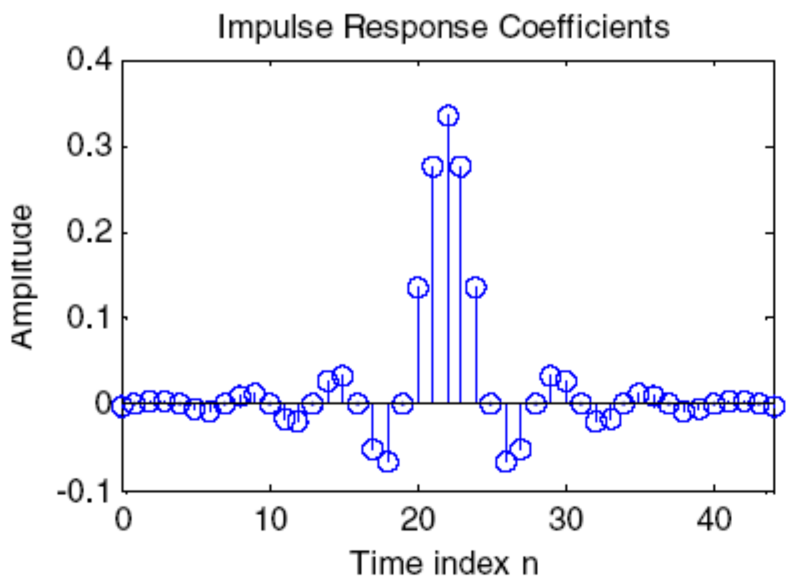
[Ex 3] Repeat Exercise using the Kaiser window. Do not use the M file fir1.

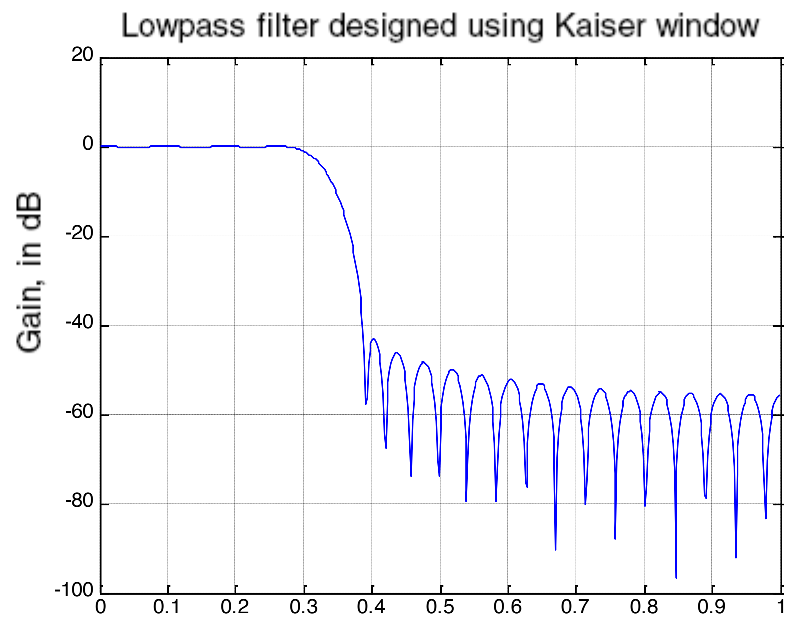
$\alpha_s = 42$, $\beta = 0.5842(42-21)^{0.4} + 0.07886(42-21) = 3.631$ using Eq. (10.41).

$$N = \frac{42-8}{2.285\left(\frac{4\pi}{18}\right)}$$ using Eq. (10.42).

$N =$ 21.3 and we choose 22 since N must be even. $M =$ 11

```
beta = 3.631;N = 22 n = -N/2:N/2;
num = (6/18)*sinc(6*n/18);
wh = kaiser(N+1,beta)';b = num.*wh;
figure(1);
stem(b);
title('Impulse Response Coefficients');
xlabel('Time index n');ylabel('Amplitude')
figure(2);
[h, w] = freqz(b,1,512);
plot(w/pi, 20*log10(abs(h)));grid;
xlabel('\omega/\pi');ylabel('Gain, in dB');
title('Lowpass filter designed using Kaiser window');
axis([0 1 -80 10]);
```


Impulse Response Coefficients
0.4
0.3
0.2
0.1
0
-0.1
Amplitude
0
10
20
30
40
Time index n


