بسم الله الرحمن الرحیم

پردازش سیگنال دیجیتال

فصل دوم :

نویز

مجتبی قربانی

نویز چیست ؟

تعریف نویز :

در یک تعریف کلی، به هر نوسان و تغییر غیرعمدی که بر روی سیگنال های مورد اندازه گیری ظاهر می شود، noise نویز گفته می شود .

نویز و انواع آن :

اطراف ما و هرکجا که برویم نویز حضور دارد.

برای مثال در خیابان، خودرو، اداره و ... . نویز به شکل های متفاوت در زندگی روزانه ما وجود دارد.

نویزها می توانند **ایستان** باشند یعنی مشخصات آماری آنها در طول زمان تغییر نکند مثل نویز فن کامپیوتر.

نویزها همچنین می توانند **غیر ایستان** باشند مانند نویز داخل رستوران یا چند گوینده که همزمان با هم صحبت می کنند.

حذف کدام نوع نویز مشکل تر است ؟

نویز و انواع آن :

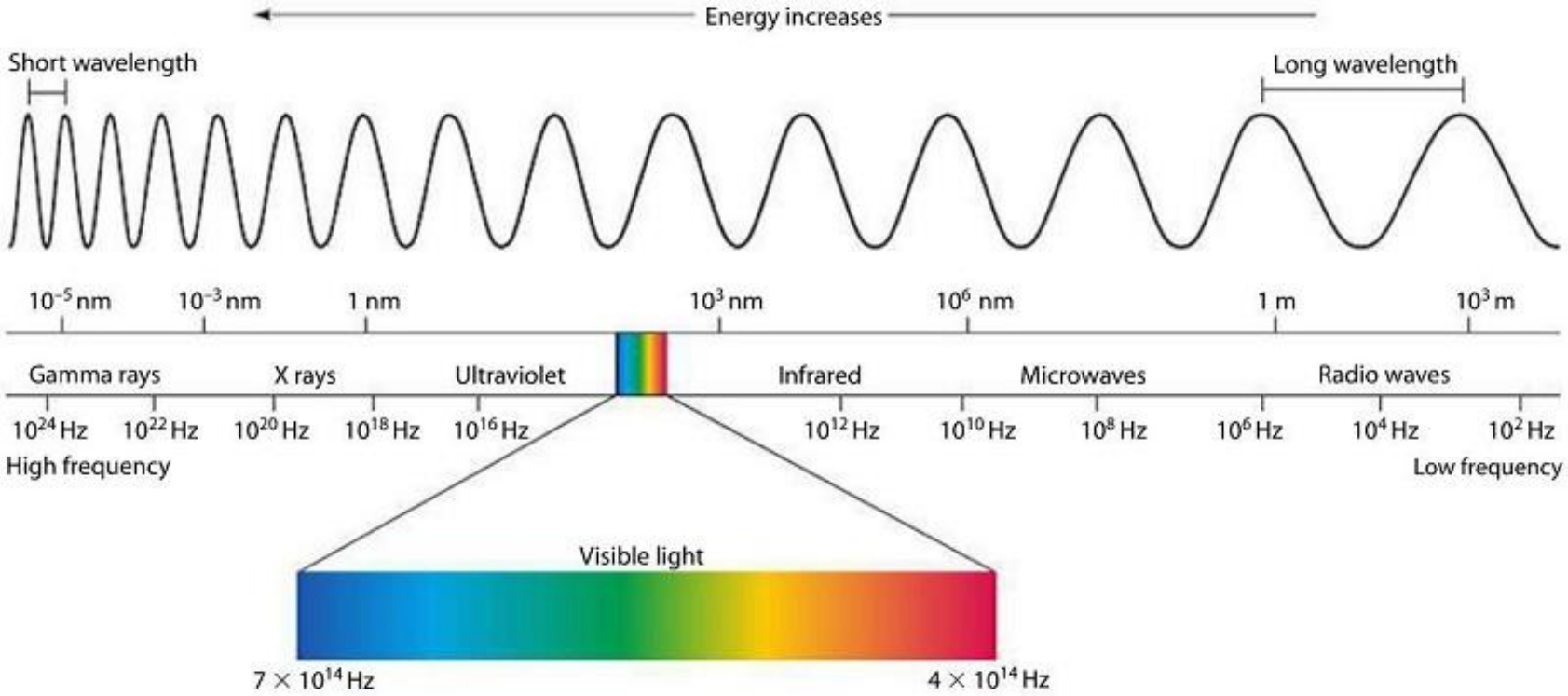
دیگر خصوصیت مشخص انواع نویز، حالت طیف آنها است مخصوصاً توزیع انرژی نویز در حوزه ی فرکانس.

برای مثال انرژی اصلی نویز باد در فرکانس پایین متمرکز شده است. از سوی دیگر نویز رستوران دامنه فرکانسی وسیع تری دارد.

فرکانس: تعداد تکرار یک رخداد در یک واحد زمانی معین است.

برای محاسبه ی فرکانس بر روی یک بازه زمانی ثابت، تعداد دفعات وقوع آن حادثه را در آن بازه می‌شماریم و سپس این تعداد را بر طول بازه زمانی تقسیم می‌کنیم. یک هرتز به این معنی است که یک رویداد یک‌بار در هر ثانیه رخ می‌دهد.

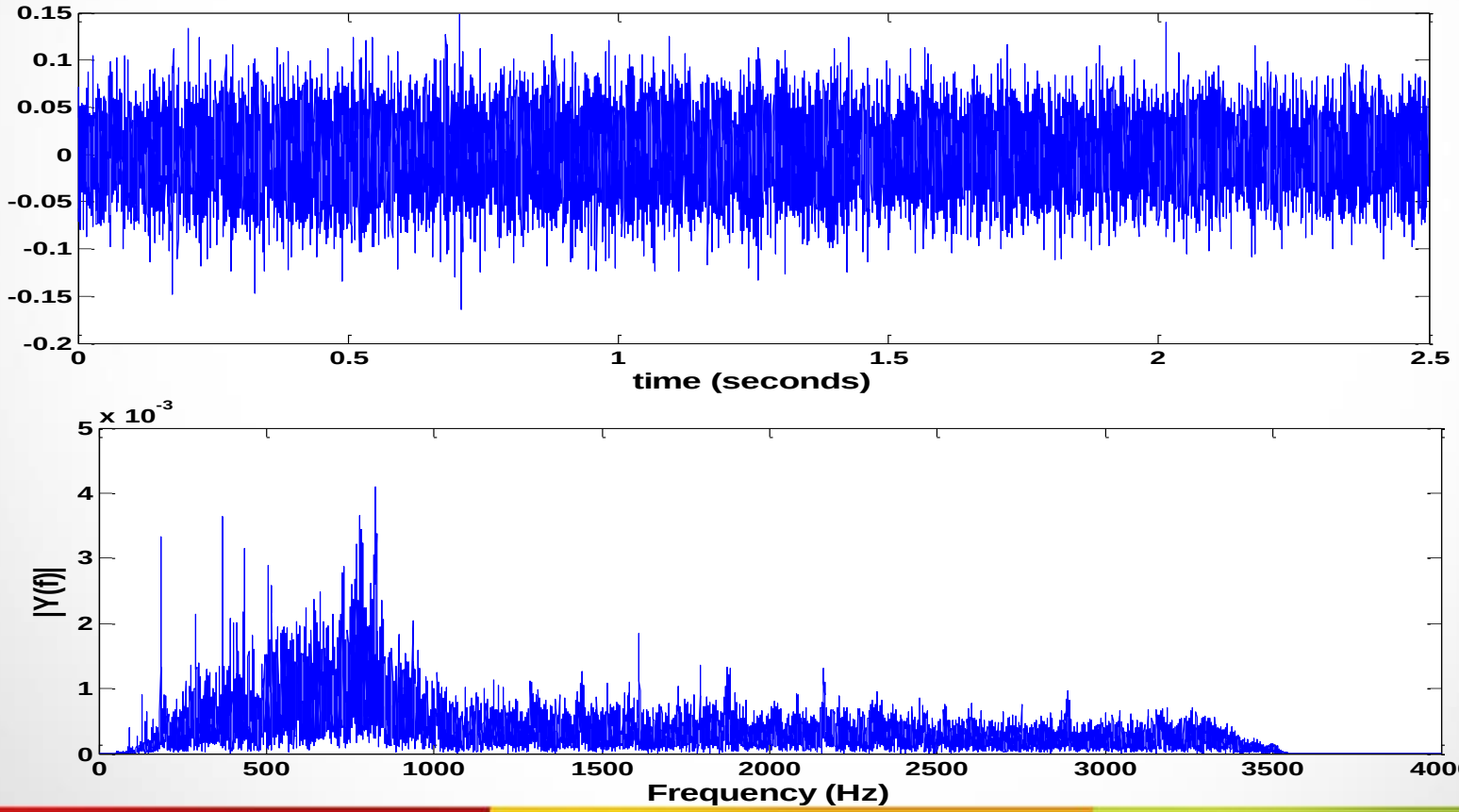
مختصری در مورد فرکانس :



انرژی و فرکانس سیگنال مفاهیم مجزایی هستند.

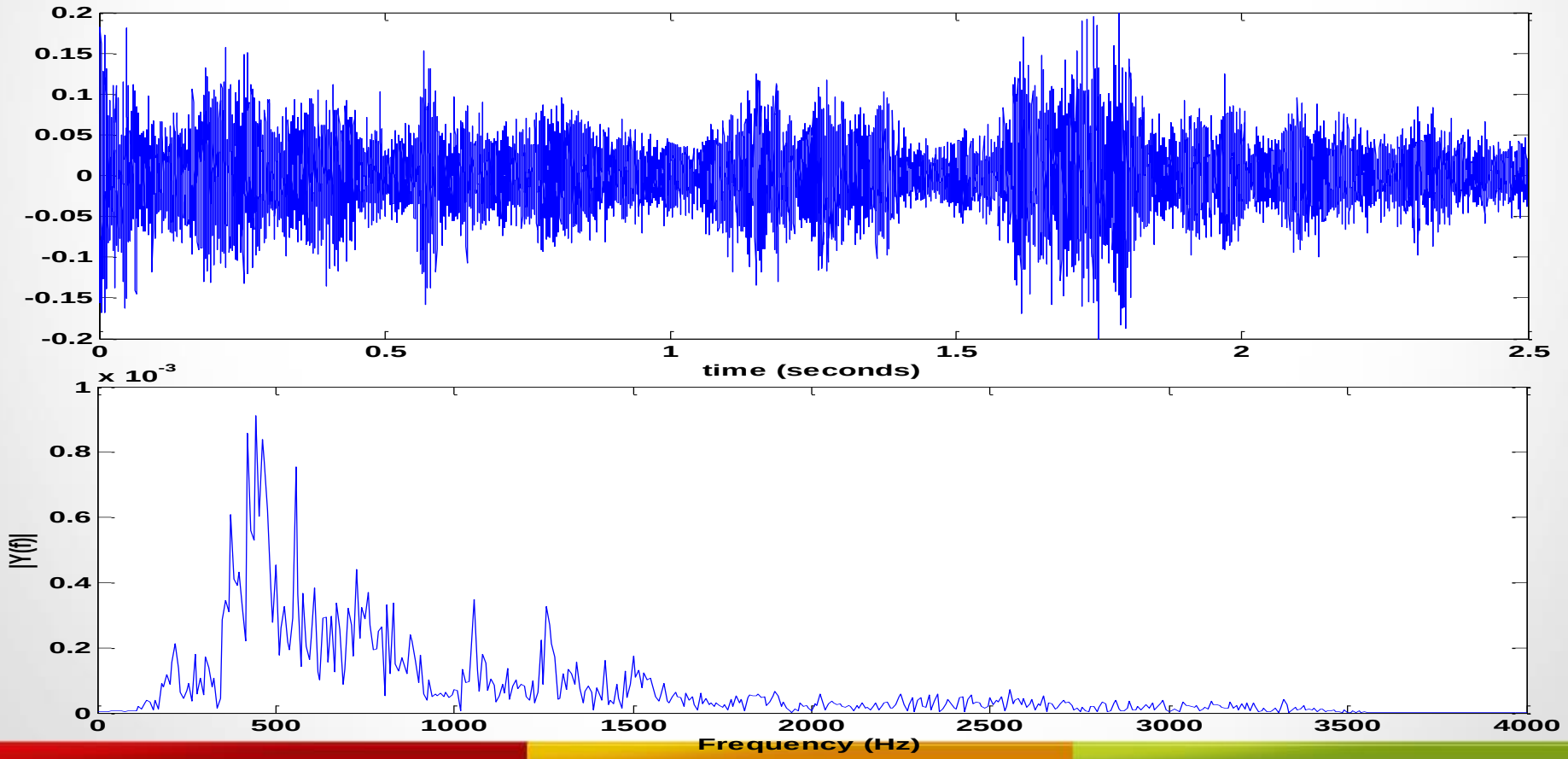
نویز و انواع آن :

شکل موج و طیف نویز خودرو



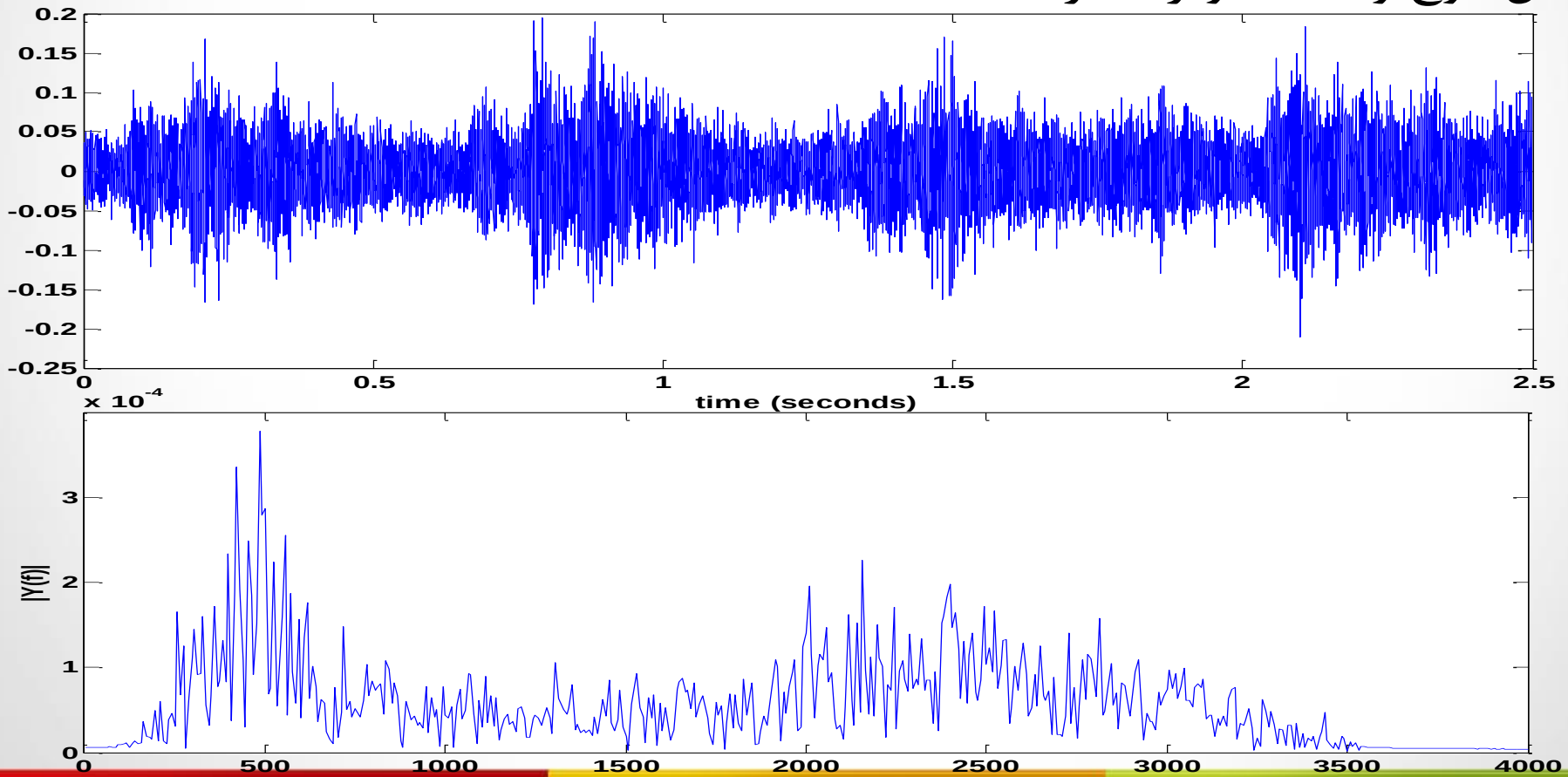
نویز و انواع آن :

شکل موج و طیف نویز رستوران



نویز و انواع آن :

شکل موج و طیف نویز قطار



نویز و انواع آن :

متوسط طیف این منابع نویز در زیر هر سیگنال نشان داده شده بود .

در این ۳ مثال، نویز ماشین تقریباً ایستان است اما نویز قطار و رستوران این طور نیستند. می‌توان به روشنی فهمید که بین انواع مختلف نویز در حوزه فرکانس تفاوت بیشتری نسبت به حوزه زمان وجود دارد.

