Iran University of Science and Technology

simulation of digital signal processing course (FIR Filter Design)

Seyed Amir Ali Mohaddes

Table of Contents

Abstract

In this project, we are going to design anFIR filter for a noisy signal with a signal-to-noise ratio of ۴۰. First, we count the number of operations required for filtering the signal, then we draw the signal and its fast Fourier transform, the noise with the functionawgn) to the signal. Then we design the filter. Finally, we performdecimation .and design the filter again and report the results

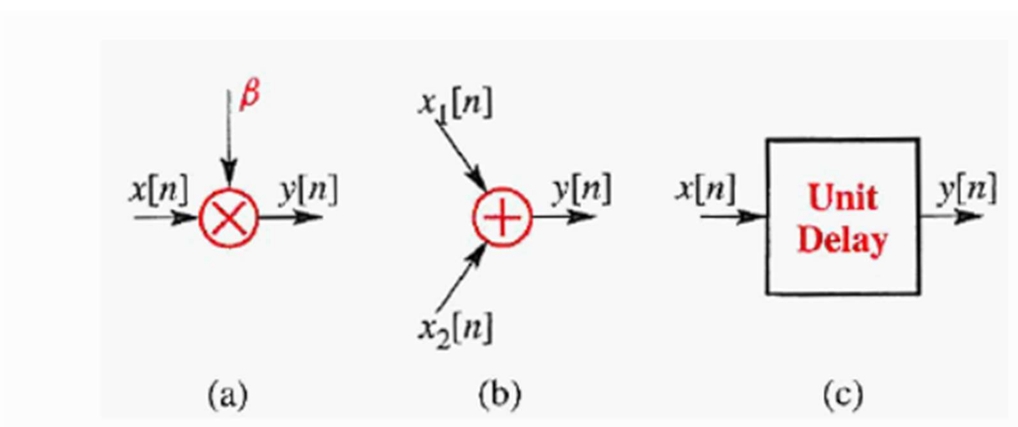
Examining the theory of the problem and presenting the relevant relationships

In anFIR filter the output is a linear combination of the input and past , :inputs and has no feedback from the output and is defined as follows

$$y(n) = \sum_{i=0}^{k} b_i x\,(n-i)$$

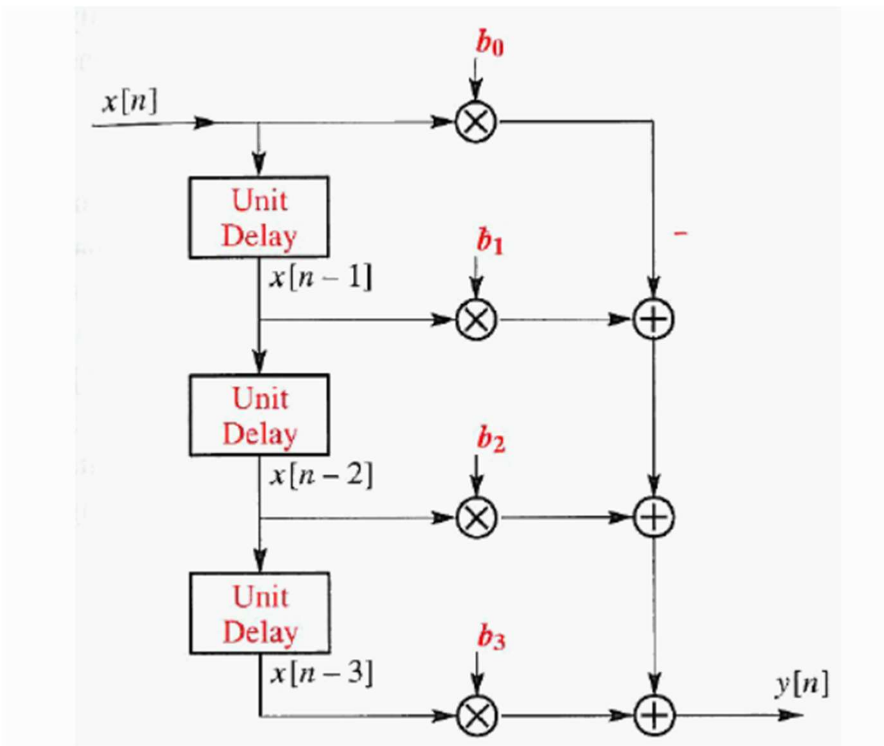
Finally, the goal is to determine the coefficients of this filter in a way to .reach the desired goal

FIR filter building blocks are: ۱- Multiplier ۲- Adder ۳- Delay generator



resolution of R ۱basic systems inFIR filter

Below is the block diagram of anFIR filter :



resolution of R ۲Block diagram of a ۳rd orderFIR filter

for designing theFIR back filter ,such as windowing, Fourier transform , .frequency sampling, etc

be[1] According to the demands of the design problem, it is supposed to ,done by the Kaiser window method, so the concepts of the windowing method .especially the Kaiser window, are explained below