Lowpass filter designed using Kaiser window
20
0
-20
-40
-60
-80
-100
Gain, in dB
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

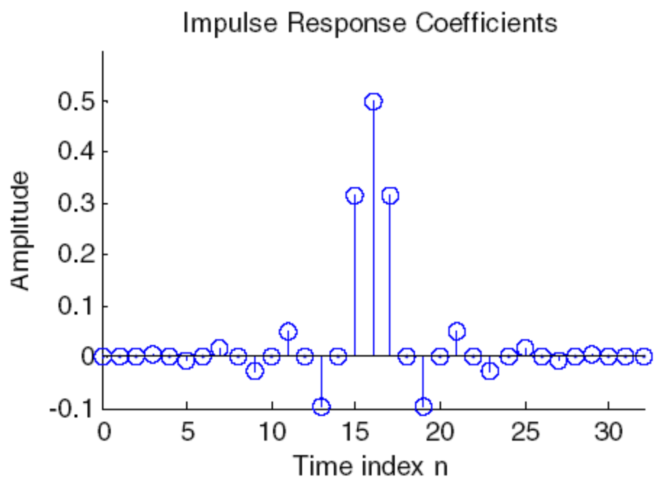
Ex4 Using the windowed Fourier series approach, design a linear-phase FIR lowpass filter of lowest order with the following specifications: passband edge at 0.4π, stopband edge at 0.6π, and minimum stopband attenuation of 42 dB. Which window function is appropriate for this design? Show the impulse response coefficients, and plot the gain response of the designed filter. Comment on your results. Do not use the M-file `fir1`.

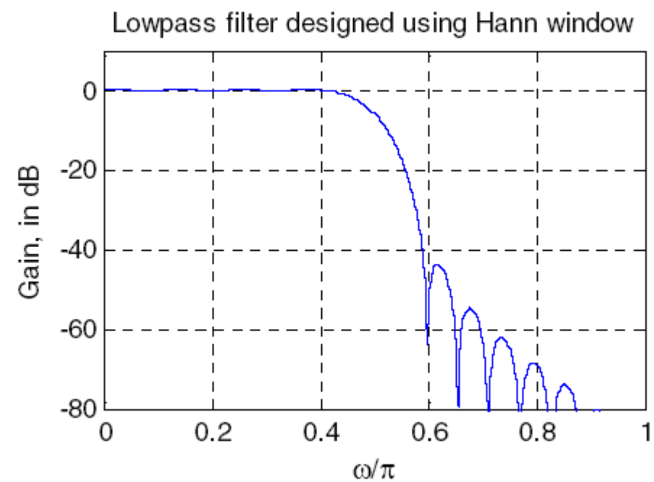
$$\omega_p = 0.4\pi\,, \omega_s = 0.6\pi\,, \alpha_s = 42\,\text{dB}, \omega_c = 0.5\pi\,, \Delta\omega = 0.2\pi$$

We will use the Hann window since it meets the requirements and has the lowest order from Table 10.2.

$$M = \frac{3.11\pi}{\Delta\omega} = 15.55 \rightarrow 16 \Rightarrow N = 32.$$

```
n = -16:16;
lp = 0.5*sinc(0.5*n);wh = hanning(33);
b = lp.*wh';
figure(1);
k=0:2*n;stem(k,b);
title('Impulse Response Coefficients');
xlabel('Time index n');ylabel('Amplitude');
figure(2);
[h, w] = freqz(b,1,512);
plot(w/pi, 20*log10(abs(h)));grid;
xlabel('\omega/\pi');ylabel('Gain, in dB');
title('Lowpass filter designed using Hann window');
axis([0 1 -80 10]);
```





Ex6 Using the M-file `fir1`, design a linear-phase FIR bandpass filter with the following specifications: stopband edges at 0.55π and 0.75π, passband edges at 0.65π and 0.85π, maximum passband attenuation of 0.2 dB, and minimum stopband attenuation of 42 dB. Use each of the following windows for the design: Hamming, Hann, Blackman, and Kaiser. Show the impulse response coefficients, and plot the gain response of the designed filters for each case. Comment on your results.