نویز سفید : یک سیگنال تصادفی است که به هر باند فرکانسی انرژی برابر با بقیه باندها در طیف فرکانس اختصاص می دهد.

حذف نویز:

انسان همواره راهکارهای مختلفی برای غلبه بر نویز اتخاذ کرده است.

برای مثال هنگامی که در یک رستوران باشد برای اینکه بتواند صدای خود را به طرف مقابل برساند، با شدت بیشتری صحبت می کند و یا در هنگام خواب برای اینکه صدای پیرامون او را اذیت نکند با استفاده از پنبه، گوش خود را از شنیدن صداهای مزاحم مصون می کند.

می توانید مثال های طبیعی دیگری ذکر کنید؟

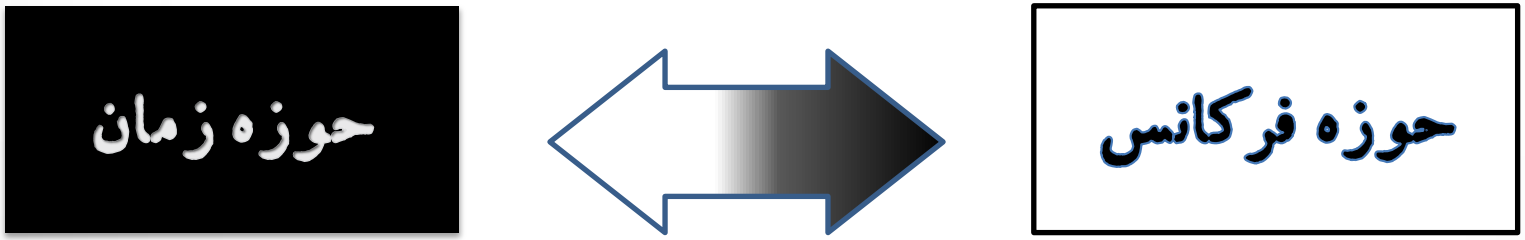
حذف نویز:

به طور کلی برای حذف اثر نویز، می توان چهار راهکار در نظر گرفت :

1. از بین بردن منبع تولید نویز
2. افزایش قدرت سیگنال اصلی برای کم کردن اثرات نامطلوب نویز
3. جلوگیری از اثرگذاری نویز و به نوعی مانع شدن از رسیدن نویز
4. حذف نویز به کمک محاسبات و الگوریتم های مهندسی

کدام راهکار مناسب تر است ؟

انتقال به حوزه فرکانس :



تبدیل فوریه :

تبدیل فوریه یک تبدیل انتگرالی است که هر تابع تعریف شده در حوزه زمان (f(t)) را به یک تابع تعریف شده در حوزه فرکانس (F(w)) تبدیل می کند.

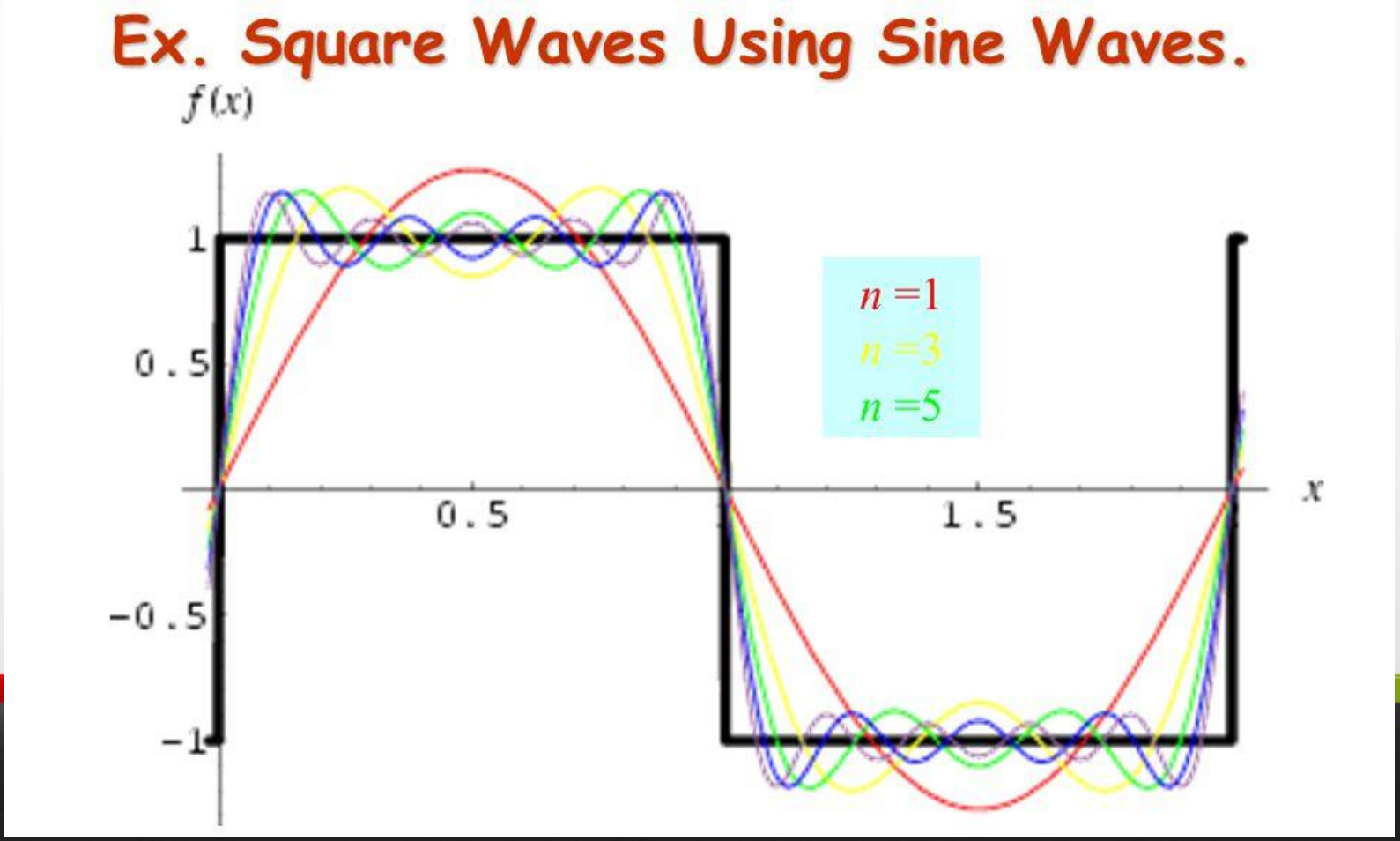
$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\omega t}\, dt$$

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t}\, d\omega$$

حالت خاص تبدیل فوریه، سری فوریه نام دارد و آن زمانی کاربرد دارد که تابع (f(t)) متناوب باشد یعنی (f(t)=f(t+T)) .

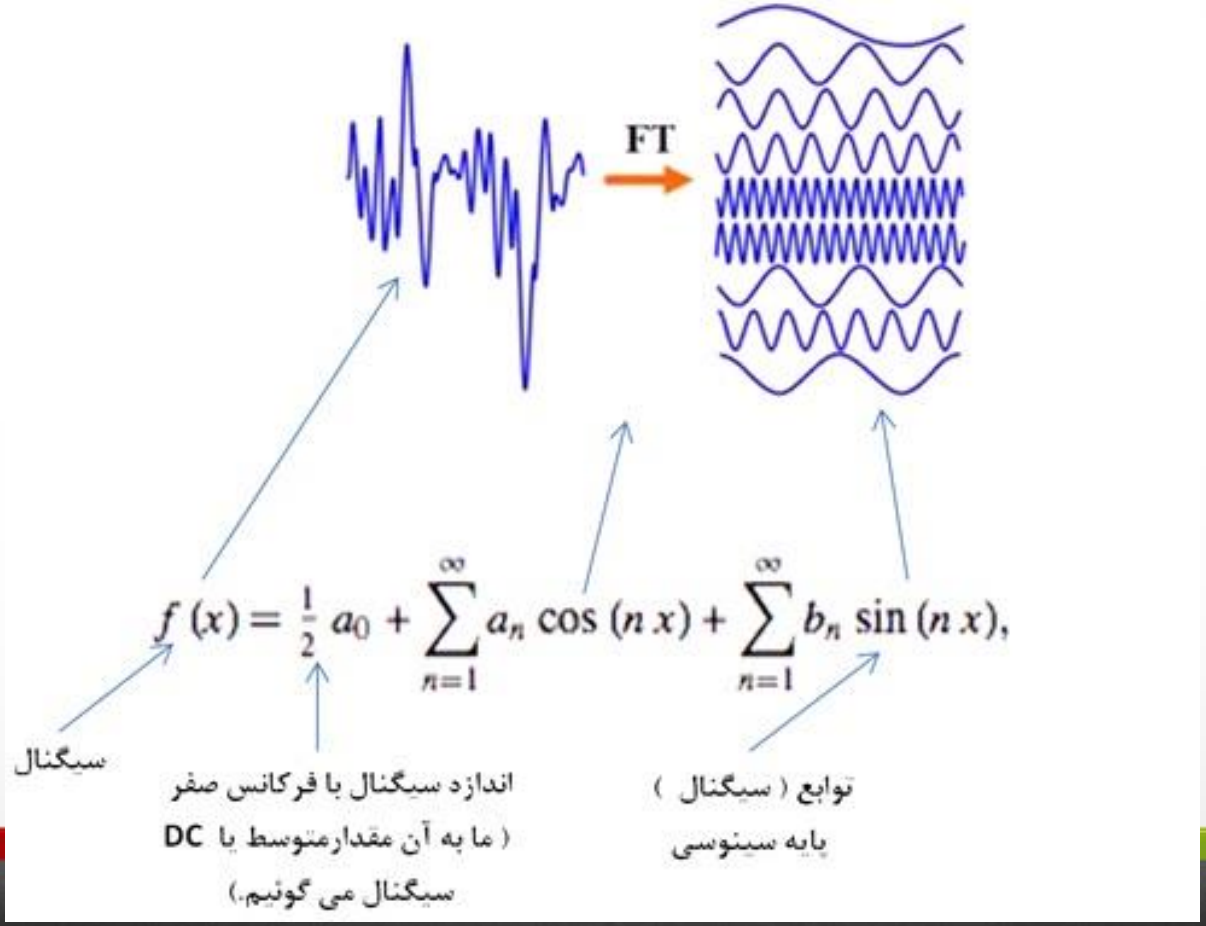
سری فوریه :

با کمک سری فوریه می توان توابع مختلف را با استفاده از مجموعه ای از سینوس ها و کسینوس ها ساخت.



سری فوریه :

هر چقدر تعداد جملات بیشتر شود، شباهت تابع ساخته شده بیشتر خواهد بود.



تبدیل فوریه :

برای یک سیگنال پیوسته و گسسته تبدیل فوریه به صورت زیر خواهد بود.

$$X(f) = \int_{-\infty}^{+\infty} x(t)\, e^{-j2\pi ft}\, dt$$

$$X[m] = \sum_{n=0}^{N-1} x[n]\, e^{-j2\pi \frac{m}{N} n} \qquad m = 0, 1, \ldots, N-1$$

تبدیل فوریه :

برای مثال فرض کنید که سیگنال گسسته ای به صورت زیر مفروض بوده و می خواهیم تبدیل فوریه گسسته آن را محاسبه نماییم.

$$x[0] = 1,\ x[1] = 2,\ x[2] = 2,\ x[3] = 1$$

$$X[m] = \sum_{n=0}^{3} x[n]\, e^{-2\pi j \frac{m}{4} n} = 1 + 2e^{-2\pi j \frac{m}{4}} + 2e^{-\pi j m} + e^{-\pi j \frac{3m}{2}} \qquad m = 0, 1, 2, 3$$

$$X[0] = 1 + 2 + 2 + 1 = 6$$

$$X[1] = 1 + 2e^{-\frac{\pi j}{2}} + 2e^{-\pi j} + e^{-\pi j \frac{3}{2}}$$

$$X[2] = 1 + 2e^{-\pi j} + 2e^{-2\pi j} + e^{-3\pi j}$$

$$X[3] = 1 + 2e^{-\frac{3}{2}\pi j} + 2e^{-3\pi j} + e^{-\frac{9}{2}\pi j}$$

تبدیل فوریه :

با توجه به رابطه $e^{j\theta} = \cos\theta + j\sin\theta$ خواهیم داشت:

$$X[0] = 1+2+2+1 = 6$$

$$X[1] = 1+2(\cos(-\frac{\pi}{2})+j\sin(-\frac{\pi}{2}))+2(\cos(-\pi)+j\sin(-\pi))+(\cos(-\frac{3}{2}\pi)+j\sin(-\frac{3}{2}\pi)) = -1-j$$

$$X[2] = 1+2(\cos(-\pi)+j\sin(-\pi))+2(\cos(-2\pi)+j\sin(-2\pi))+(\cos(-3\pi)+j\sin(-3\pi)) = 0$$

$$X[3] = 1+2(\cos(-\frac{3}{2}\pi)+j\sin(-\frac{3}{2}\pi))+2(\cos(-3\pi)+j\sin(-3\pi))+(\cos(-\frac{9}{2}\pi)+j\sin(-\frac{9}{2}\pi)) = -1+j$$

تبدیل فوریه سریع :

تبدیل سریع فوریه (Fast Fourier transform – FFT) نام الگوریتمی است برای انجام تبدیلات مستقیم و معکوس گسسته ی فوریه به صورتی سریع و بسیار کارآمد.

تعداد زیادی الگوریتم‌های تبدیل فوریه سریع مجزا وجود دارد که شامل محدوده عظیمی از ریاضیات می‌شوند: از محاسبات ساده به وسیله اعداد مختلط تا نظریه اعداد.

تبدیل فوریه سریع تبدیل فوریه گسسته را محاسبه می کند و دقیقاً همان نتایجی را تولید می کند که مستقیماً با تعریف تبدیل فوریه گسسته به دست می آید تنها تفاوت آن این است که بسیار سریع تر است .

متلب ...

برای محاسبه تبدیل فوریه در متلب می توان از دستور زیر استفاده نمود.

fourier

Fourier transform

Syntax

```
fourier(f,trans_var,eval_point)
```

```
>> syms x
>> f = exp(-x^2);
>> fourier(f)

ans =

pi^(1/2)*exp(-w^2/4)
```

متلب ...

اما باتوجه به اینکه هدف ما از استفاده تبدیل فوریه، تحلیل در حوزه فرکانس است ؛ لذا روش تبدیل فوریه سریع استفاده می کنیم. برای استفاده از این روش در نرم افزار متلب کافیست از دستور معرفی شده زیر استفاده کنیم.

fft

Fast Fourier transform

Syntax

```
Y = fft(x)
Y = fft(X,n)
Y = fft(X,[],dim)
Y = fft(X,n,dim)
```

مثال ۱ :

مثال قبل را با این روش در متلب محاسبه می کنیم.

$$x[0] = 1, x[1] = 2, x[2] = 2, x[3] = 1$$

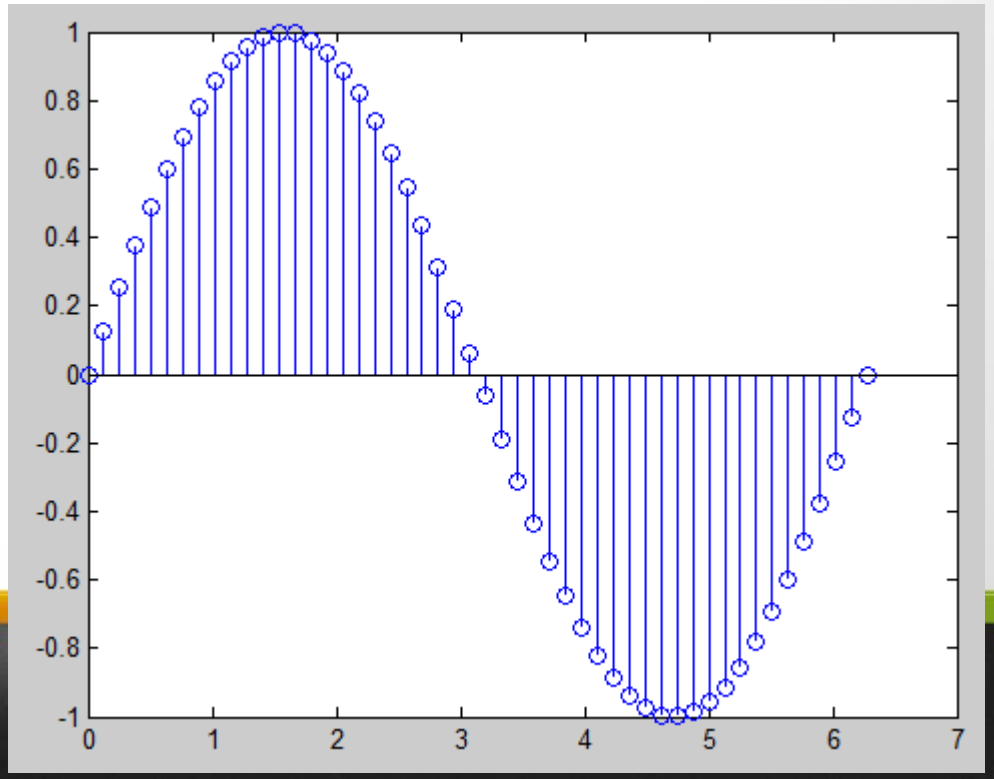
```
1 -     clear all;
2 -     close all;
3 -     clc;
4 -     x=[1 2 2 1];
5 -     X = fft(x)
```

```
  6.0000 + 0.0000i
 -1.0000 + 1.0000i
  0.0000 + 0.0000i
 -1.0000 - 1.0000i
```