[1] Kaiser

H d (n unit A frequency filter Sample Response to Title (

:Select an idea Consider Al with a linear phase

$$H_d(e^{j\omega}) = A(e^{j\omega})e^{-j(\alpha\omega-\beta)}$$

becauseH d (n) In general, it has an infinite length We have to approximate it using anFIR filter . With the window design method, the desired filter using the window The sample response of the unit is designed:

$$h(n) = h_d(n)w(n)$$

,In this relationw(n) is a window with a finite length, which is equal to zero outside the interval0 ≤ n ≤ N and is symmetric with respect to its midpoint:

$$w(n) = w(N-n)$$

The effect of the window on the frequency response can be obtained from the mixed convolution theory:

$$H(e^{j\omega}) = \frac{1}{2\pi} H_d(e^{j\omega}) * w(e^{j\omega}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\theta})w(e^{-j(\omega-\theta)})d\theta$$

Table 1some From window Hi Usual

Rectangular	$w(n) = \begin{cases} 1 & 0 \le n \le N \\ 0 & \text{else} \end{cases}$
Hanning[1]	$w(n) = \begin{cases} 0.5 - 0.5\cos\left(\frac{2\pi n}{N}\right) & 0 \le n \le N \\ 0 & \text{else} \end{cases}$
Hamming	$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N}\right) & 0 \le n \le N \\ 0 & \text{else} \end{cases}$
Blackman	$w(n) = \begin{cases} 0.42 - 0.5\cos\left(\frac{2\pi n}{N}\right) + 0.08\cos\left(\frac{4\pi n}{N}\right) & 0 \le n \le N \\ 0 & \text{else} \end{cases}$

In addition to these windows introduced in table ١ became Kiser, a new family from the window introduced which are described by the following relation:

$$w(n) = \frac{I_o\left[\beta\left(1-[(n-\alpha)/\alpha]^2\right)\right]}{I_o(\beta)} \qquad 0 \le n \le N$$

that in this regard$\alpha = N/2$ andI_0 Bessel function corrected It is the first type of bile that is used It is obtained from the expansion of the following power series Aide:

$$I_o(x) = 1 + \sum_{k=1}^{\infty}\left[\frac{(x/2)^k}{k!}\right]^2$$

There are two empirical relations for using the Kaiser window in the design ofFIR filters which simplify the design more does The first is the relation of the , attenuation coefficient of the stop band of the low filter. Go$\alpha_s = -20\, log(\delta_s)$ to parameter _β It relates :

$$\beta = \begin{cases} 0.1102(\alpha_s - 8.7) & \alpha_s > 50 \\ 0.5842(\alpha_s - 21)^{0.4} + 0.07886(\alpha_s - 21) & 21 \le \alpha_s \le 50 \\ 0.0 & \alpha_s < 2 \end{cases}$$

and the secondN to the transient bandwidthΔf Stop band attenuation : coefficient It relates

$$N = \frac{\alpha_s - 7.9s}{14.36\Delta f} \quad \alpha_s \ge 21$$

Simulation method

We perform the simulation in MATLAB software step by step according to the provided instructions, and the relevant calculations and answers to the questions :follow

Simulation and presentation of results

Calculations of the number of multiplication and addition operations for a filter of lengthM and a signal to Lengthof N :

:Number of addition operations

In order to create a complete overlap of the filter, an addition operation is performed on theM-1 signal which actually determines only the delay. After ,N overlapping the addition operation takes place. Considering the delay, the , :number of additions is equal to

$$(M-1)+N$$

If we do not consider the delay, we haveN addition operations

The number of times the sigma index changes in convolution was considered as) (.the number of addition operations

:Number of multiplication operations

If Takhbar is calculated, the number of multiplication operations is obtained through arithmetic expansion as follows. We assume thatN > M :

$1,2,3,4,\ldots,(M-1) \rightarrow d=1 \quad \rightarrow \quad M-1+\frac{M-2}{2}$Delay:

$N\ times \quad M \quad \rightarrow \quad N*M$full overlap :

$(M-1),(M-2),\ldots,1 \quad \rightarrow d=-1 \quad \rightarrow \quad M^2-\frac{(M-1)}{2}$Exit:

Number of delayed multiplication operations$M^2+N*M+\frac{M-1}{2}+\frac{(M-2)}{2}$

