به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

تمرین کامپیوتری ۱

سیگنال ها و سیستم ها

دکتر اخوان

عرفان پناهی ۸۱۰۱۹۸۳۶۹

بهار ۱۴۰۰-۱۳۹۹

فهرست:

سوال ۱ : دترمینان ماتریس دو بعدی

***** m-file مربوط به این قسمت با نام p1.m پیوست شده است.**

- برای مثال یک ماتریس ۴ در ۴ به صورت زیر تعریف میکنیم و دترمینان آنرا به کمک تابع بالا محاسبه می‌نماییم:

```
>> A = [ 3 5 7 2;
2 5 3 9;
3 1 6 4;
1 5 8 0];
>> detA = p1(A)

detA =

  -518
```

- برای اینکه از درستی نتیجه اطمینان حاصل نماییم از دستور det متلب برای محاسبه دترمینان نیز استفاده مینماییم:

```
>> det(A)

ans =

 -518.0000
```

- همچنین برای اینکه تابع نوشته شده به ازای ورودی نامعتبر مقاوم باشد، شرطی برقرار ساختیم که به ازای ورودی ماتریس غیر مربعی عدم وجود دترمینان را نتیجه دهد:

```
B =

     1     2     3
     4     5     6

>> p1(B)

ans =

    "Matrix must be square."
```

سوال ۲ : معکوس ماتریس دو بعدی

*** m-file مربوط به این قسمت با نام p2.m پیوست شده است.

- برای مثال معکوس ماتریس ۴ در ۴ (که در قسمت اول هم دترمینان آنرا محاسبه نمودیم) را با این تابع نشان میدهیم:

```
>>  A = [ 3 5 7 2;
2 5 3 9;
3 1 6 4;
1 5 8 0];
>> invA = p2(A)

invA =

    0.5753   -0.1236   -0.0097   -0.4498
    0.2124    0.0618   -0.2452   -0.0251
   -0.2046   -0.0232    0.1544    0.1969
   -0.1776    0.1120    0.0869    0.0483
```

- برای اینکه از درستی نتیجه اطمینان حاصل نماییم از دستور inv متلب برای محاسبه دترمینان نیز استفاده مینماییم:

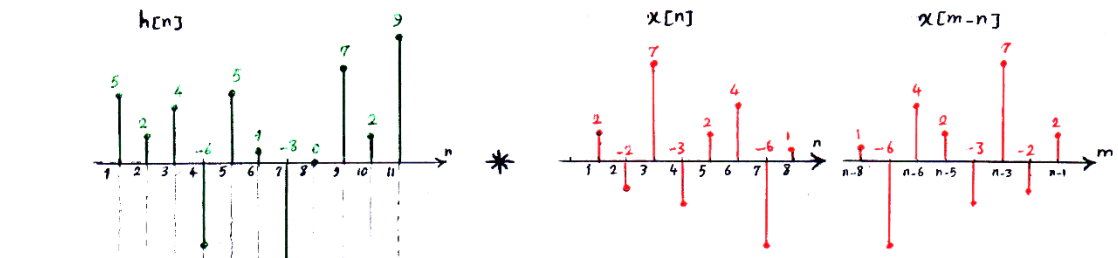
```
>> invA = inv(A)

invA =

    0.5753   -0.1236   -0.0097   -0.4498
    0.2124    0.0618   -0.2452   -0.0251
   -0.2046   -0.0232    0.1544    0.1969
   -0.1776    0.1120    0.0869    0.0483
```

- همچنین از یک ماتریس نمونه که دترمینان صفر دارد نیز استفاده میکنیم تا عدم وارون پذیری زا در تابعمان نشان بدهیم:

```
>> B=[7 4;
14 8];
>> detB = p1(B)

detB =

     0

>> invB = p2(B)

invB =

    "Matrix is not invertible."
```

سوال ۳ : دستور conv متلب

ابتدا به صورت دستی خروجی این سیستم را برای سیگنال $x[n]$ محاسبه مینماییم. برای این کار از حالت زیر در قضیه کانولوشن استفاده می‌کنیم: ($y[n] = \sum_{n=-\infty}^{+\infty} h[m]\, x[n-m]$)