معرفی دستور stem :

با توجه به اینکه در توابع گسسته فقط مقادیر در نقاط صحیح وجود دارد، لذا لازم است نمودار به گونه ای نشان داده شود که این نکته مورد توجه قرار گیرد. برای این موارد می توان از دستور stem استفاده نمود.

```
1 -     X = linspace(0,2*pi,50);
2 -     Y = sin(X);
3 -     stem(Y)
```

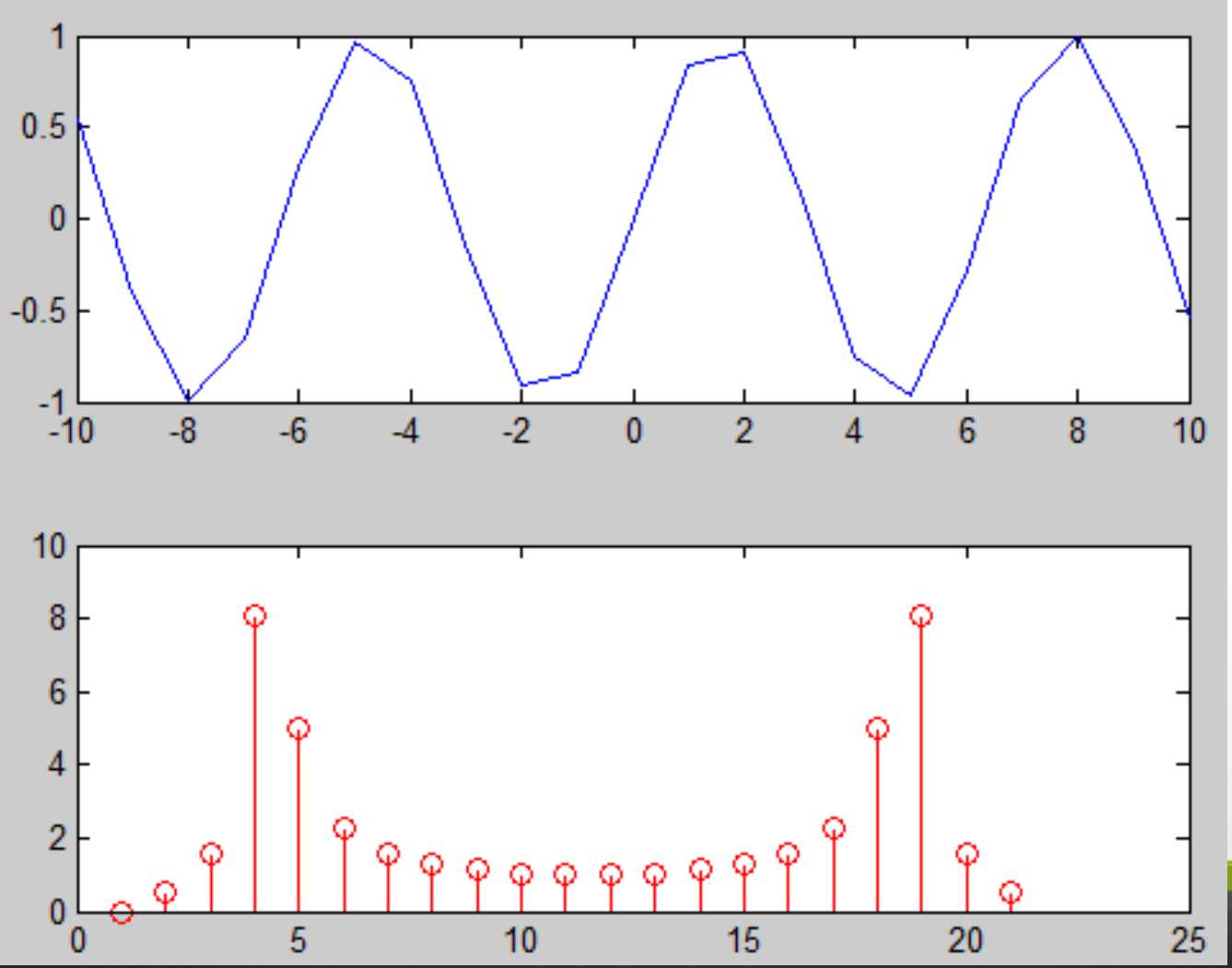


مثال ۲:

در این مثال می خواهیم برای یک تابع سینوسی ساده ، طیف فرکانسی آن را محاسبه و مقدار اندازه (با توجه به مختلط بودن مقادیر) آن را نشان دهیم.

```
1 -    clear all;close all;clc;
2 -    x=-10:1:10;
3 -    y=(sin(x));
4 -    subplot(2,1,1);
5 -    plot(x,y)
6 -    y1=fft(y);
7 -    subplot(2,1,2);
8 -    stem(abs(y1),'r')
```

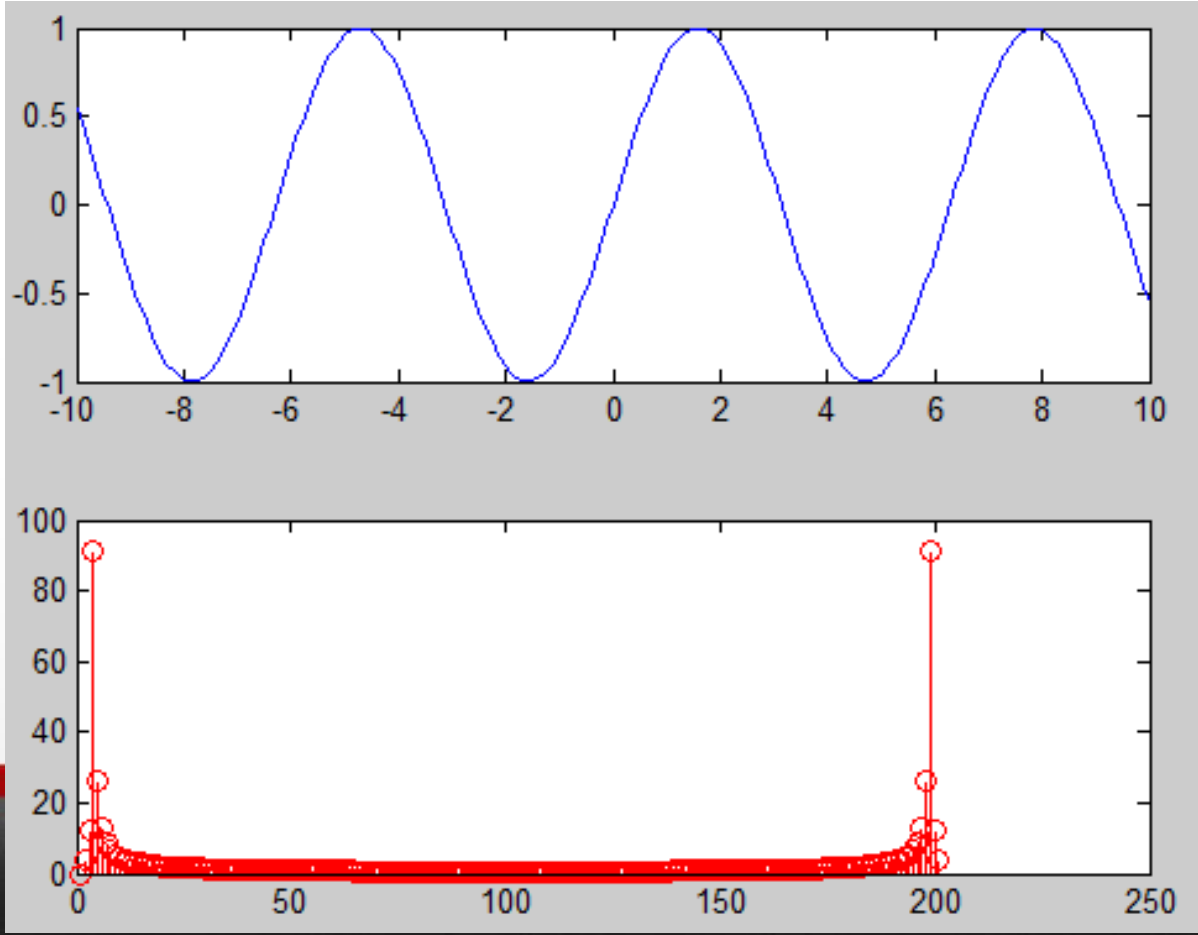
مثال ۲:



مثال ۲:

حال طول گام ها را کمتر می کنیم، تا شکل سینوسی نرم تر و شبیه تر بشود.

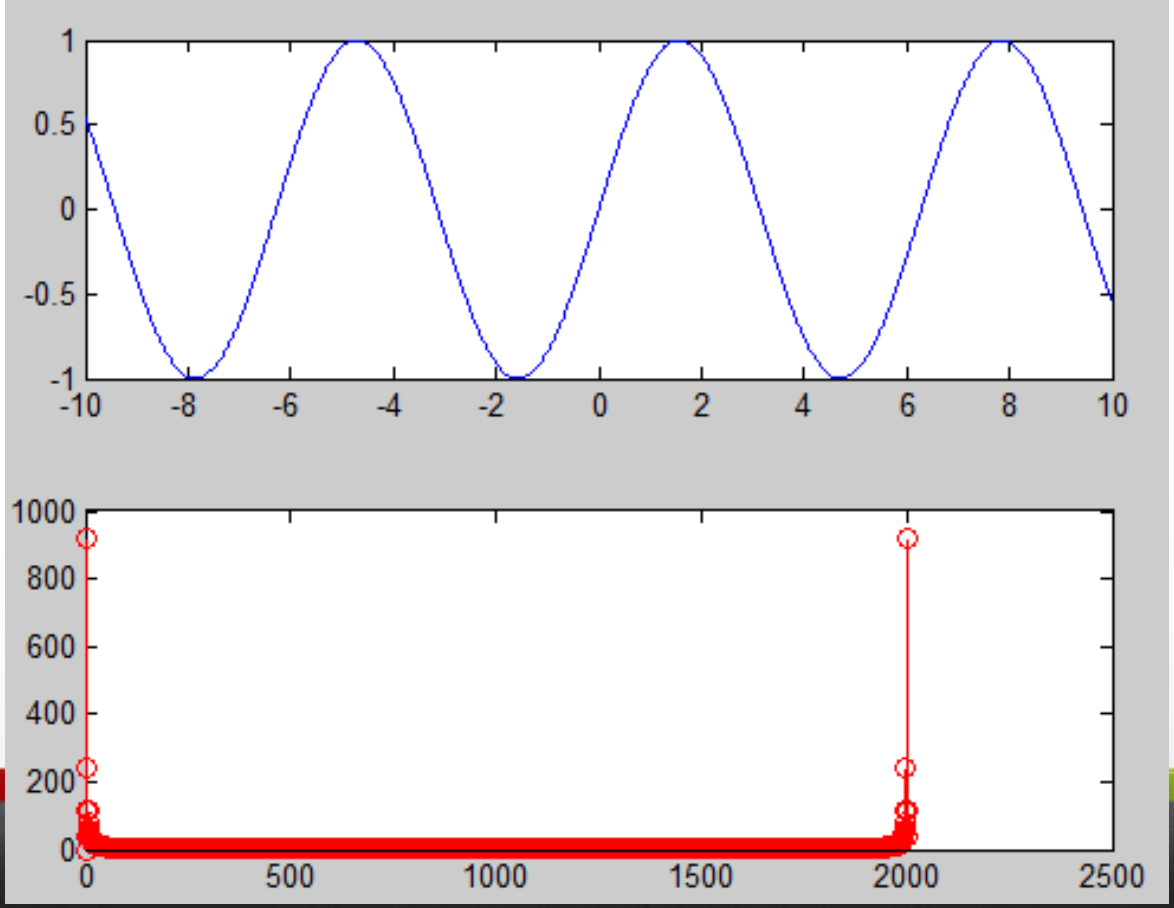
```
2 —     x=-10:0.1:10;
```



مثال ۲:

و با کمتر شدن طول گام ها نتیجه زیر حاصل می شود.

```
2 —     x=-10:0.01:10;
```



تناوب طیف فرکانسی :

با توجه به رابطه تبدیل فوریه گسسته، مشخصا نتیجه گرفته می شود که این تبدیل متناوب بوده و دوره تناوب آن نیز برابر زیر می باشد.

$$X(e^{j\omega}) = X(e^{j[\omega+2\pi]})$$

لذا برای نشان دادن طیف فرکانسی، لازم است تا طیف فرکانسی را در محدوده ای به طول $[2\pi]$ در نظر بگیریم.

در متلب، به صورت پیش فرض ابتدا طیف برابر صفر و انتهای آن برابر 2π است.

پژوهشگران برای نمایش طیف فرکانسی از $[-\pi, +\pi]$ استفاده می نمایند.

اصلاح نمایش طیف فرکانسی:

با استفاده از دستور زیر، نمایش طیف از بازه $[0,2\pi]$ به بازه $[-\pi,+\pi]$ منتقل می شود.

fftshift

Shift zero-frequency component to center of spectrum

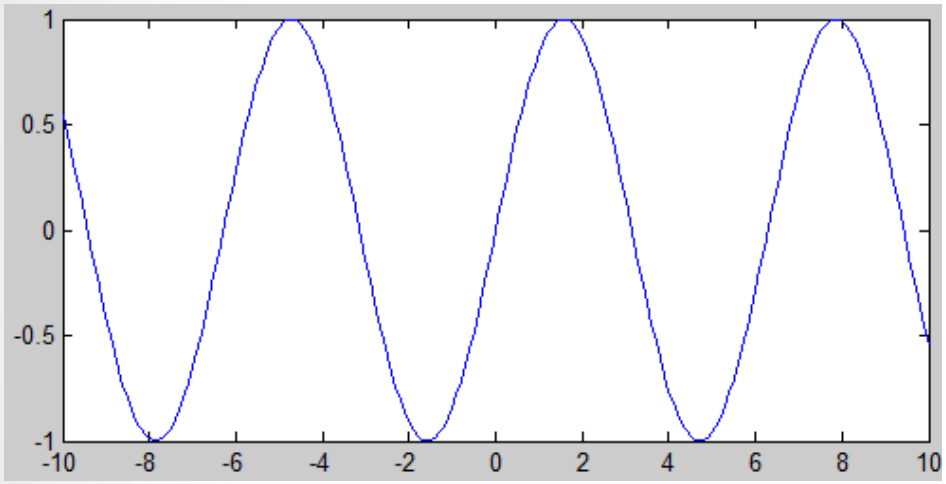
Syntax

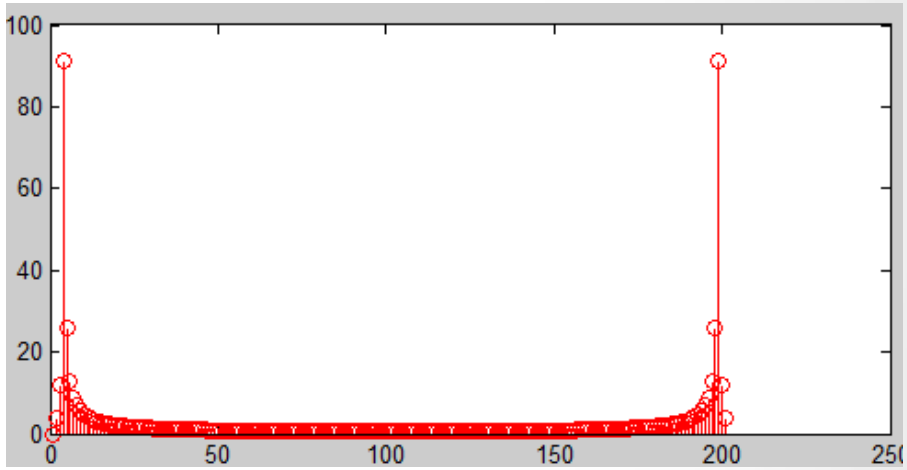
```
Y = fftshift(X)
Y = fftshift(X,dim)
```

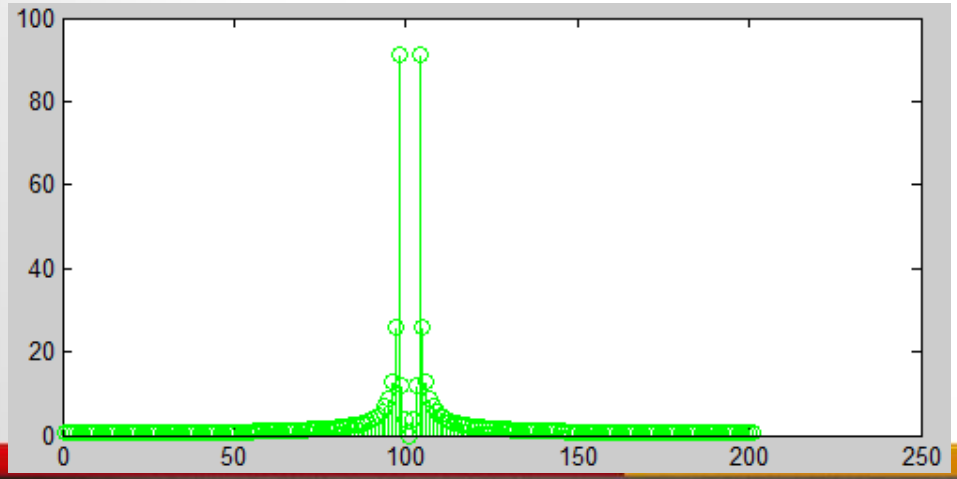
مثال ۲:

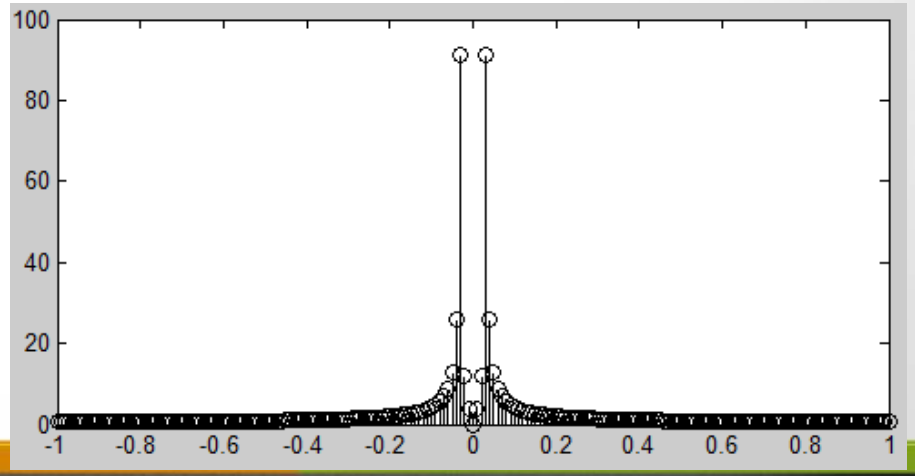
```
clear all;close all;clc;
x=-10:0.1:10;
y=(sin(x));
subplot(2,2,1);
plot(x,y)
y1=fft(y);
subplot(2,2,2);
stem(abs(y1),'r')
subplot(2,2,3);
Y2=fftshift(y1);
stem(abs(Y2),'g');
x2=linspace(1,-1,length(Y2));
subplot(2,2,4);
stem(x2,abs(Y2),'k');
```

مثال ۲:


1
0.5
0
-0.5
-1
-10
-8
-6
-4
-2
0
2
4
6
8
10


100
80
60
40
20
0
0
50
100
150
200

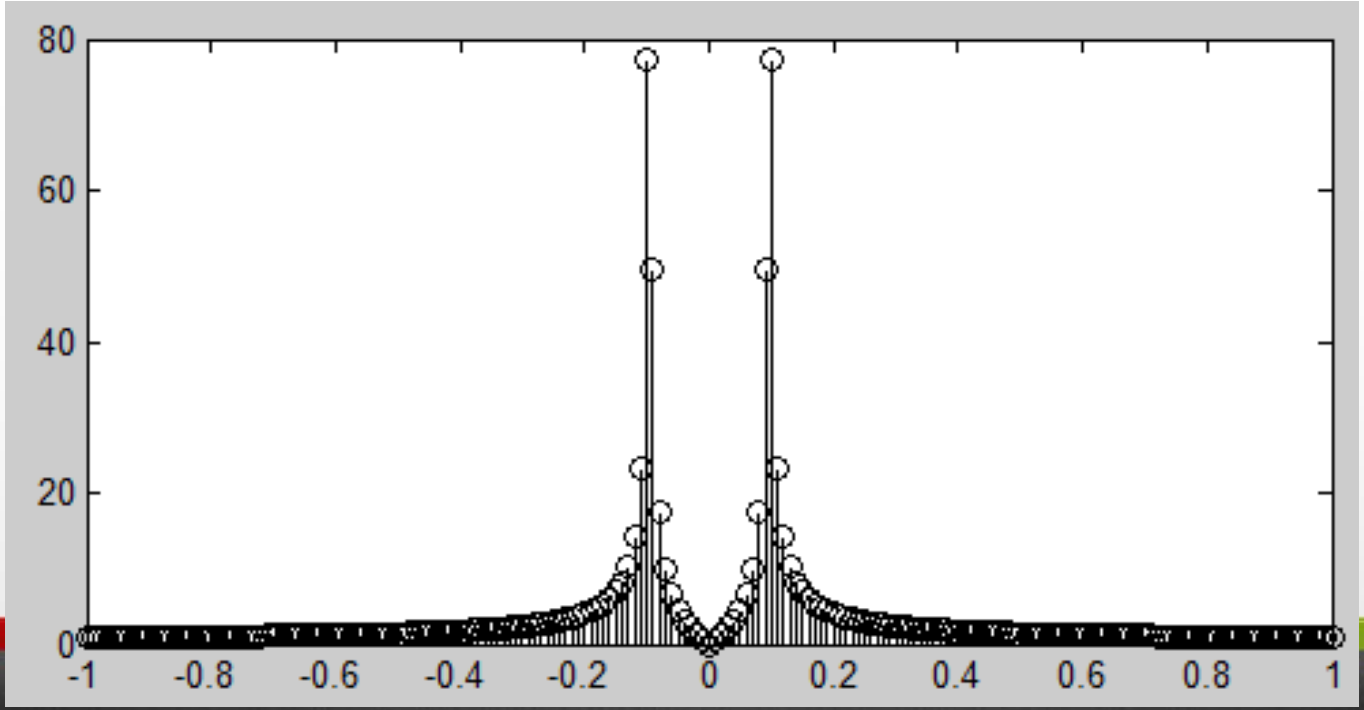

100
80
60
40
20
0
0
50
100
150
200
250


100
80
60
40
20
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

مثال ۲:

حال با تغییر فرکانس تابع، طیف فرکانسی را بررسی می کنیم.

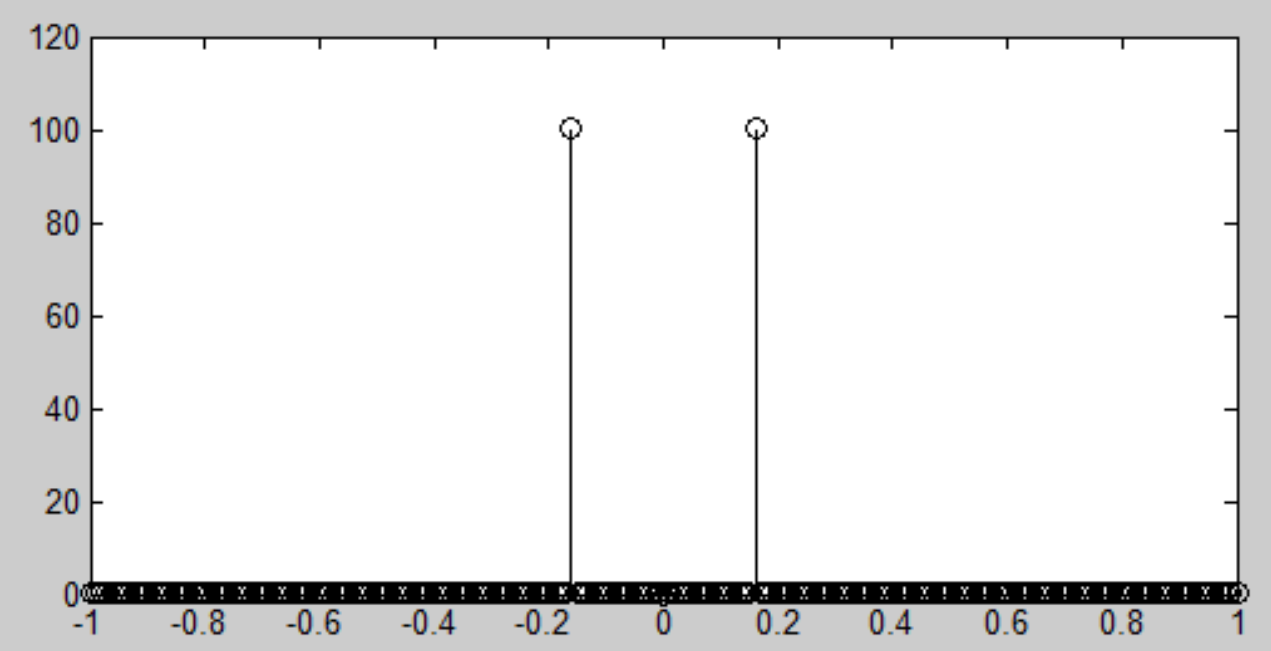
```
2 -     x=-10:0.1:10;
3 -     y=sin(3*x);
```



مثال ۲:

حال با تغییر فرکانس تابع، طیف فرکانسی را بررسی می کنیم.

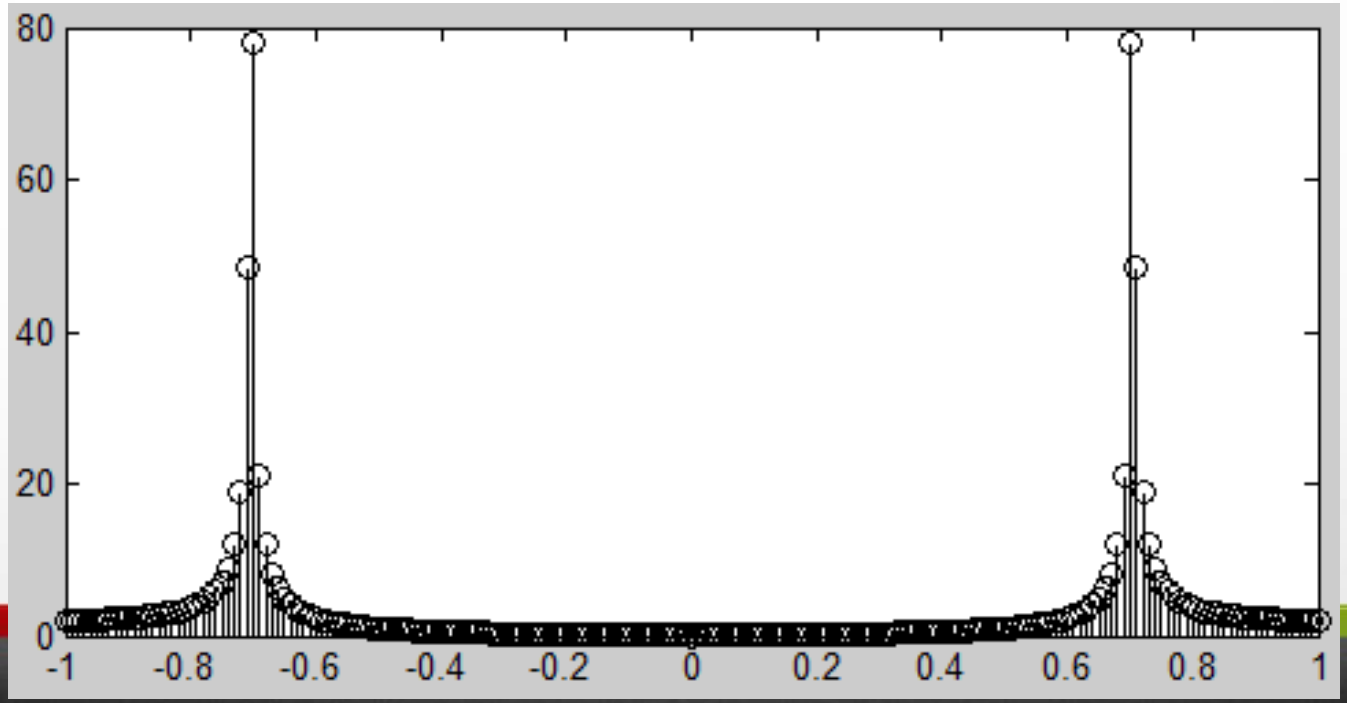
```
2 -     x=-10:0.1:10;
3 -     y=sin(5*x);
```



مثال ۲:

حال با تغییر فرکانس تابع، طیف فرکانسی را بررسی می کنیم.

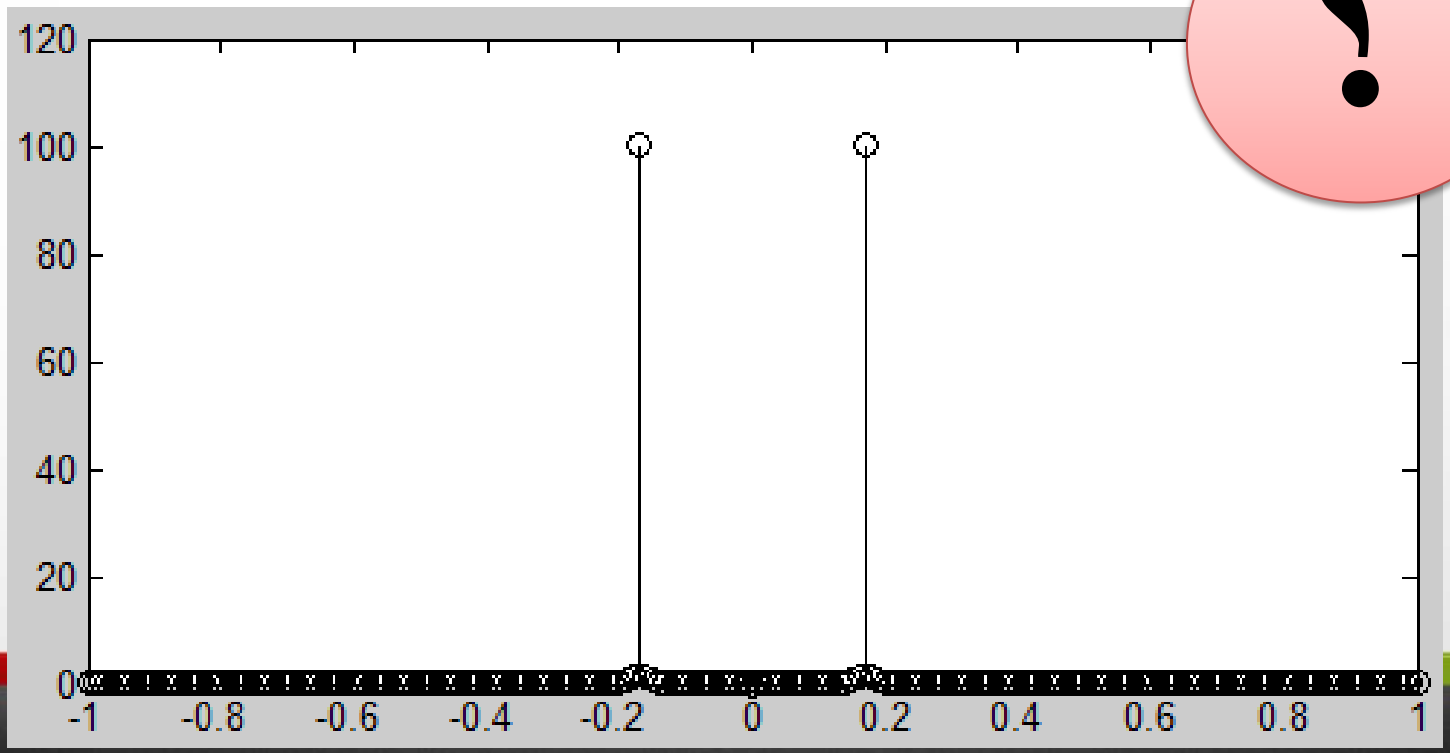
```
2 -     x=-10:0.1:10;
3 -     y=sin(22*x);
```



مثال ۲:

حال با تغییر فرکانس تابع، طیف فرکانسی را بررسی می کنیم.