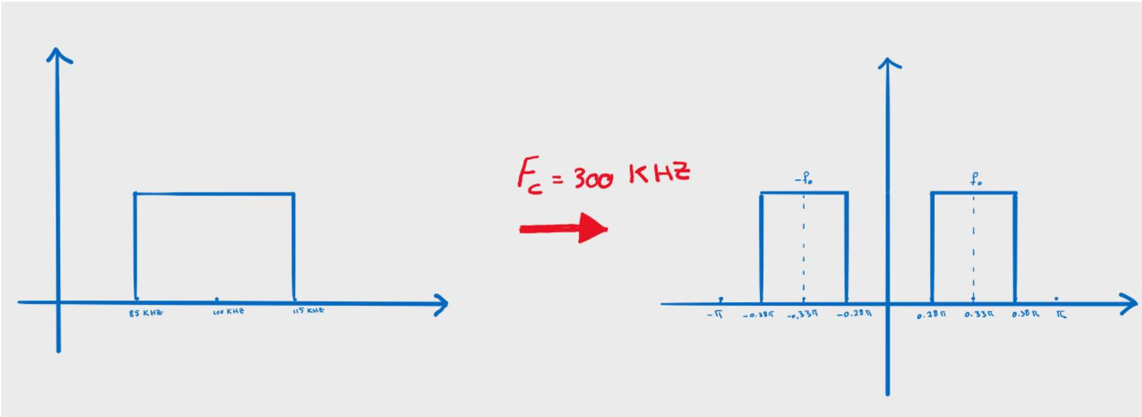
Number of multiplication operations without delay$M^2+N*M-\frac{M-1}{2}$

Review part ۱

Examining the difference betweenpassband andbaseband

Inpassband mode ,The signal is placed in a specific central frequency (modulated) but inbaseband mode.The signal is without modulation (baseband)

Continue to review part ١ of the problem



-) At Shape Above Response The frequency of K _ Letter _ _ It 's passing oh ten _ Ah _ At Interval Image ٢pi,pi (_ draw round ten _ is

:The hit response of this filter will be as follows

$$h[n] = \frac{\omega}{\pi}\text{sinc}[\omega n]$$

So the frequency response in terms ofH(w) :will be as follows

$$B(w) = H(\omega - \omega_0) + H(\omega + \omega_0)$$

:Now the image of the Fourier transform of this filter is calculated as follows

$$b[n] = f^{-1}\{H(\omega - \omega_0) + H(w + \omega_0)\}$$

$$b[n] = e^{j\omega n}h[n] + e^{-jwn}h[n]$$

$$b[n] = 2h[n]\cos(\omega_0 n)$$

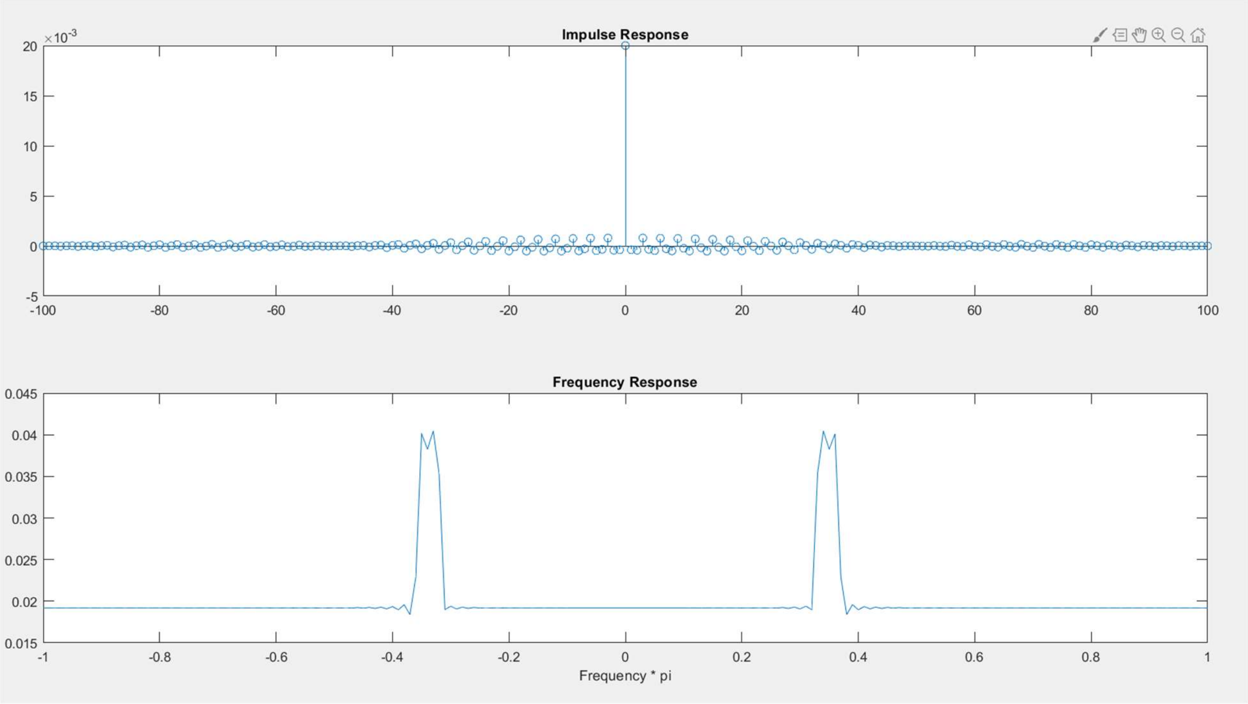
$$b[n] = \frac{\omega}{\Pi} * 2\,\text{sinc}(wn)\cos(\omega_0 n)$$