$n<0 \rightarrow y[n] = 0$

$n=1 \rightarrow y[1] = 5\times 2 = 10$

$n=2 \rightarrow y[2] = 5\times(-2) + 2\times 2 = -6$

$n=3 \rightarrow y[3] = 5\times 7 + 2(-2) + 4\times 2 = 39$

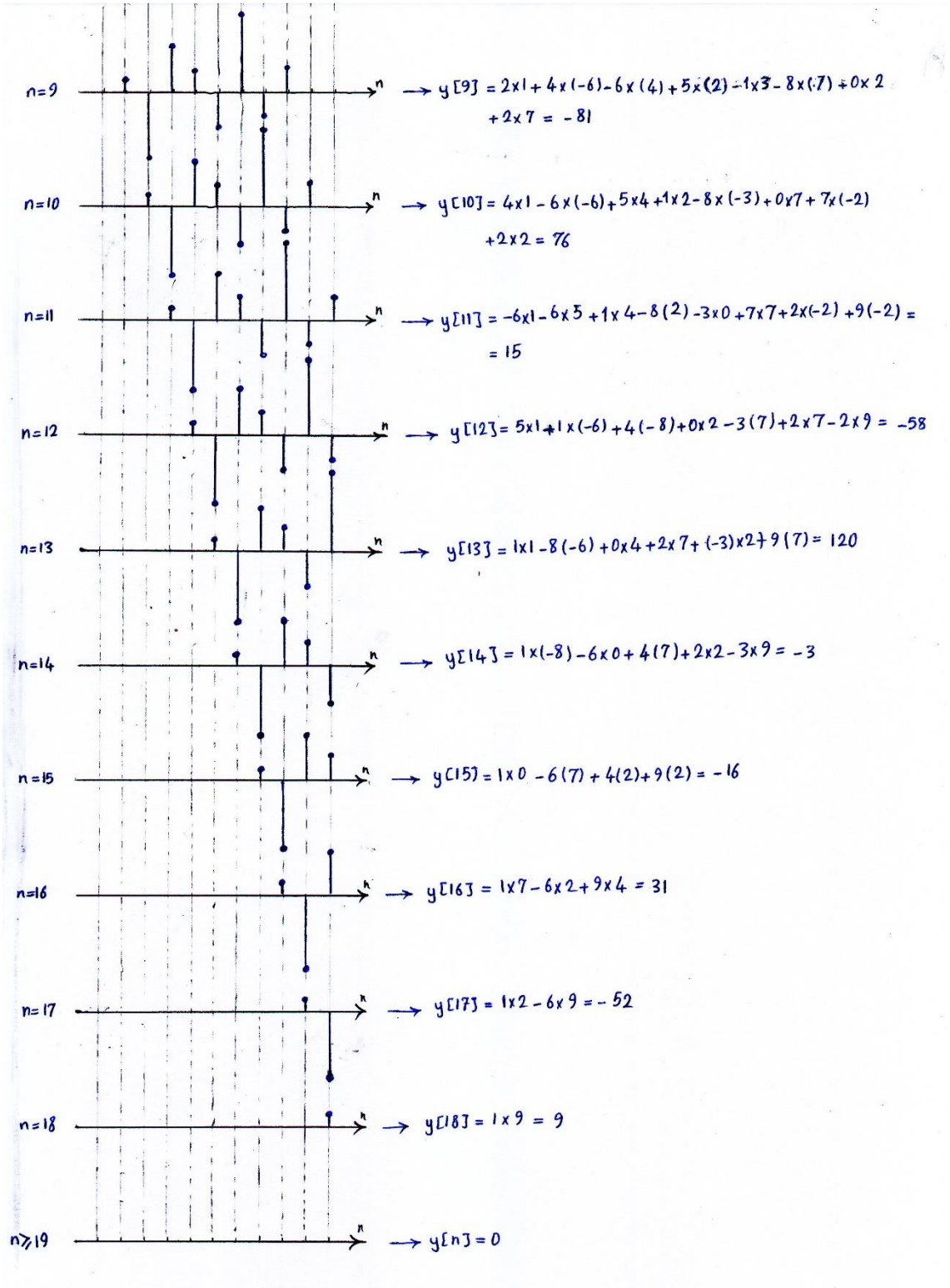
$n=4 \rightarrow y[4] = -3\times 5 + 7\times 2 + 4\times(-2) - 6\times 2 = -21$