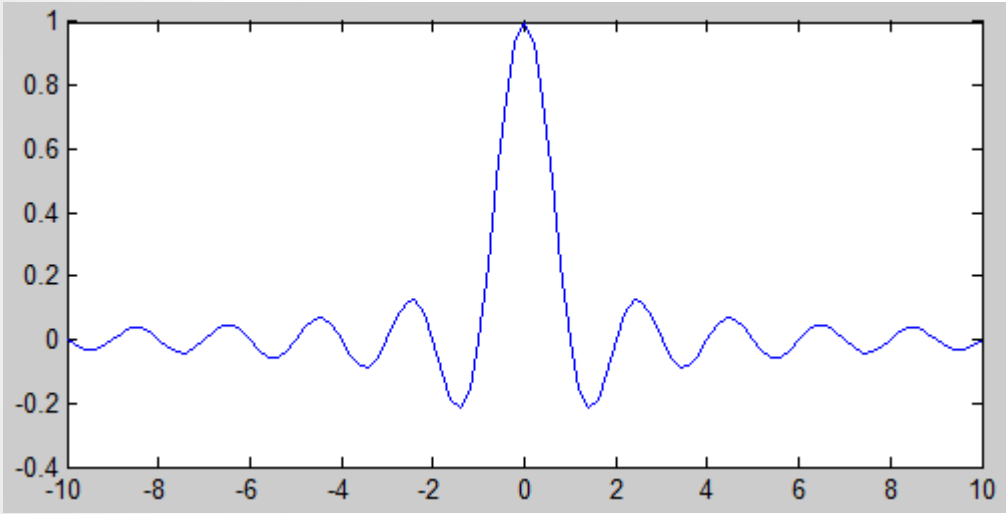
```
2 -     x=-10:0.1:10;
3 -     y=sin(1000*x);
```

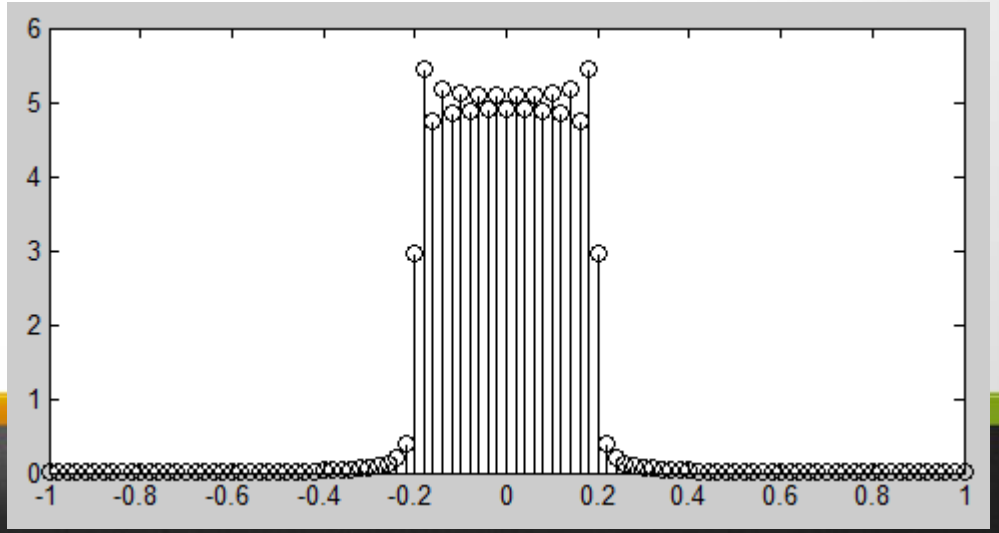


مثال ۳:

```
clear all;close all;clc;
x=-10:0.2:10;
y=sinc(x);
subplot(2,2,1);
plot(x,y)
y1=fft(y);
subplot(2,2,2);
stem(abs(y1),'r')
subplot(2,2,3);
Y2=fftshift(y1);
stem(abs(Y2),'g');
x2=linspace(1,-1,length(Y2));
subplot(2,2,4);
stem(x2,abs(Y2),'k');
```

مثال ۳:

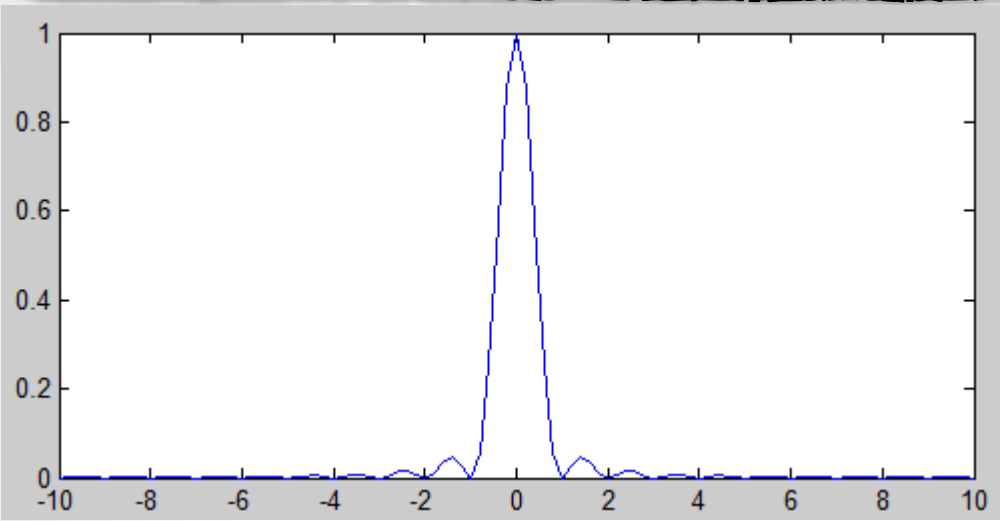

1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-10
-8
-6
-4
-2
0
2
4
6
8
10

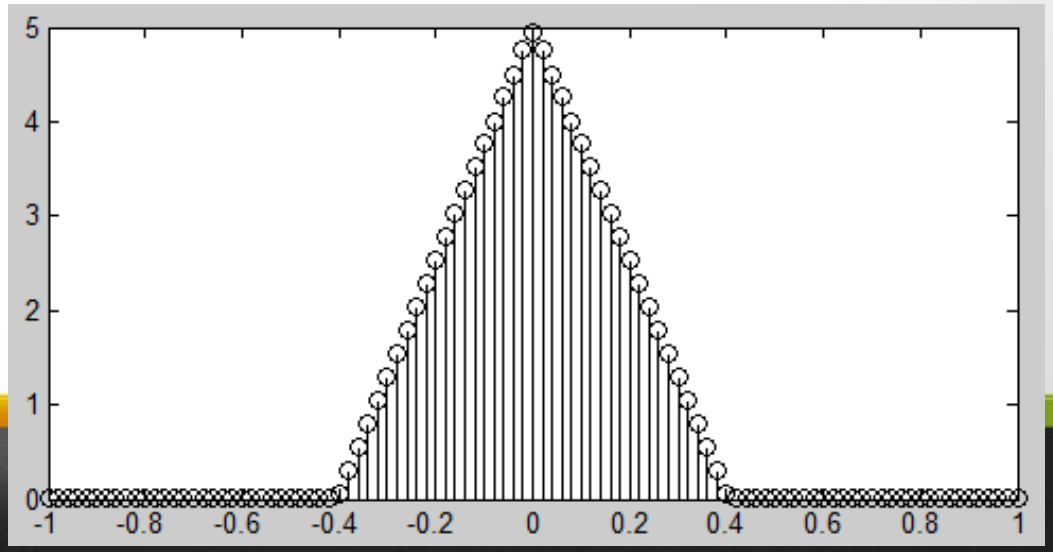

6
5
4
3
2
1
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

مثال ۴:

```
clear all;close all;clc;
x=-10:0.2:10;
y=(sinc(x)).^2;
subplot(2,2,1);
plot(x,y)
y1=fft(y);
subplot(2,2,2);
stem(abs(y1),'r')
subplot(2,2,3);
Y2=fftshift(y1);
stem(abs(Y2),'g');
x2=linspace(1,-1,length(Y2));
subplot(2,2,4);
stem(x2,abs(Y2),'k');
```

مثال ۴:

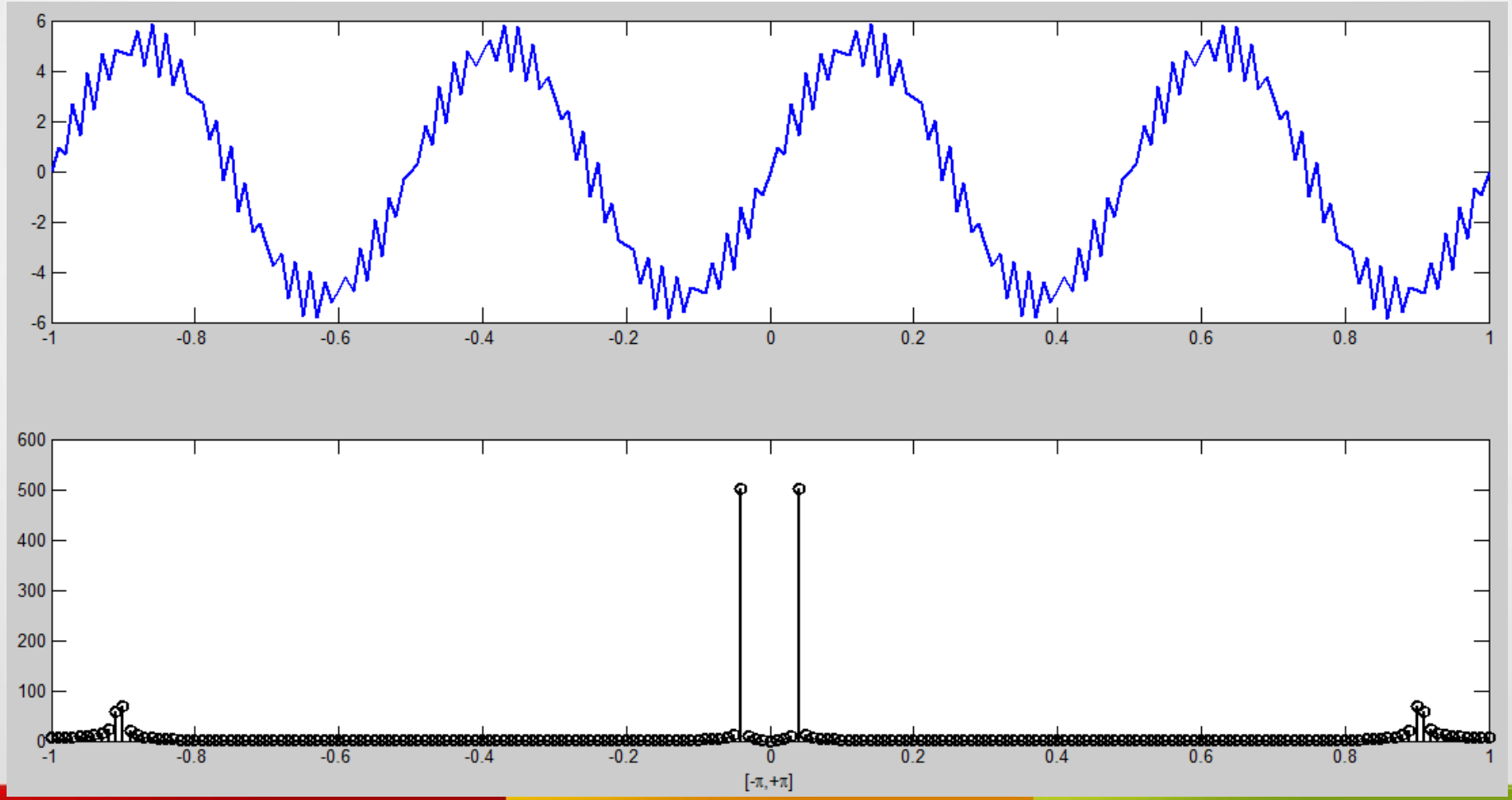




مثال ۵:

```
clear all;close all;clc;
x=-1:0.01:1;
y=sin(90*pi*x)+5*sin(4*pi*x);
subplot(2,1,1);
plot(x,y,'LineWidth',2)
y1=fft(y);
Y2=fftshift(y1);
x2=linspace(1,-1,length(Y2));
subplot(2,1,2);
stem(x2,abs(Y2),'k','LineWidth',2);
xlabel('[-\pi,+\pi]')
```

مثال ۵:



عکس تبدیل فوریه :

با استفاده از تبدیل فوریه می توان از حوزه زمان به حوزه فرکانس رفت ، و متقابلا با استفاده از عکس تبدیل فوریه می توان از حوزه فرکانس به حوزه زمان انتقال یافت.

دستور زیر ، عکس تبدیل فوریه سریع انجام می دهد.

ifft

Inverse fast Fourier transform

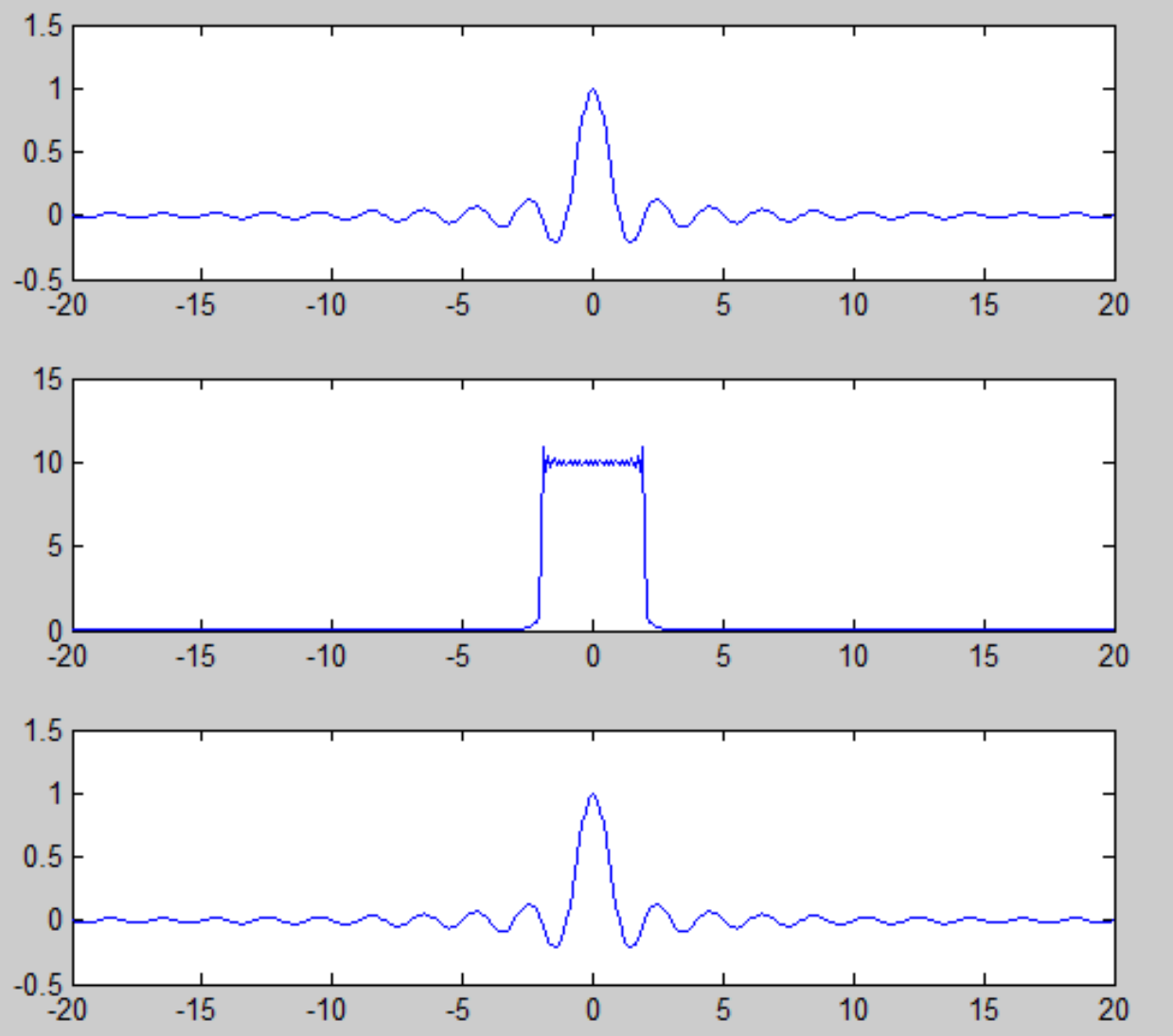
Syntax

```
y = ifft(X)
y = ifft(X,n)
y = ifft(X,[],dim)
y = ifft(X,n,dim)
y = ifft(..., 'symmetric')
y = ifft(..., 'nonsymmetric')
```

مثال ۶:

```
close all;clear all;clc
x=-20:0.1:20;
y=sinc(x);
subplot(3,1,1)
plot(x,y,'LineWidth',3)
Y=fft(y);
subplot(3,1,2)
Y2=fftshift(Y);
plot(x,abs(Y2),'LineWidth',3)
Y3=ifft(Y);
subplot(3,1,3)
plot(x,Y3,'LineWidth',3)
```

مثال ۶:

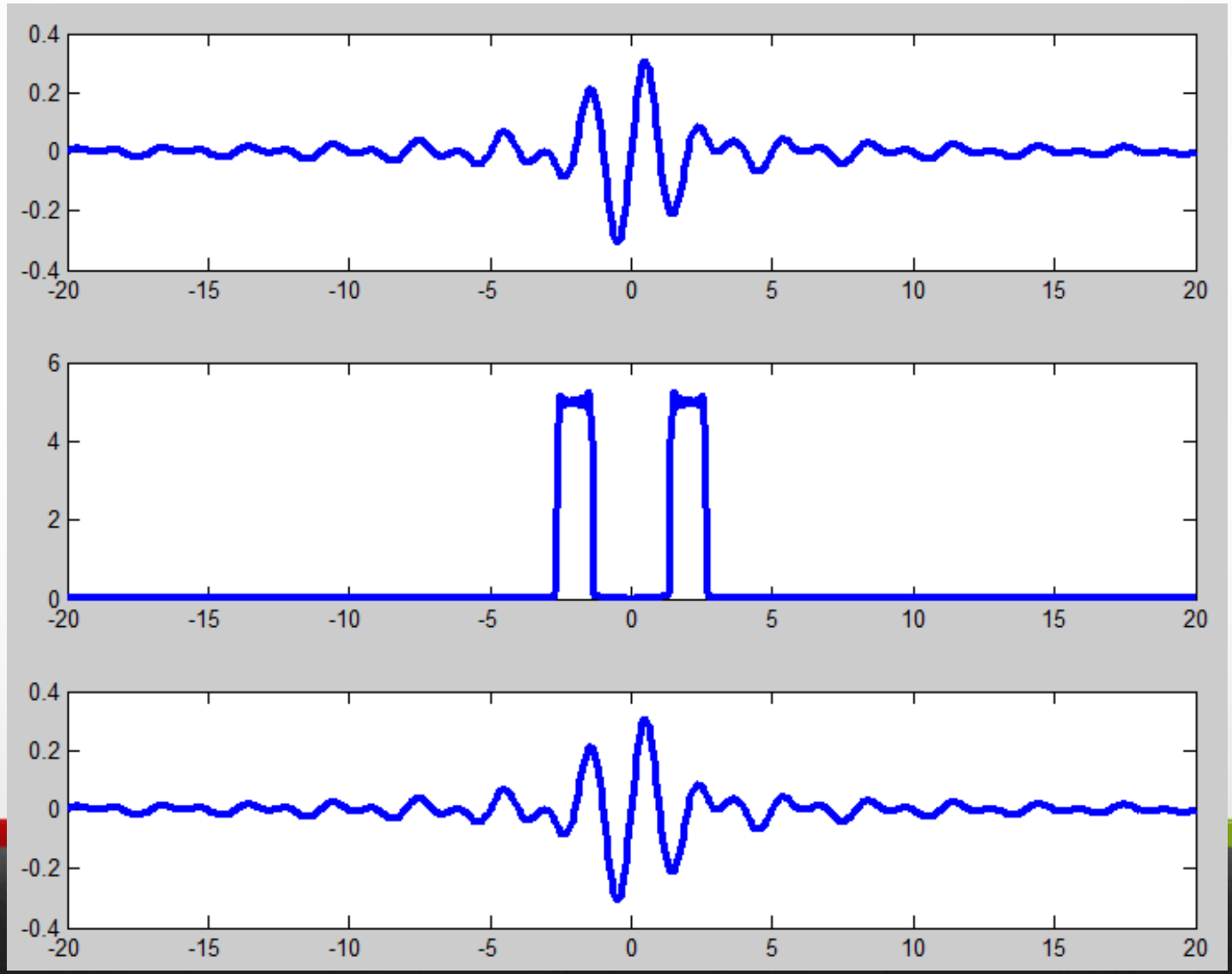

1.5
1
0.5
0
-0.5
-20 -15 -10 -5 0 5 10 15 20
15
10
5
0
-20 -15 -10 -5 0 5 10 15 20
1.5
1
0.5
0
-0.5
-20 -15 -10 -5 0 5 10 15 20

مثال ۷:

```
close all;clear all;clc
x=-20:0.1:20;
y=sinc(x).* sin(x);
subplot(3,1,1)
plot(x,y,'LineWidth',3)
Y=fft(y);
subplot(3,1,2)
Y2=fftshift(Y);
plot(x,abs(Y2),'LineWidth',3)
Y3=ifft(Y);
subplot(3,1,3)
plot(x,Y3,'LineWidth',3)

```

مثال ۷:

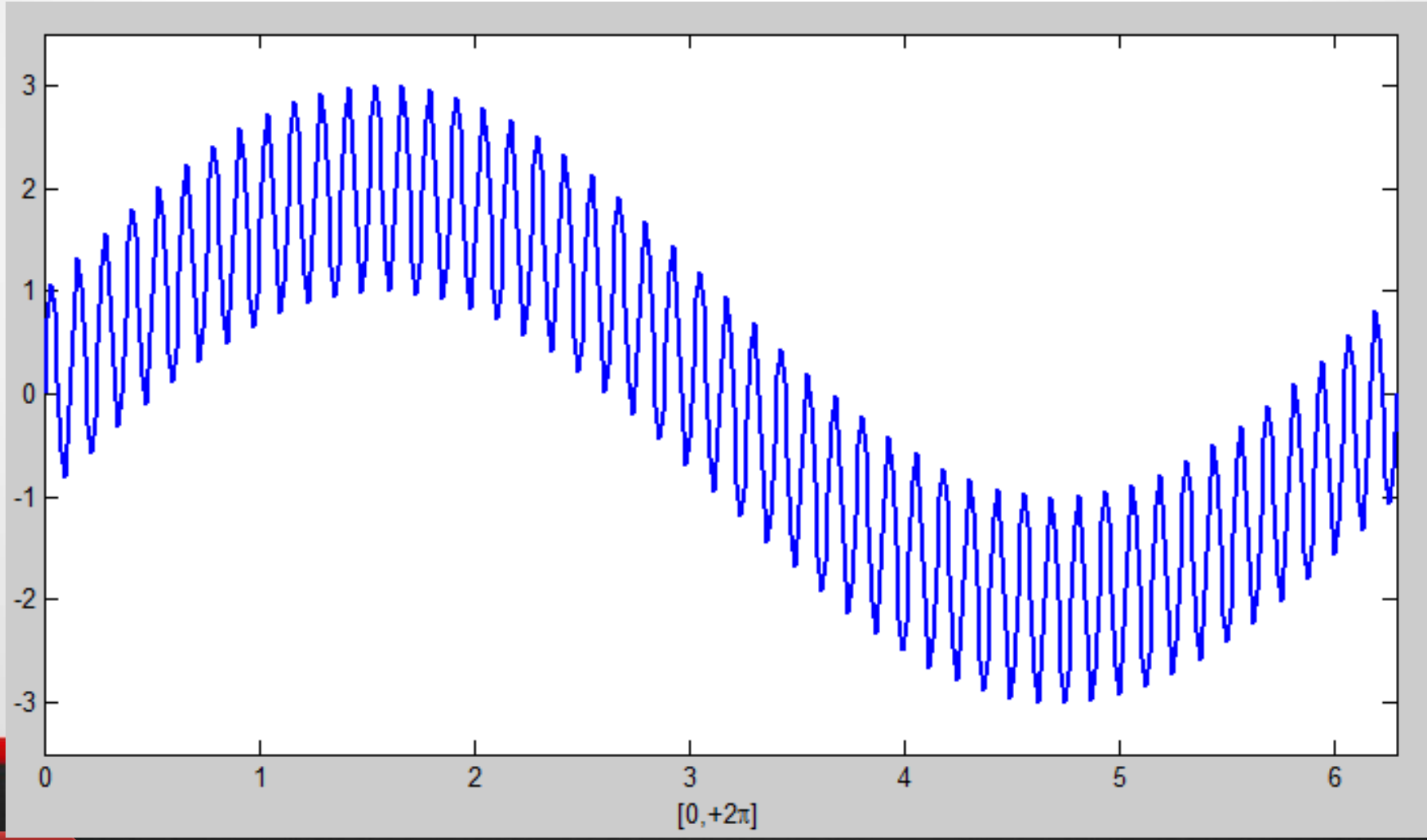

0.4
0.2
0
-0.2
-0.4
-20
-15
-10
-5
0
5
10
15
20
6
4
2
0
-20
-15
-10
-5
0
5
10
15
20
0.4
0.2
0
-0.2
-0.4
-20
-15
-10
-5
0
5
10
15
20

حذف نویز :

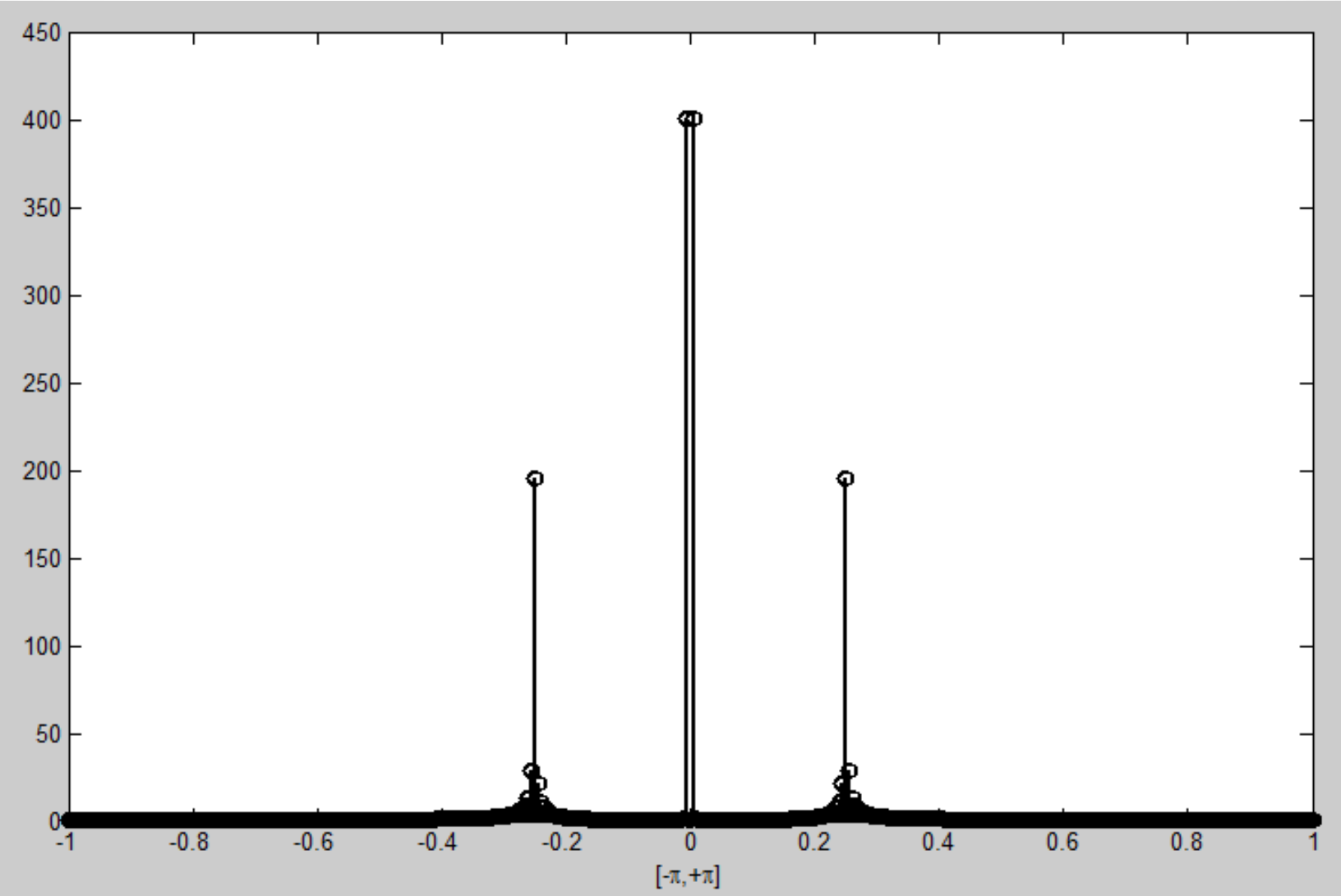
با توجه به توضیحات ارائه شده، می توان برای تحلیل رفتار یک نویز از حوزه فرکانس استفاده نمود.

برای مثال فرض کنید که داده اصلی ما یک موج سینوسی است که یک موج سینوسی با فرکانس بالاتر به صورت ناخواسته(نویزی شکل) بر روی داده اصلی ما اثر گذاشته است. حال خواهان حذف و از بین بردن آن هستیم.

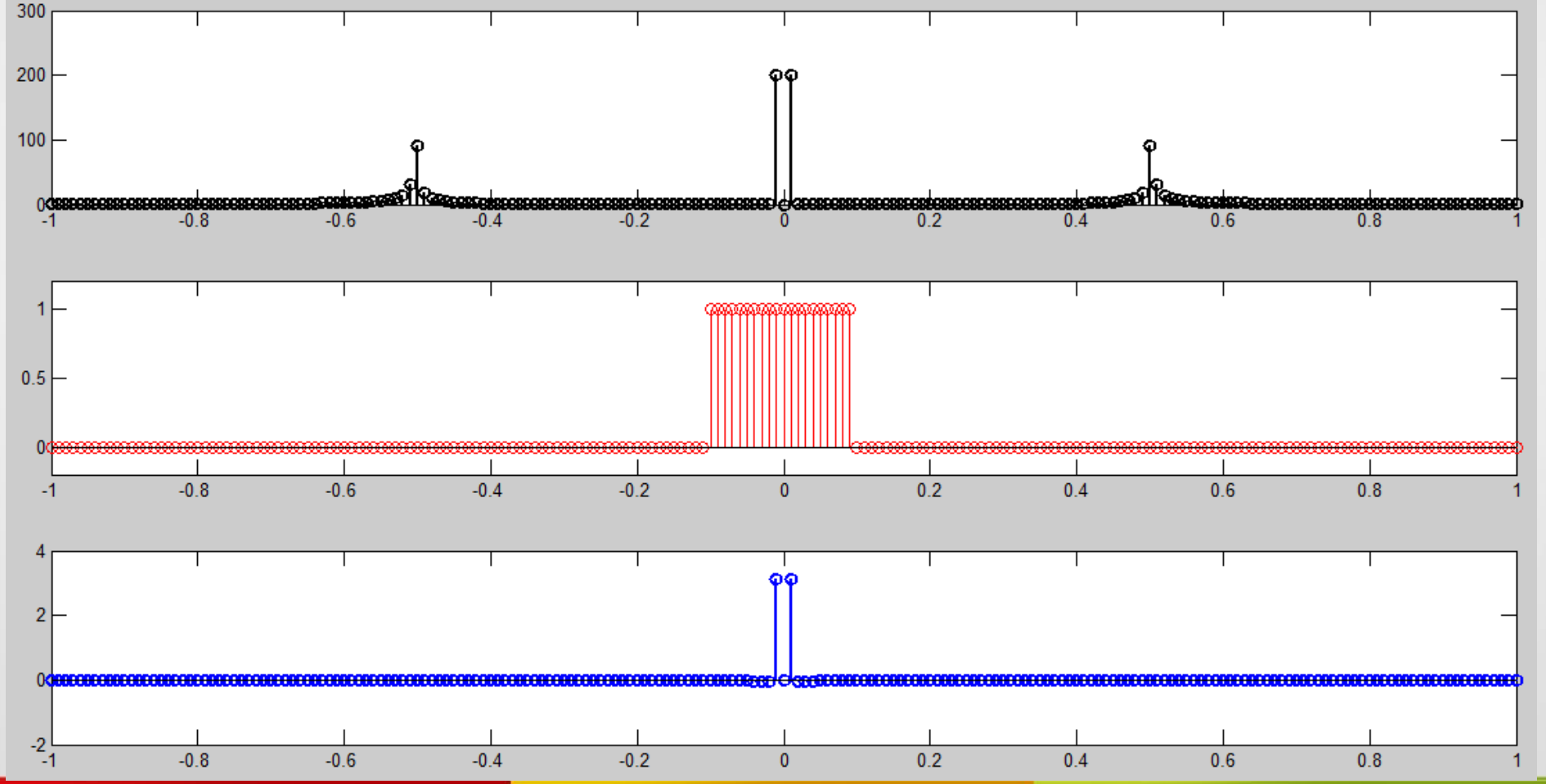
مثال ۸ :



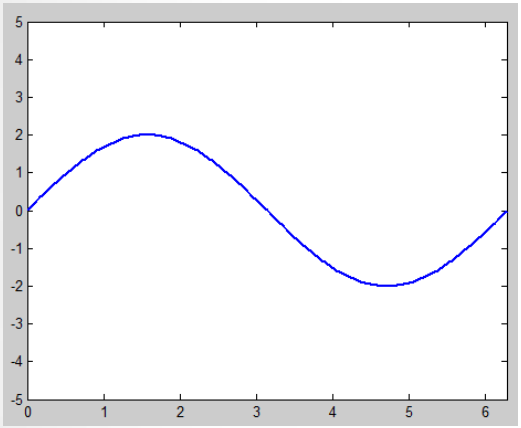
مثال ۸ :