Now that we have advanced the calculations theoretically, we will perform the :simulation using the following code

```
clc
```

```
clear
close all

n=-100:100;
w0=0.33*2*pi;
w=0.01*2*pi;

bn=(2*w/ pi)* (sin(w*n)./(pi*n)).*cos(w0*n);
bn( 101)=w/pi; Hospital _
subplot(2,1,1 ),stem ( n,bn )
title( 'Impulse Response' )


Hw = fft (bn);

f=-1:1/(( length( Hw )-1)/2):1;
subplot( 2,1,2), plot( f,abs ( Hw ))
title( 'Frequency Response' )
xlabel ( 'Frequency * pi' )
```

The impulse response diagram and the Fourier transform of the signal will be as follows



resolution of R :Impulse response diagram of signalB along with its Fourier transform diagram

$\omega_0 = 0{,}33\pi * 2\pi$

$$\omega = \frac{0,1\pi}{2} * 2\pi$$

Review part ۲

According to part ۱, the Fourier transform of the signal was drawn .according to Figure ۴

Due to the non-causality of the rectangular pulse signal, in general, this signal cannot be drawn correctly, and the spectrum drawn inmatlab always has .differences from our expected spectrum

FFT .command closer to our desired value

.The results obtained can be seen in the graph below

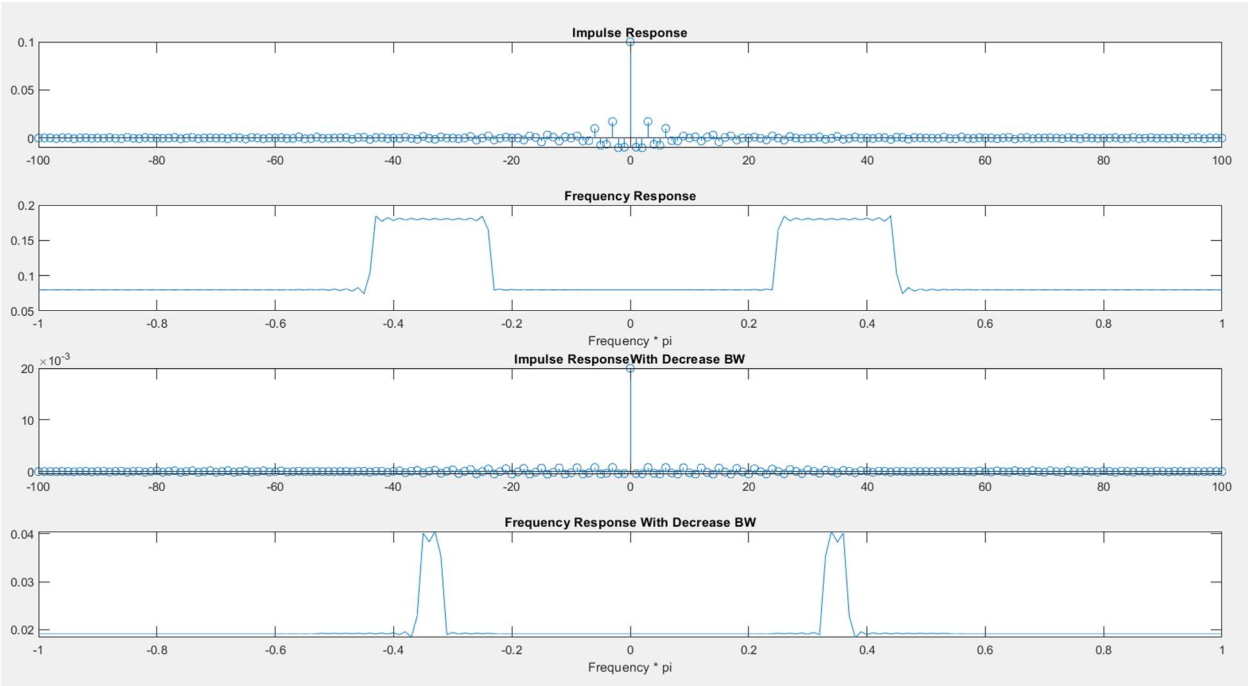


Image ۵Comparison of spectrum and impulse response of the signal after reducing the bandwidth with the initial state

The corresponding MATLAB code can be seen below

```
clc
clear
close all
n=-100:100;
w0=0.33*2*pi;
w=0.05*2*pi;
w2=0.01*2*pi;
bn=(2*w/ pi)* (sin(w*n)./(pi*n)).*cos(w0*n);
bn(101)=w/pi; Hospital
bn2=(2*w2/ pi)* (sin(w2*n)./(pi*n)).*cos(w0*n);
bn2(101)=w2/pi; Hospital
subplot(4,1,1),stem(n,bn)
title( 'Impulse Response' )

Hw=fft(bn);
Hw2=fft(bn2);

f=-1:1/((length(Hw)-1)/2):1;
subplot(4,1,2), plot(f,abs(Hw))
title( 'Frequency Response' )
xlabel( 'Frequency * pi' )

subplot(4,1,3),stem(n,bn2)
title( 'Impulse ResponseWith Decrease BW' )

subplot(4,1,4), plot(f,abs(Hw2))
```

```
title( 'Frequency Response With Decrease BW' )
xlabel ( 'Frequency * pi' )
```

Review of the third part

awgn() function function

The MATLAB communication toolbox has an internal function called awgn() which can be used to add an incremental Gaussian white noise to , obtain the desiredSNR (signal to noise ratio). The main use of this function is to addAWGN to a clean signal with unlimitedSNR In order to obtain a signal with an assumedSNR .