$\omega_{p1} = 0.65\pi$, $\omega_{p2} = 0.85\pi$, $\omega_{s1} = 0.55\pi$, $\omega_{s2} = 0.75\pi$, $\alpha_p = 0.2$ dB, $\alpha_s = 42$ dB

$\Delta\omega_1 = \omega_{p1} - \omega_{s1} = 0.1\pi$, $\Delta\omega_2 = \omega_{p2} - \omega_{s2} = 0.1\pi = \Delta\omega$

(a) Hamming window: $M = \dfrac{3.32\pi}{0.1\pi} = 33.2 \rightarrow 34 \therefore N = 2M = 68$

(b) Hann: $M = \dfrac{3.11\pi}{0.1\pi} = 31.1 \rightarrow 32 \therefore N = 2M = 64$

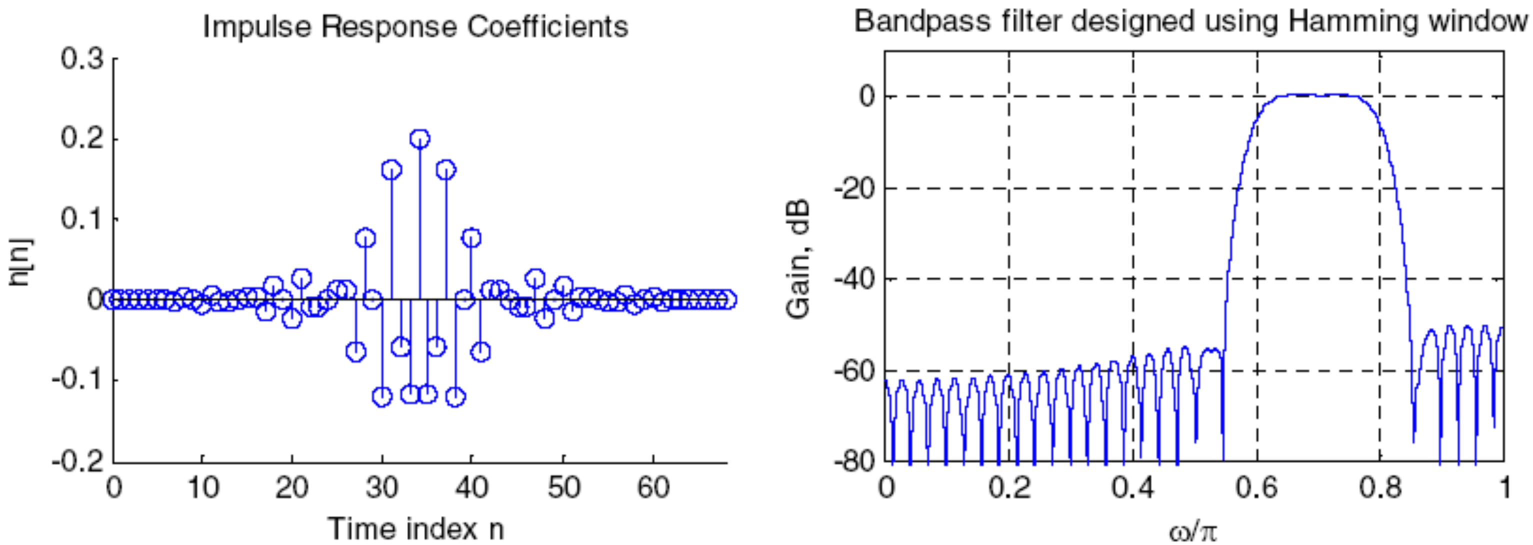
(c) Blackman: $M = \dfrac{5.56\pi}{0.1\pi} = 55.6 \rightarrow 56 \therefore N = 2M = 112$

(d) Kaiser: $\delta_s = 10^{-\alpha_s/20} = 0.00794$, $\delta_p = 10^{-\alpha_p/20} = 0.97724$