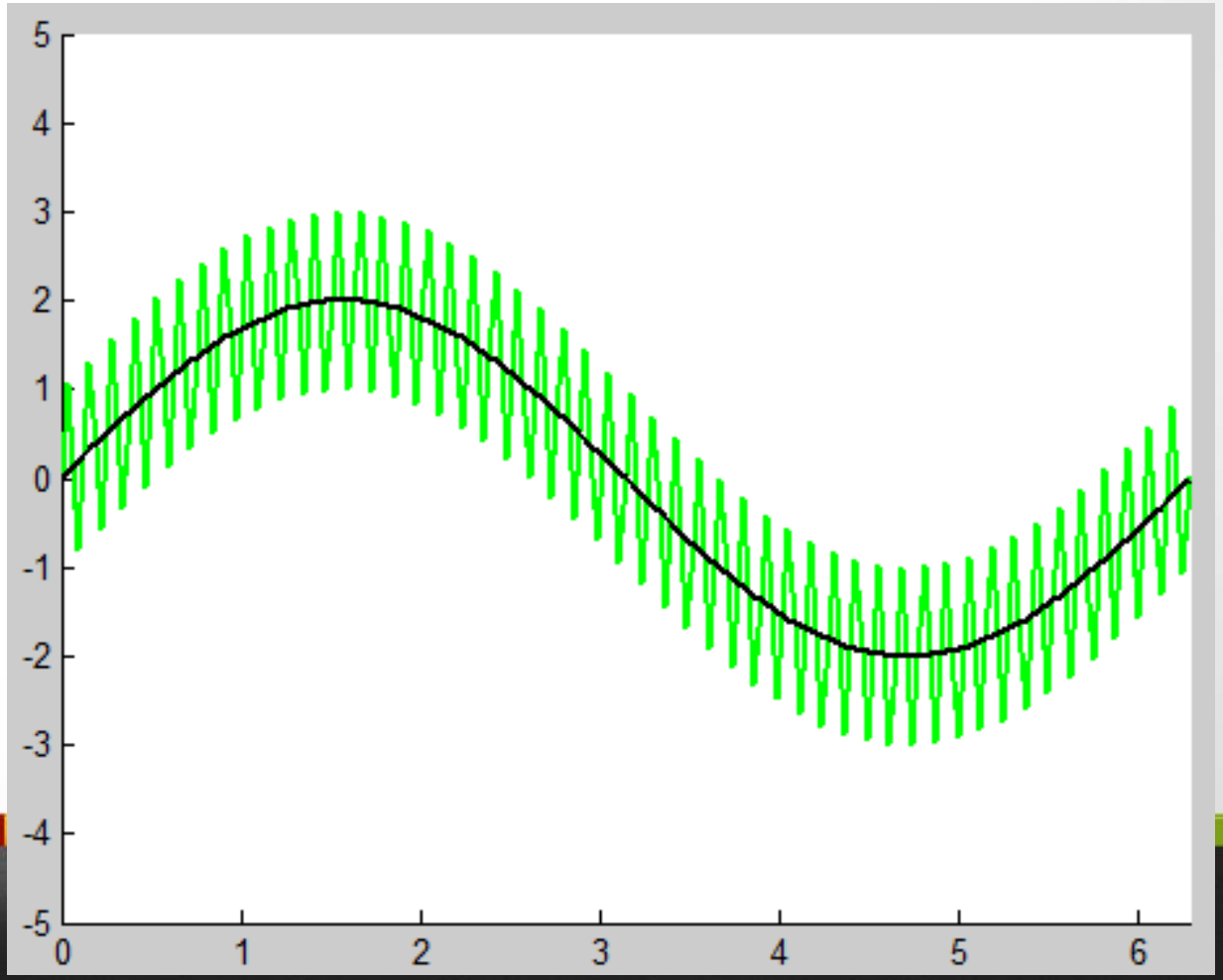


مثال ۸ :



مثال ۸ :





مثال ۸ :

```
close all;clear all;clc;

x=0:pi/100:2*pi;
y=2*sin(x)+sin(50*x);

figure(1)
plot(x,y,'LineWidth',2);
xlim([0 2*pi]); ylim([-5 5])

y1=fft(y);
y2=fftshift(y1);
y3=abs(y2);
x2=linspace(-1,+1,length(y2));

figure(2)
stem(x2,y3,'k','LineWidth',2);
xlabel('[-\pi,+\pi]')
```

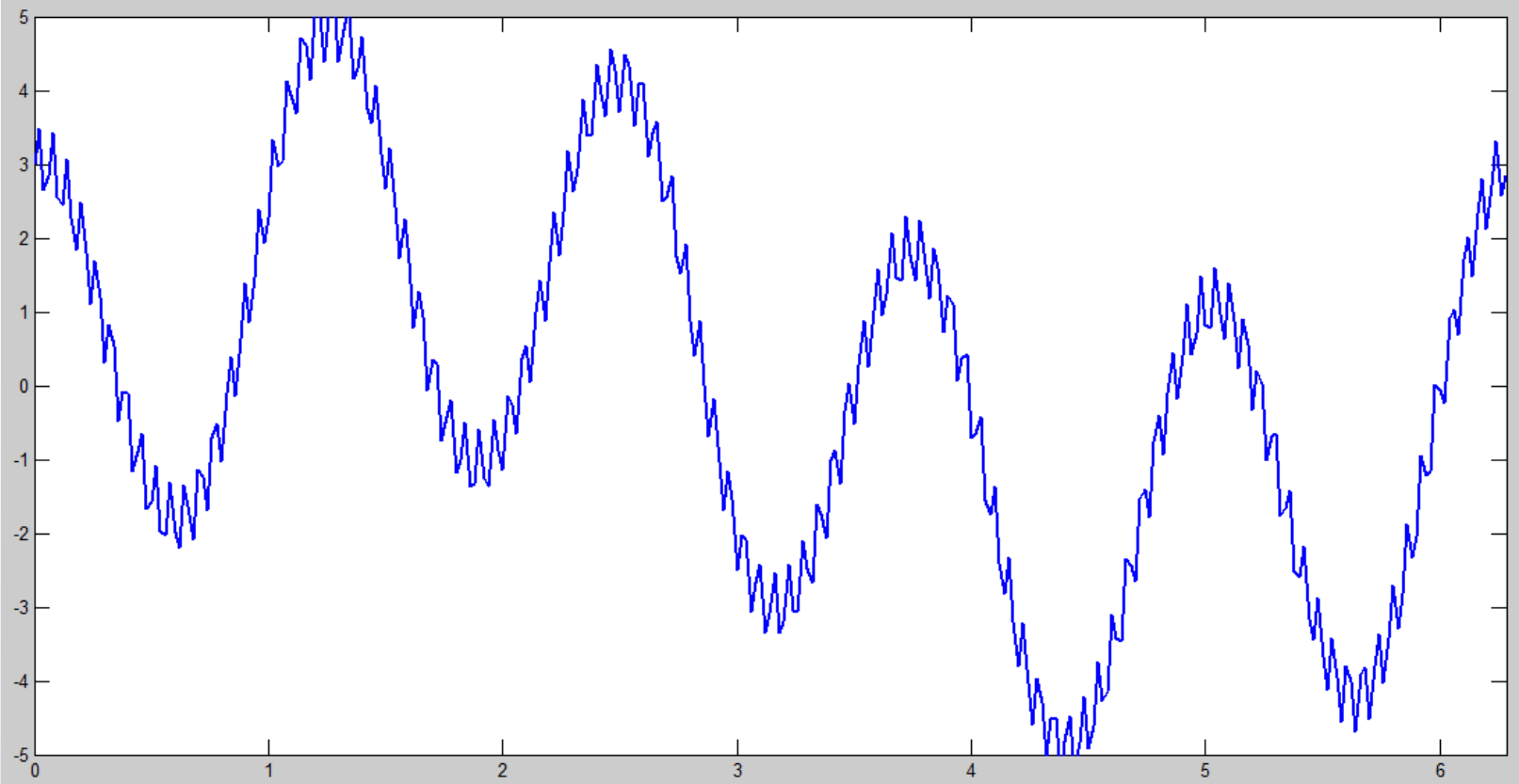
مثال ۸ :

```
18
19 -    figure(3)
20 -    subplot(3,1,1); stem(x2,y3,'k','LineWidth',2);
21
22 -    H=zeros(1,length(x));
23 -    for i=1:length(x)
24 -        if x2(i)>-0.1 & x2(i)< 0.1
25 -            H(i)=1;
26 -        end
27 -    end
28
29 -    subplot(3,1,2); stem(x2,(H),'r');
30 -    ylim([-0.2 1.2]);
31
32 -    H2=ifftshift(H);
33 -    z=y1.*H2;
34 -    subplot(3,1,3);
35 -    stem(x2,fftshift(z),'LineWidth',2);
```

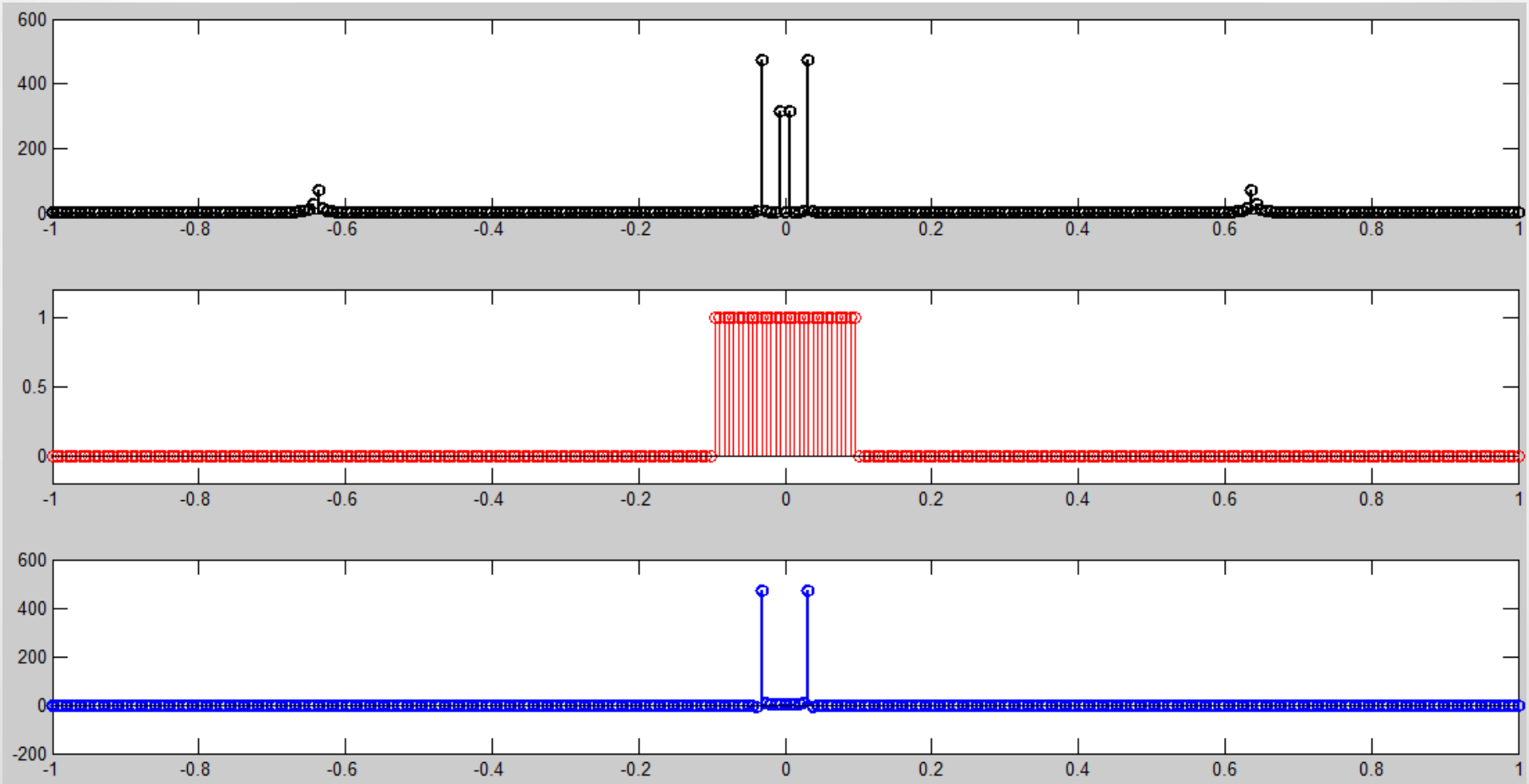
مثال ۸ :

```
36
37
38 -    figure(4)
39 -    z2=ifft(z);
40 -    plot(x,z2,'LineWidth',2)
41 -    xlim([0 2*pi]); ylim([-5 5])
42
43 -    figure(5)
44 -    hold on
45 -    plot(x,y,'g','LineWidth',2)
46 -    plot(x,z2,'k','LineWidth',2);
47 -    xlim([0 2*pi]); ylim([-5 5])
```

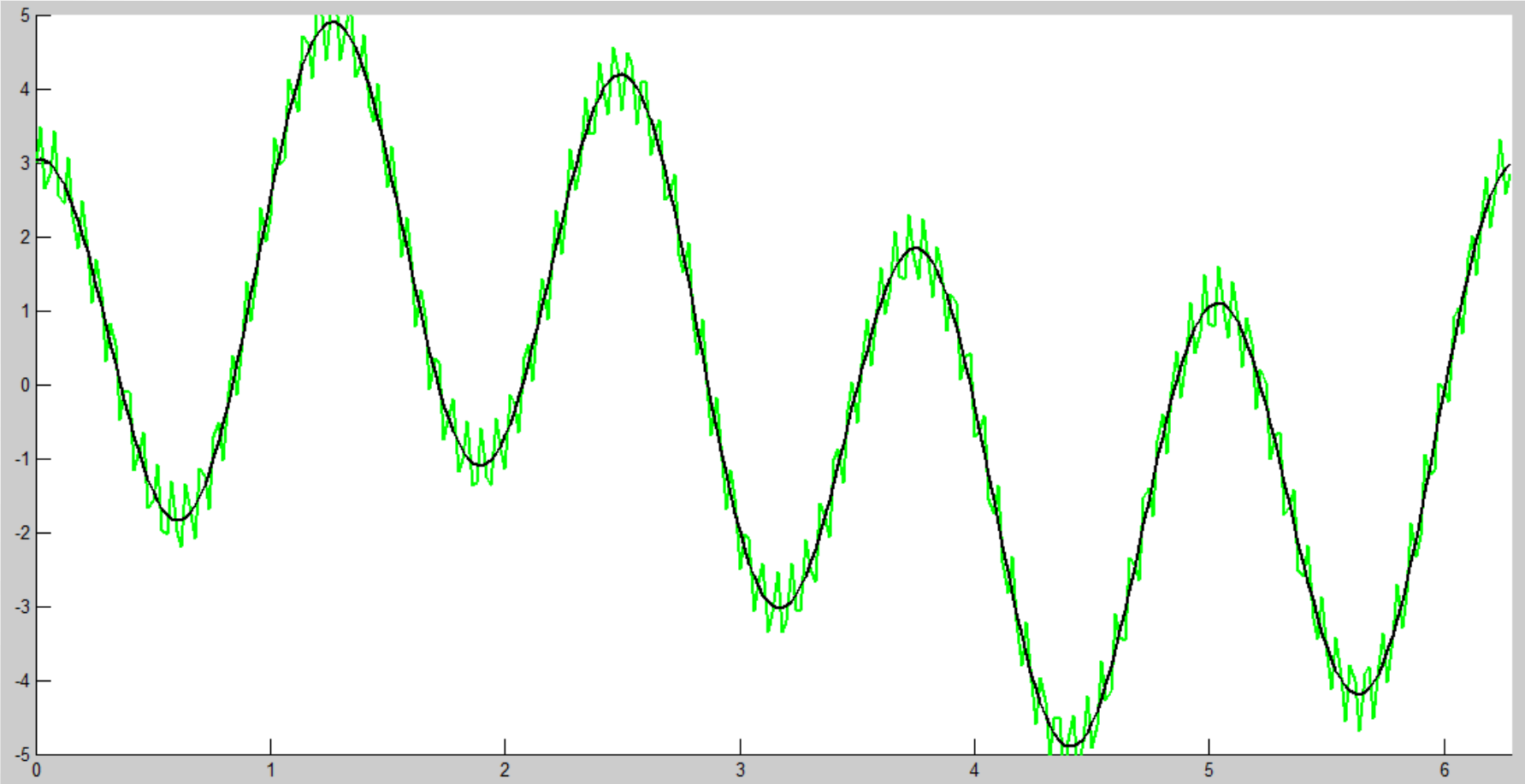
مثال ۹ :



مثال ۹ :



مثال ۹ :



مثال ۹ :

```
3 -     x=0:0.02:2*pi;
4 -     y=2*sin(x)+3*cos(5*x)+0.5*sin(100*x);
```

```
22 -    H=zeros(1,length(x));
23 -    for i=1:length(x)
24 -        if x2(i)>-0.1 & x2(i)< 0.1
25 -            H(i)=1;
26 -        end
27 -    end
```

نویز سفید :

همان طور که قبلا نیز به آن اشاره گردید، طیف فرکانسی نویز سفید تمام فرکانس ها را در بر می گیرد. برای شبیه سازی آن می توان از کد زیر استفاده نمود.

wgn

Generate white Gaussian noise

Syntax

```
y = wgn(m,n,p)
y = wgn(m,n,p,imp)
y = wgn(m,n,p,imp,state)
y = wgn(...,powertype)
y = wgn(...,outputtype)
```

awgn

Add white Gaussian noise to signal

Syntax

```
y = awgn(x,snr)
y = awgn(x,snr,sigpower)
y = awgn(x,snr,'measured')
y = awgn(x,snr,sigpower,s)
y = awgn(x,snr,'measured',state)
y = awgn(...,powertype)
```

متلب ...