This function takes the input signal and the desiredSNR as input, and the .output is a signal to which Gaussian noise has been added

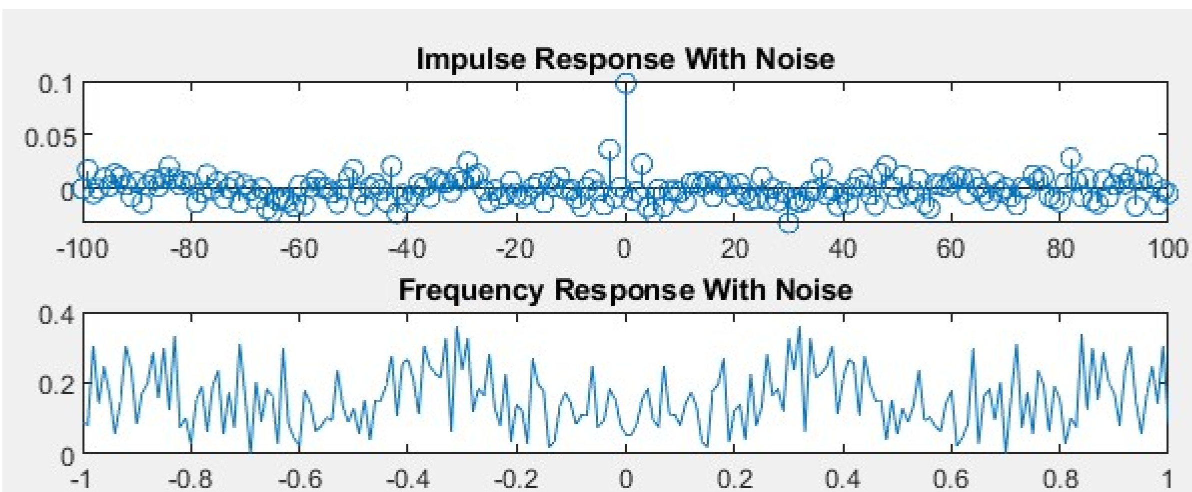


Image 1hit response and the Fourier transform of the signal along withawgn noise

:The relevant MATLAB simulation code

```
clc
clear
close all

n=-100:100;
w0=0.33*2*pi;
w=0.05*2*pi;
```

```
bn=(2*w/ pi)* (sin(w*n)./(pi*n)).*cos(w0*n);
bn( 101)=w/pi; Hospital _

NoiseBn =awgn(bn,40);

DS_NoiseBn = NoiseBn ( 1:2:end );

Hw_N = fft ( NoiseBn );

DS_Hw_N = fft ( DS_NoiseBn );

f=-1:1/(( length( Hw_N )-1)/2):1;
f2=-1:1/(( length( DS_NoiseBn )-1)/2):1;

subplot(4,1,1 ), stem ( n,NoiseBn );
title( 'Impulse Response With Noise' )
subplot(4,1,2 ),plot ( f,abs ( Hw_N ))
title( 'Frequency Response With Noise' )
```