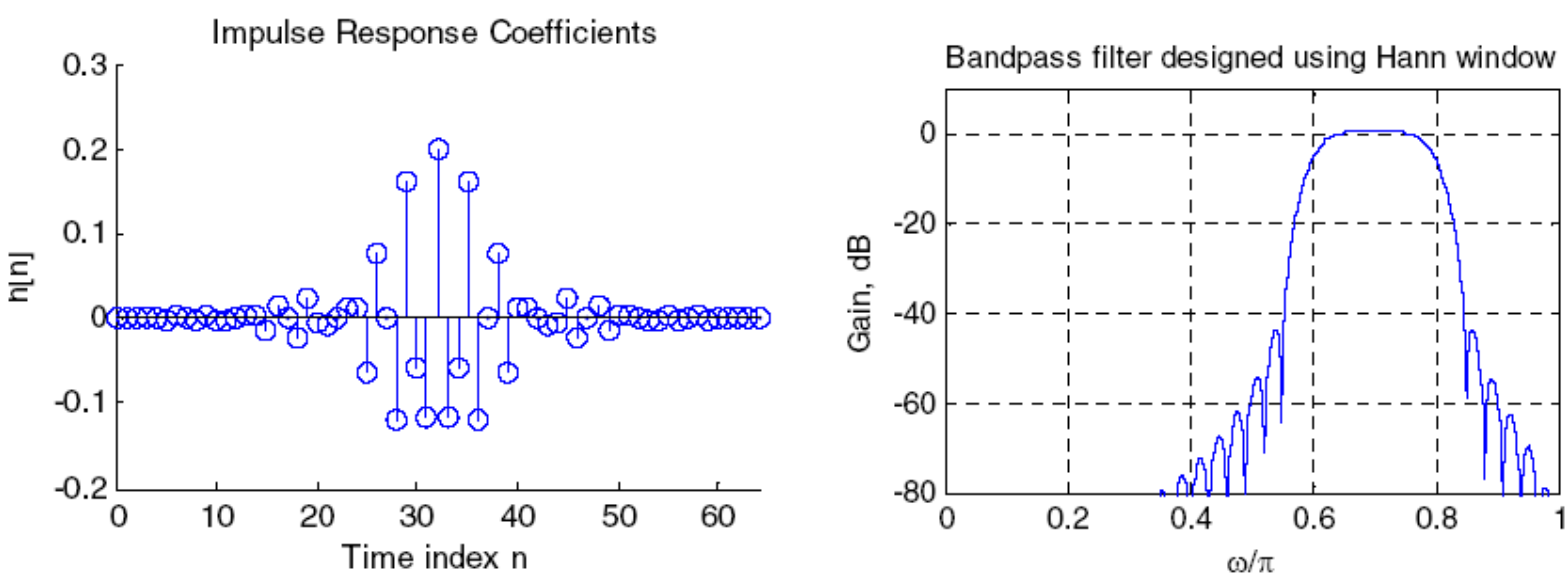
```
% Hamming
N = 68;
b = fir1(N, [0.6 0.8]);
[H, w] = freqz(b,1,512);
figure(1);
stem(b);
title('Impulse Response Coefficients');
xlabel('Time index n');ylabel('h[n]');
figure(2);
plot(w/pi, 20*log10(abs(H)));grid;
xlabel('\omega/\pi');ylabel('Gain, dB');
title('Bandpass filter designed using Hamming window');
axis([0 1 -80 10]);

% Hann
N = 64;
b = fir1(N, [0.6 0.8], hanning(N+1));
[H, w] = freqz(b,1,512);
figure(3);
stem(b);
title('Impulse Response Coefficients');
xlabel('Time index n');ylabel('h[n]');
figure(4);
plot(w/pi, 20*log10(abs(H)));grid;
xlabel('\omega/\pi');ylabel('Gain, dB');
title('Bandpass filter designed using Hann window');
axis([0 1 -80 10]);
```

```
% Blackman
N = 112;
b = fir1(N, [0.6 0.8], blackman(N+1));
[H, w] = freqz(b,1,512);
figure(5);
stem(b);
title('Impulse Response Coefficients');
xlabel('Time index n');ylabel('h[n]');
figure(6);
plot(w/pi, 20*log10(abs(H)));grid;
xlabel('\omega/\pi');ylabel('Gain, dB');
title('Bandpass filter designed using Blackman window');
axis([0 1 -80 10]);

% Kaiser
[N, Wn, beta, type] = kaiserord([0.6 0.8], [1 0], [0.97724
0.00794]);

 b = fir1(2*N, [0.6 0.8], kaiser(2*N+1, beta));
 [H, w] = freqz(b,1,512);
 figure(7);
 stem(b);
 title('Impulse Response Coefficients');
 xlabel('Time index n');ylabel('h[n]');

 figure(8);
 plot(w/pi, 20*log10(abs(H)));grid;
 xlabel('\omega/\pi');ylabel('Gain, dB');
 title('Bandpass filter designed using Kaiser window');
 axis([0 1 -80 10]);
```