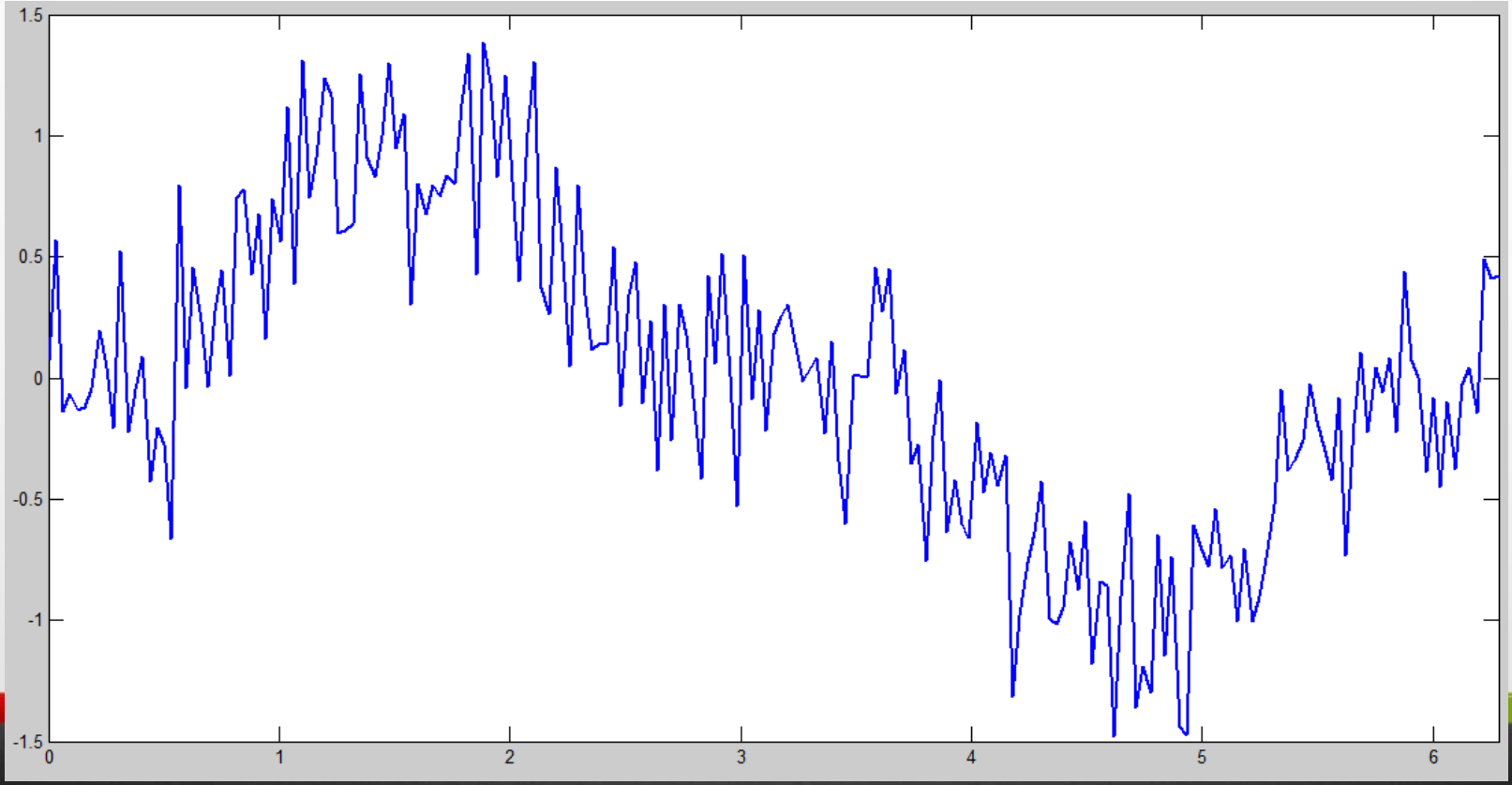
برای شبیه سازی به روش های مخختلفی می توان نویز سفید را به سیگنال وارد نمود. برای مثال هر دو روش زیر برای این کار مورد استفاده قرار می گیرند.

```
1 -     close all;clear all;clc;
2 -     x=0:pi/100:2*pi;
3 -     RND=randn(1,length(x));
4 -     y=(sin(x)).^3+0.3*RND;
```

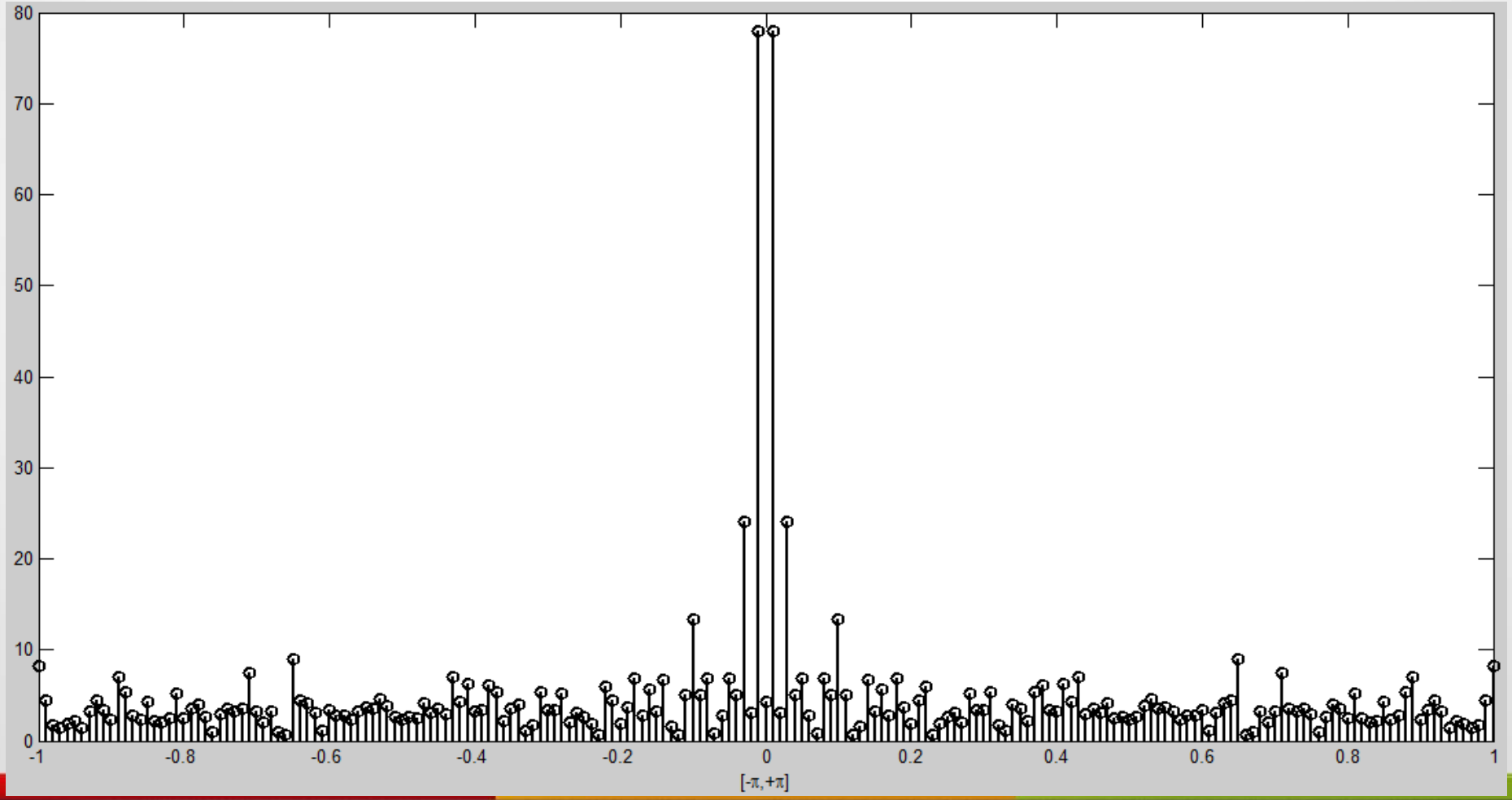
```
1 -     close all;clear all;clc;
2 -     x=0:pi/100:2*pi;
3 -     y=(sin(x)).^3+0.3*wgn(1,length(x),0);
```

مثال ۱۰ :

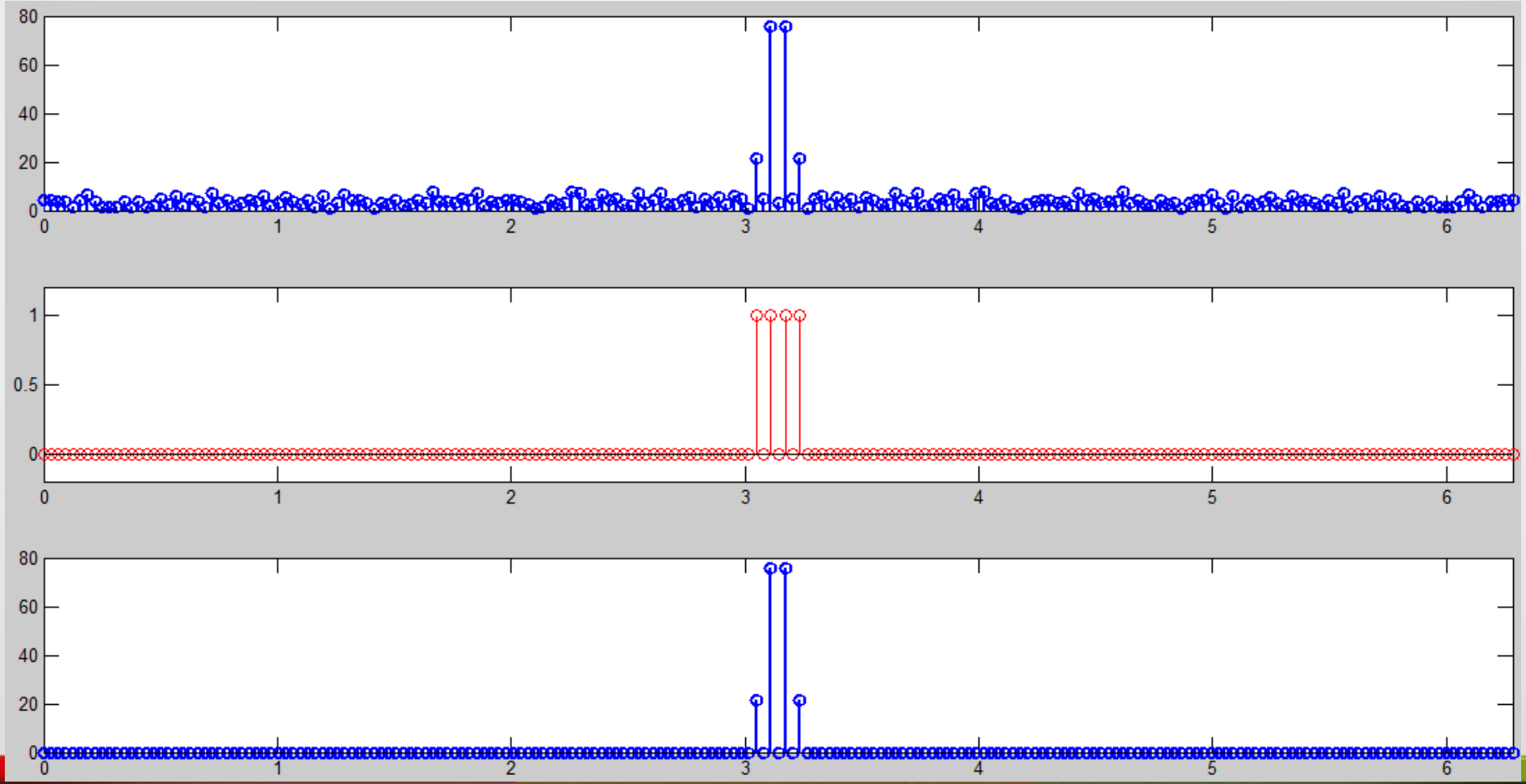
سیگنال میخواهیم .شود می گرفته نظر در سفید نویز یک با $y = (\sin(x))^3$
اثر نویز را بر روی سیگنال اصلی حذف نماییم.



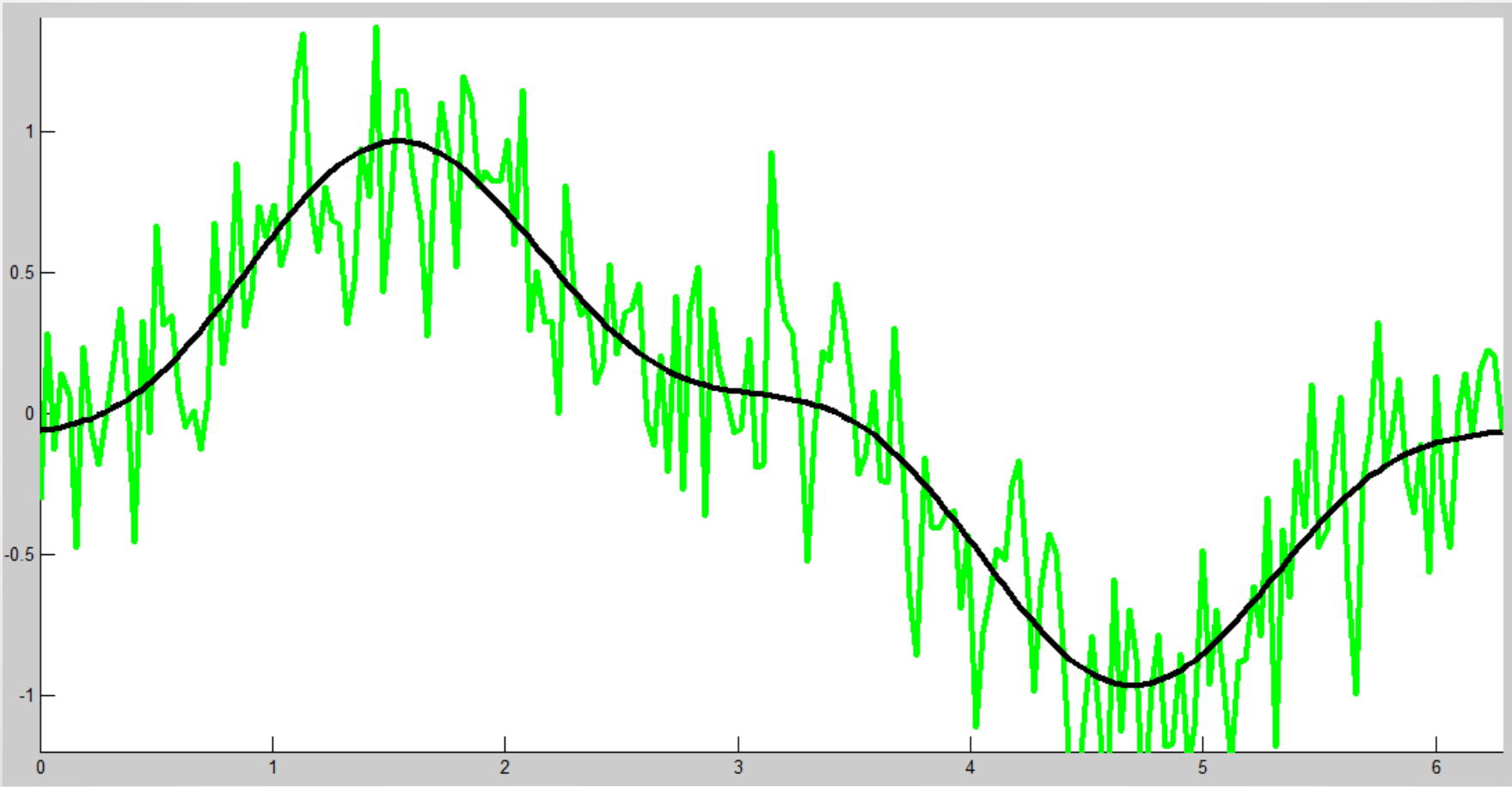
مثال ۱۰ :



مثال ۱۰ :



مثال ۱۰ :



مثال ۱۰ :

```
close all;clear all;clc;
x=0:pi/100:2*pi;
RND=randn(1,length(x));
y=(sin(x)).^3+0.3*RND;

% y=(sin(x)).^3+0.2*wgn(1,length(x),0);

plot(x,y,'LineWidth',2);
xlim([0 2*pi]);

%%
ft_y=(fft(y));
figure
stem(x,abs(fftshift(ft_y)),'LineWidth',2)
xlim([0 2*pi])
```

مثال ۱۰ :

```
16      %%
17 -    figure
18 -    subplot(3,1,1); stem(x,abs(fftshift(ft_y)),'LineWidth',2)
19 -    xlim([0 2*pi]);
20
21 -    H=zeros(1,length(x));
22 -    Mx=max(abs(fftshift(ft_y)));
23 -    Mn=min(abs(fftshift(ft_y)));
24 -    Md=(Mx-Mn)*0.2;
25 -    Md2=mean(abs(fftshift(ft_y)));
26 -    fts_y=abs(fftshift(ft_y));
27 -    for i=1:length(x)
28 -        if fts_y(i)>2*Md2
29 -            H(i)=1;
30 -        end
31 -    end
```

مثال ۱۰ :

```
32      %%
33 -    subplot(3,1,2); stem(x,(H),'r');
34 -    ylim([-0.2 1.2]);xlim([0 2*pi]);
35 -    H2=ifftshift(H);
36 -    Z=ft_y.*H2;
37 -    subplot(3,1,3); stem(x,abs(fftshift(Z)),'LineWidth',2);
38 -    xlim([0 2*pi]);
39      %%
40 -    figure
41 -    Z2=ifft(Z);
42 -    plot(x,Z2,'LineWidth',2)
43 -    xlim([0 2*pi]); ylim([-5 5])
44      %%
45 -    figure
46 -    hold on
47 -    plot(x,y,'g','LineWidth',3)
48 -    plot(x,Z2,'k','LineWidth',3);
49 -    xlim([0 2*pi]); ylim([-1.2 1.4])
```

برخی منابع مورد استفاده :

1-Digital Signal Processing Using MATLAB

۲- حذف بهینه نویز صنعتی با استفاده از فیلترهای تک‌کاناله ، سید فرید موسوی پور ، دانشگاه شاهد

۳- راهنمای نرم افزار متلب

۴- منابع مختلف موجود در اینترنت

۵- و ...