:Review of the fourth part

with thefdatool :toolbox. The specifications of the filter are as follows

Order=56

Wstop1=0.48

Wpass1=0.56

Wpass2=0.76

Wstop2=0.84

Astop1=Astop2=40 db

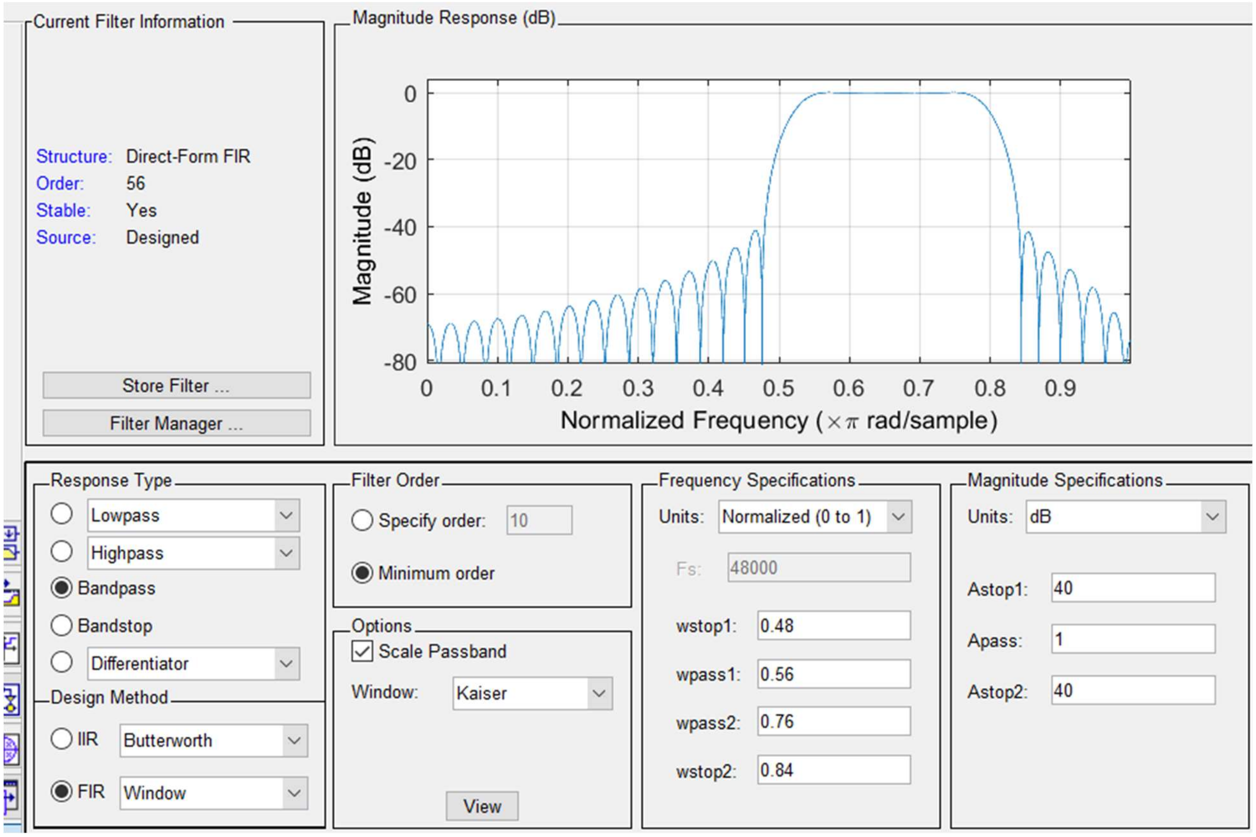


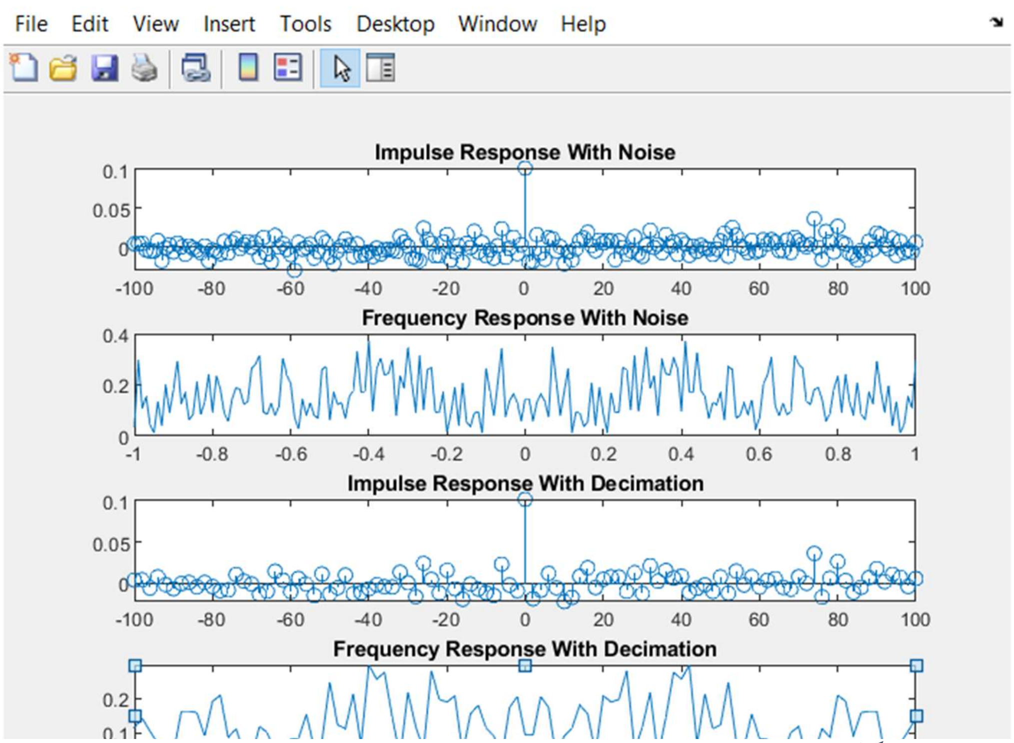
Image ٧filters designed infdatool

Review of the fifth part

First, we write and execute the following code in MATLAB, then we :analyze the results

```
clc
clear
close all

n=-100:100;
w0=0.33*2*pi;
w=0.05*2*pi;

bn=(2*w/ pi)* (sin(w*n)./(pi*n)).*cos(w0*n);
bn( 101)=w/pi; Hospital _

NoiseBn =awgn(bn,40);
```

```
DS_NoiseBn = NoiseBn ( 1:2:end );

Hw_N = fft ( NoiseBn );

DS_Hw_N = fft ( DS_NoiseBn );

f=-1:1/(( length( Hw_N )-1)/2):1;
f2=-1:1/(( length( DS_NoiseBn )-1)/2):1;

subplot(4,1,1 ), stem ( n,NoiseBn );
title( 'Impulse Response With Noise' )
subplot(4,1,2 ),plot ( f,abs ( Hw_N ))
title( 'Frequency Response With Noise' )

subplot(4,1,3 ), stem (n(1:2:end), DS_NoiseBn );
title( 'Impulse Response With Decimation' )
subplot(4,1,4 ),plot (f2,abs( DS_Hw_N ))
title( 'Frequency Response With Decimation' )
```



تصویر ۸ طیف سیگنال که با ضریب ۲ downsample شده است

Analysis