$n=5 \rightarrow y[5] = 5\times 2 - 3\times 2 + 4\times 7 - 6(-2) + 5\times 2 = 54$

$n=6 \rightarrow y[6] = 4\times 5 + 2\times 2 - 3\times 4 + 7\times(-6) - 2\times 5 - 1\times 2 = -38$

$n=7 \rightarrow y[7] = -6\times 5 + 4\times 2 + 2\times 4 - 3(-6) + 5\times 7 + 1(-2) - 8\times 2 = 21$

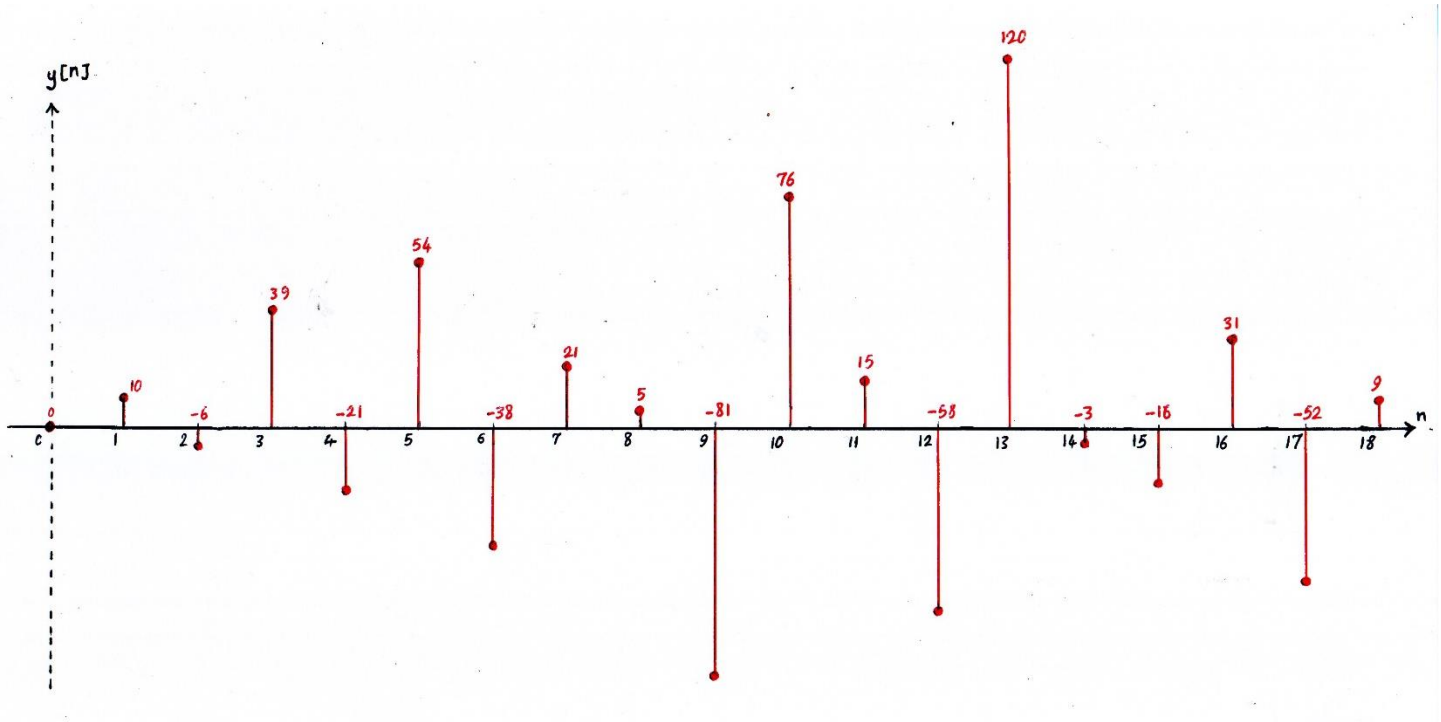
$n=8 \rightarrow y[8] = 5\times 1 - 6\times 2 + 4\times 4 - 6\times 2 - 3\times 5 + 7\times 1 + 8\times 2 + 2\times 0 = 5$