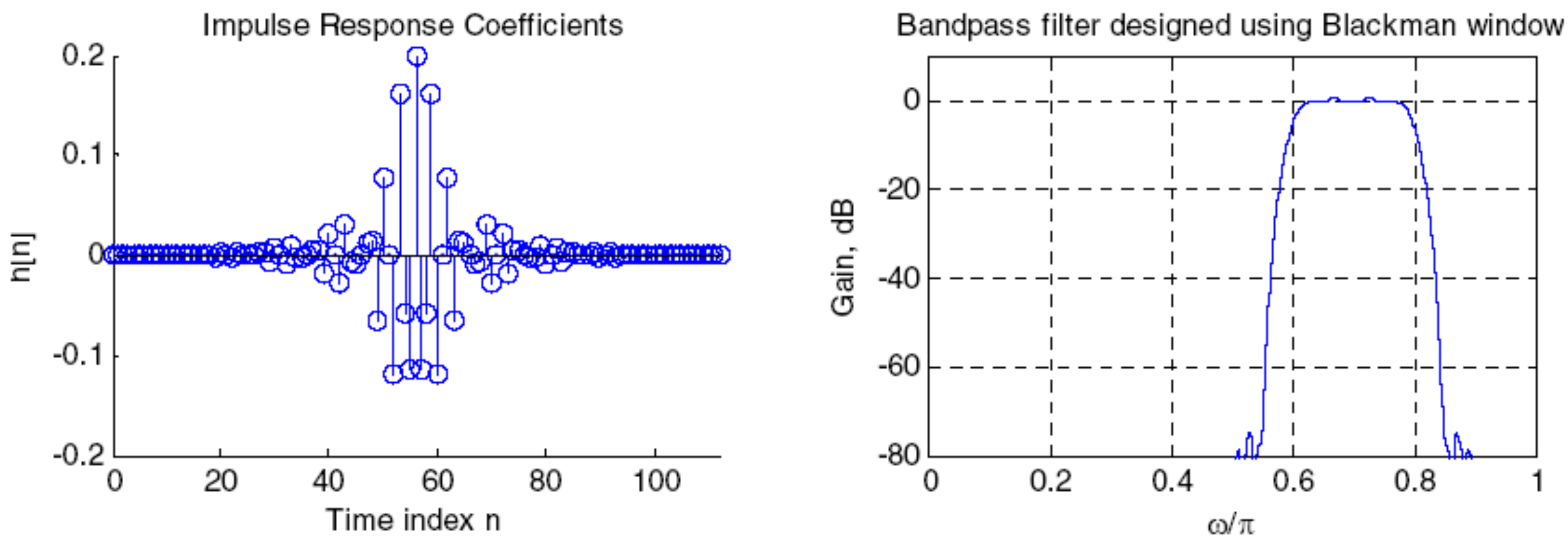
(a) Hamming window using `fir1`



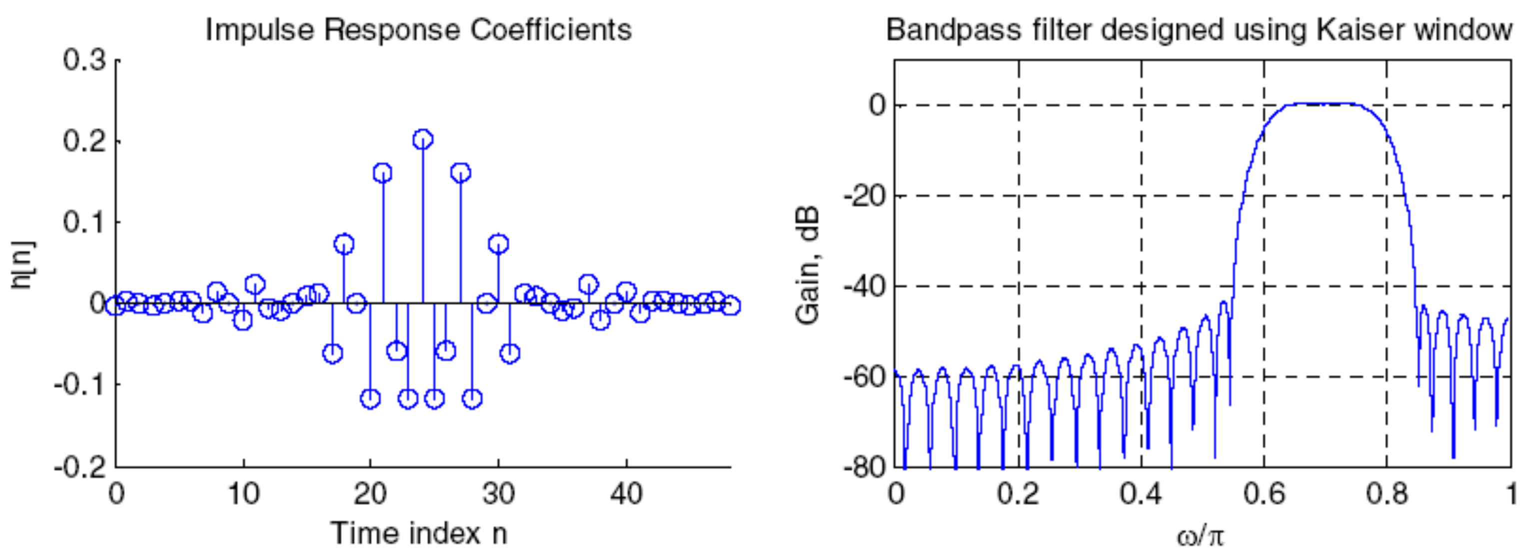
(b) Hann window using `fir1`



(c) Blackman window using `fir1`



(d) Kaiser window using `fir1`



Realization structures for FIR filters

The FIR filter is characterized by the transfer function, $H(z)$, given by

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n}$$

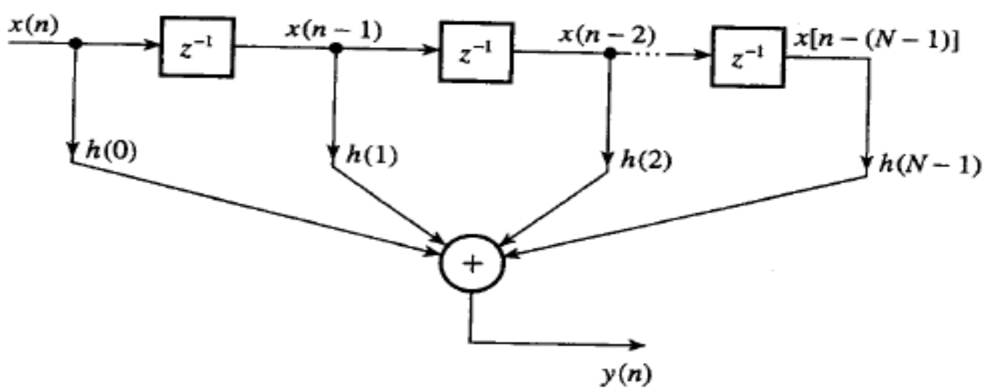
Realization structures are essentially block (or flow) diagram representations of the different theoretically equivalent ways the transfer function can be arranged. In most cases, they consist of an interconnection of multipliers, adders/summers and delay elements. There are many FIR realization structures, but only those that are in common use are presented here.

Transversal structure

The transversal (or tapped delay) structure is depicted in Figure 7.28. The input, $x(n)$, and output, $y(n)$, of the filter for this structure are related simply by

$$y(n) = \sum_{m=0}^{N-1} h(m)x(n-m) \tag{7.39}$$

In the figure, the symbol z^{-1} represents a delay of one sample or unit of time. Thus $x(n-1)$ is $x(n)$ delayed by one sample. In digital implementations, the boxes labelled



z^{-1} could represent shift registers or more commonly memory locations in a RAM. The transversal filter structure is the most popular FIR structure.

The output sample, $y(n)$, is a weighted sum of the present input, $x(n)$, and $N-1$ previous samples of the input, that is $x(n-1)$ to $x(n-N)$. For the transversal structure, the computation of each output sample, $y(n)$, requires

- $N-1$ memory locations to store the $N-1$ input samples,
- N memory locations to store the N coefficients,
- N multiplications, and
- $N-1$ additions.

Linear phase structure

A variation of the transversal structure is the linear phase structure which takes advantage of the symmetry in the impulse response coefficients for linear phase FIR filters to reduce the computational complexity of the filter implementation.