of the results

By performing the decimation operation On a signal with a factor of ۲, as expected, the bandwidth of the signal is doubled.

Suppose that the signal $x(t)$ is sampled at the rate T. Then we have:

$$\sum_{n=-\infty}^{\infty} x(nT)e^{-jwnt} \rightarrow \frac{1}{T}\sum_{k=-\infty}^{\infty} x\left(f-\frac{k}{T}\right)$$

Now, if this signal is at the rate of $\mathbb{M}$ downsample , :we have

$$\sum_{n=-\infty}^{\infty} x(n - MT)e^{-jwn(MT)} \rightarrow \frac{1}{MT}\sum_{k=-\infty}^{\infty} x\left(f - \frac{k}{MT}\right)$$

Review of the sixth part

In this case, with attention By halving the sampling rate and doubling the passband is bandwidth, onlyneeded is doubled in the filter and the order of the .filter does not change

Order=56

Wstop1=0.38

Wpass1=0.46

Wpass2=0.86

Wstop2=0.94

Astop1=Astop2=40 db

Review of the seventh episode

Coefficients of filters in two variablesCoef_Filter andCoef_Filter2 have been .saved

Now the signal contains Gaussian white noise (section ٣) as well as the decimate signal (Section ٥) we pass through the designed filters Coef_Filter and Coef_Filter2 respectively and .check the results