- در نتیجه سیگنال خروجی y[n] بصورت زیر است:

$$Y[n] = [10, \quad -6, \quad 39, \quad -21, \quad 54, \quad -38, \quad 21, \quad 5, \quad -81, \quad 76, \quad 15, \quad -58, \quad 120, \quad -3, \quad -16, \quad 31, \quad -52, \quad 9]$$

- و همچنین نتیجه سیگنال خروجی روی محور نیز بصورت زیر میباشد:



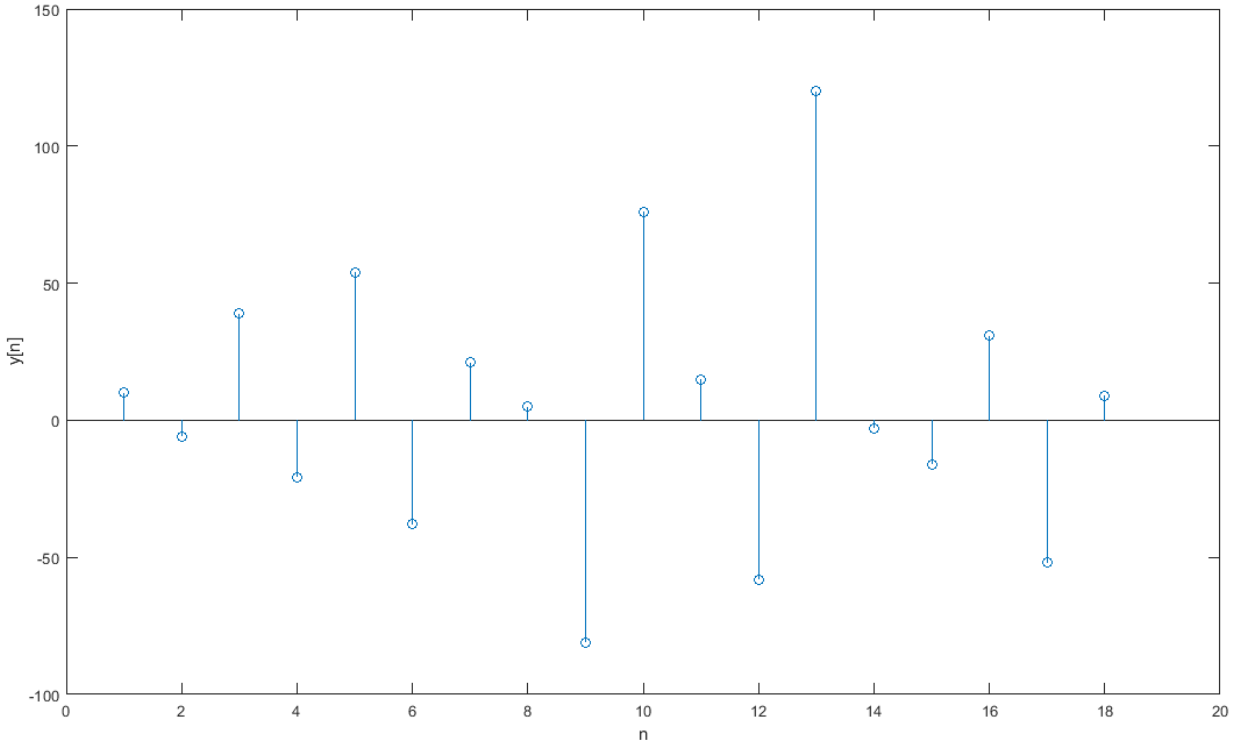
تصویر۲: نمودار خروجی محاسبه دستی سوال ۳

دستور $conv$ در متلب:

- حال با استفاده از دستور $conv$ در متلب و با وارد کردن دستورات زیر در محیط $command\ window$ خروجی این سیستم را مشاهده مینمایم:

```
>> x=[2 -2 7 -3 2 4 -6 1];
>> h=[5 2 4 -6 5 1 -8 0 7 2 9];
>> y = conv(x,h)

y =

  Columns 1 through 16

    10    -6    39   -21    54   -38    21     5   -81    76    15   -58
120    -3   -16    31

  Columns 17 through 18

   -52     9

>> stem(y);
>> xlabel('n');
>> ylabel('y[n]');
>> xlim([0 20]);
```

که تصویر نمودار خروجی (تصویر۲) به صورت زیر است:



تصویر۲: نمودار خروجی دستور $conv$ سوال ۳

سوال ۴ : سیگنال صوتی

***** m-file مربوط به این قسمت با نام p4a.m پیوست شده است.**

- با استفاده از دستور زیر فایل صوتی ذخیره شده در آدرس مورد نظر را وارد متلب میکنیم:

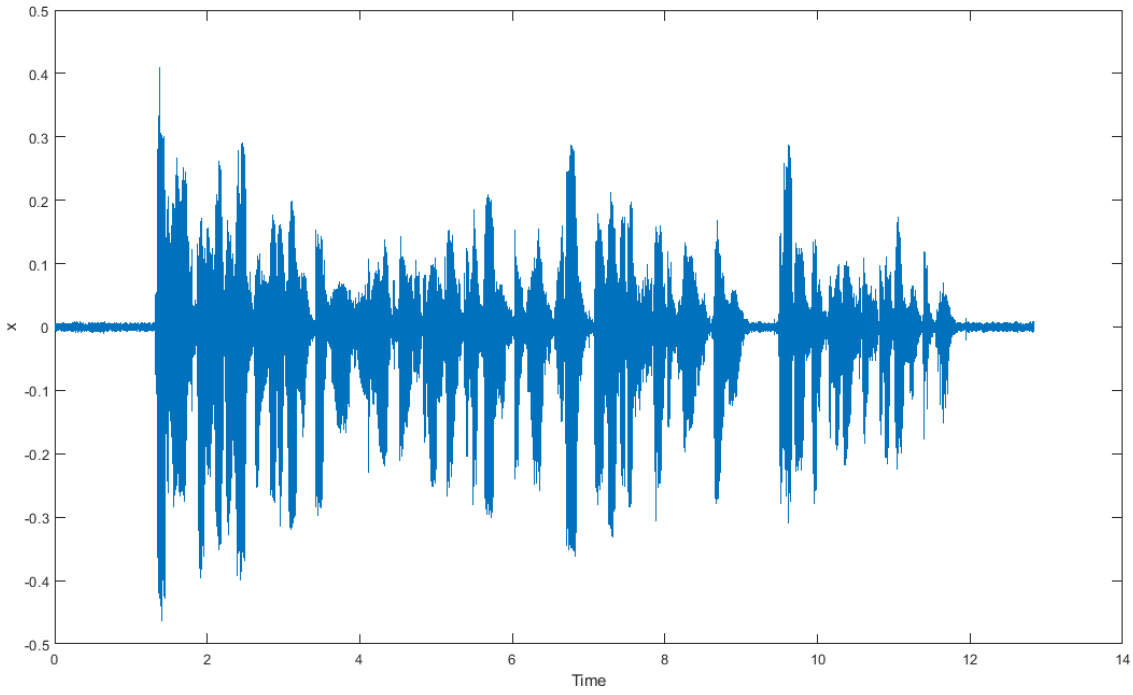
```
filename='E:\Desktop\SS_CA1\MyTest.wav';
[x,fs]=audioread(filename);
```

x برداری است که سیگنال مورد نظر را در آن ذخیره میکنیم و fs فرکانس نمونه برداری سیگنال است.