In a linear phase filter, the coefficients are symmetrical, that is $h(n) = \pm h(N-n-1)$. Thus the filter equation can be re-written to take account of this symmetry with a consequent reduction in both the number of multiplications and additions. For type 1 and 2 linear phase filters, the transfer function can be written as

$$H(z) = \sum_{n=0}^{(N-1)/2-1} h(n)[z^{-n} + z^{-(N-1-n)}] + h\left(\frac{N-1}{2}\right)z^{-(N-1)/2} \quad N \text{ odd} \tag{7.40a}$$

$$H(z) = \sum_{n=0}^{N/2-1} h(n)[z^{-n} + z^{-(N-1-n)}] \quad N \text{ even} \tag{7.40b}$$

The corresponding difference equations are given by

$$y(n) = \sum_{k=0}^{(N-1)/2-1} h(k)\{x(n-k) + x[n-(N-1-k)]\} + h[(N-1)/2]x[n-(N-1)/2] \tag{7.41a}$$

$$y(n) = \sum_{k=0}^{(N-1)/2-1} h(k)\{x(n-k) + x[n-(N-1-k)]\} \tag{7.41b}$$

A comparison of Equations 7.39 and 7.41 shows that the linear phase structure is computationally more efficient, requiring approximately half the number of multiplications and additions. However, in most DSP processors Equation 7.39 leads to a more efficient implementation, because the computational advantage in Equation 7.41 is lost in the more complex indexing of data implied.

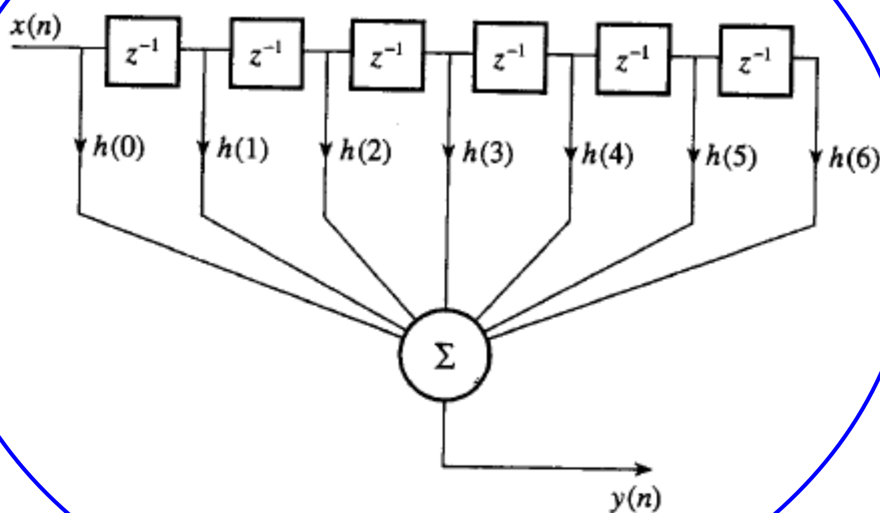
A linear phase FIR filter has seven coefficients which are listed below. Draw the realization diagrams for the filter using (a) direct (transversal) and (b) linear phase structures. Compare their computational complexities.

$h(0) = h(6) = -0.032$

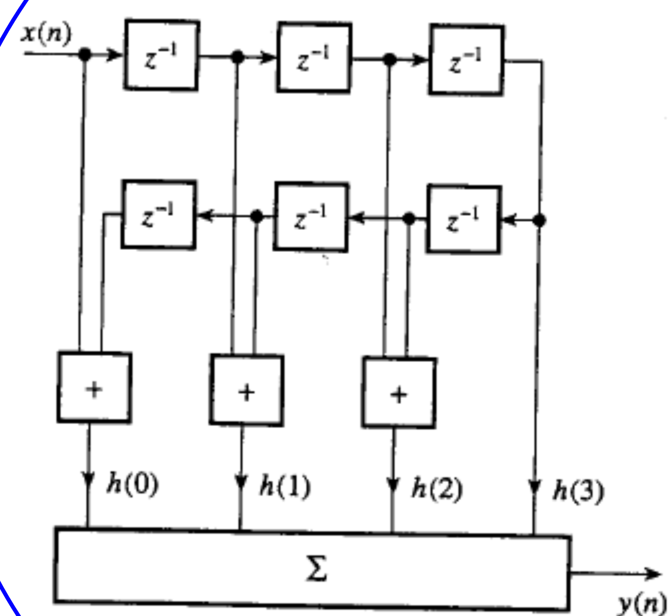
$h(1) = h(5) = 0.038$

$h(2) = h(4) = 0.048$

$h(3) = -0.048$



(a)



(b)