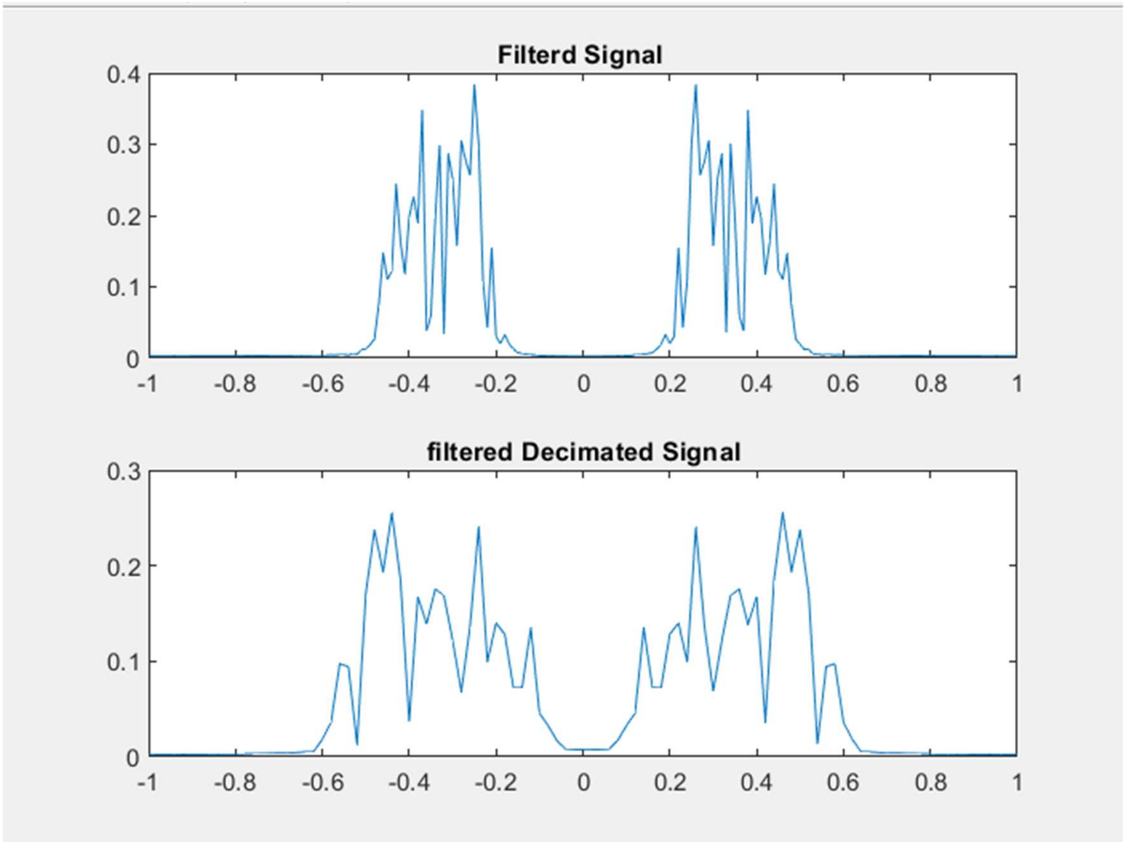


Image ۹Figuresignal of step ۳ and ۵ after passing through the designed filters

The relevant MATLAB code

```
clc
clear
close all

n=-100:100;
w0=0.33*2*pi;
w=0.05*2*pi;

bn=(2*w/ pi)* (sin(w*n)./(pi*n)).*cos(w0*n);
bn( 101)=w/pi; Hopital _

NoiseBn =awgn(bn,40);
DS_NoiseBn=NoiseBn(1:2:end);

Coef_Filter=[-5.28314437091259e-05,-
0.00228983863622678,0.00415496601204029,-1.55126665749000e-
18,-
0.00602105960956121,0.00483544554379611,0.00016452453780058
```

```
4,0.00207934538532895,-0.00674641209654254,-
0.00179325730571808,0.0173633493118839,-
0.0153111953217375,-
0.00312337750034839,0.00813947292652343,0.00287567598899596
,0.00476712701010950, -
0.0332922096283142,0.0315376044106861,0.0156287144823543,-
0.0438612557552704,0.0166943762843342,-
0.00193265939917407,0.0477567425359735,-
0.0581809161140468,-0.0650250634442802,0.202959507396785,-
0.131210039079736,-0.131087715097407,0.281589325730372,-
0.131087715097407,-0.131210039079736,0.202959507396785,-
0.0650250634442802,-0.0581809161140468,0.0477567425359735,-
0.00193265939917407,0.0166943762843342,-
0.0438612557552704,0.0156287144823543,0.0315376044106861,-
0.0332922096283142,0.00476712701010950,0.00287567598899596,
0.00813947292652343,-0.00312337750034839,-
0.0153111953217375,0.0173633493118839,-
0.00179325730571808,-0.00674641209654254,0.002079345
38532895,0.000164524537800584,0.00483544554379611,-
0.00602105960956121,-1.55126665749000e-
18,0.00415496601204029,-0.00228983863622678,-
5.28314437091259e-05];

Coef_Filter2=[0.000162411235492125,0.00355729806723443,-
0.00311893875684266,8.58353706454336e-19,-
0.00451972798793100,0.00751193591331289,0.00050577140386458
2,-0.00137996542519361,-0.00669382686865763,-
0.00199446584387669,0.0145748499670944,-
0.00406340651682391,0.00218096041782039,-
0.0248554627922029,0.0205987512329366,0.00324959089545704,0
.0144021676114968,-0.0285386879134216, -
0.0155068957287943,0.0287606613450254,0.0111901304239521,0.
0257854081217173,-0.0966161610750685,0.0419414272706542,-
0.00823209040488874,0.160196934896505,-0.168628407776185,-
0.209113661650226,0.478961939013600,-0.209113661650226,-
0.168628407776185,0.160196934896505,-
0.00823209040488874,0.0419414272706542,-
0.0966161610750685,0.0257854081217173,0.0111901304239521,0.
0287606613450254,- 0.0155068957287943,-
0.0285386879134216,0.0144021676114968,0.00324959089545704,0
.0205987512329366,-0.0248554627922029,0.00218096041782039,-
0.00406340651682391,0.0145748499670944,-
0.00199446584387669,-0.00669382686865763,-0.0013799654251
9361,0.0005571403864582,0.007511935912812 9, -
```

```
0.004519727983100,8.58353706454545436e-19, -
0.003113189386666666666666666666666666666666666666666
666524444

tic
y = filter(Coef_Filter,1,NoiseBn);
toc
DS_NoiseBn = NoiseBn ( 1:2:end );
tic
y2 = filter( Coef_Filter2,1,DS_NoiseBn);
toc
Hw_F = fft (y);
DS_Hw_F = fft (y2);

f=-1:1/(( length( Hw_F )-1)/2):1;
f2=-1:1/(( length( DS_NoiseBn )-1)/2):1;


subplot( 2,1,1), plot( f,abs ( Hw_F ))
title( ' Filtered Signal' )

subplot( 2,1,2), plot(f2,abs( DS_Hw_F ))
title( 'filtered Decimated Signal' )
```

:Also, the time spent in the process of filtering the signals is as follows

:First signal

```
Elapsed time is 0.000045 seconds
```

:Second signal

```
Elapsed time is 0.000016 seconds
```

Conclusion

The conclusion of each part of the simulation has been reviewed while .presenting the simulation results

List of images and charts