بخش ۴-۱ : رسم سیگنال ورودی

- برای اینکه بخواهیم سیگنال را روس محور زمان نشان دهیم باید بردار زمان را از به کمک تعداد نمونه برداری ها و فرکانس نمونه برداری بدست آوریم. از دستورهای زیر برای بدست آوردن بردار زمان استفاده میکنیم: (خروجی در تصویر ۳)

```
t=linspace(0,numel(x)/fs,numel(x));
plot(t,x);
xlabel('Time');
ylabel('x');
```



تصویر۳: رسم سیگنال صوتی ورودی سوال ۴-۱

- همچنین با اسفاده از دستور زیر میتوانیم سیگنال x را ذخیره کنیم:

```
audiowrite('x.wav',x,fs);
```

بخش ۴-۲ : اثبات LTI بودن و پاسخ ضربه

- ابتدا ثابت میکنیم این سیستم یک سیستم LTI است:

۱. تغییر ناپذیری با زمان (TI):

$$y[n] = x[n] + ax[n - n_0]$$

$$z[n] = x[n - n_1] \rightarrow sys. \rightarrow w[n] = z[n] + az[n - n_0] = x[n - n_1] + ax[n - n_1 - n_0]$$

$$y[n - n_1] = x[n - n_1] + ax[n - n_1 - n_0] = w[n] \rightarrow y\ is\ a\ TI\ (Time - Invariant)\ system.$$

۲. خطی بودن (L):

$$y[n] = x[n] + ax[n - n_0]$$

$$y_1[n] = x_1[n] + ax_1[n - n_0]$$

$$y_2[n] = x_2[n] + ax_2[n - n_0]$$

$$\begin{aligned} z[n] = \alpha x_1[n] + \beta x_2[n] \rightarrow sys. \rightarrow w[n] &= z[n] + az[n - n_0] \\ &= \alpha x_1[n] + \beta x_2[n] + \alpha x_1[n - n_0] + \beta x_2[n - n_0] \\ &= \alpha(x_1[n] + ax_1[n - n_0]) + \beta(x_2[n] + ax_2[n - n_0]) = \alpha(y_1[n]) + \beta(y_2[n]) \\ &\rightarrow y\ is\ a\ L\ (Linear)\ system. \end{aligned}$$

در نتیجه $y[n]$ یک سیستم LTI می‌باشد.

- همچنین پاسخ ضربه این سیستم بصورت زیر می‌باشد:

$$h[n] = \delta[n] + a\delta[n - n_0]$$

بخش ۴-۳ : یافتن a و n_0 و خروجی سیگنال و فایل صوتی اکودار

- این قسمت از تابع بصورت زیر است:

```
h(1)=1;h(n0)=a;
y=conv(x,h);
t_y=linspace(0,numel(y)/fs,numel(y));
plot(t_y,y);
audiowrite(output,y,fs);
```

حال باید برای این قسمت مقادیر a و n_0 را محاسبه نماییم و آنرا به تابع نوشته شده بفرستیم:

- برای داشتن ۱ ثانیه تأخیر مقدار n_0 باید برابر فرکانس نمونه برداری باشد: ($fs = 44100$)

```
>> n0=44100;
```

- برای اینکه قدرت سیگنال اکو ۸۱٪ قدرت سیگنال اصلی باشد، طبق روابط انرژی سیگنال داریم:

$$\frac{81}{100} = \frac{\sum_{n=-\infty}^{\infty}|ax[n-n_0]|^2}{\sum_{n=-\infty}^{\infty}|x[n]|^2} = a^2 \rightarrow a^2 = 0.81 \rightarrow a = 0.9$$

```
>> a=0.9;
```

- در نتیجه میتوانیم با مقادیر بدست آمده پاسخ ضربه را بصورت زیر بازنویسی کنیم:

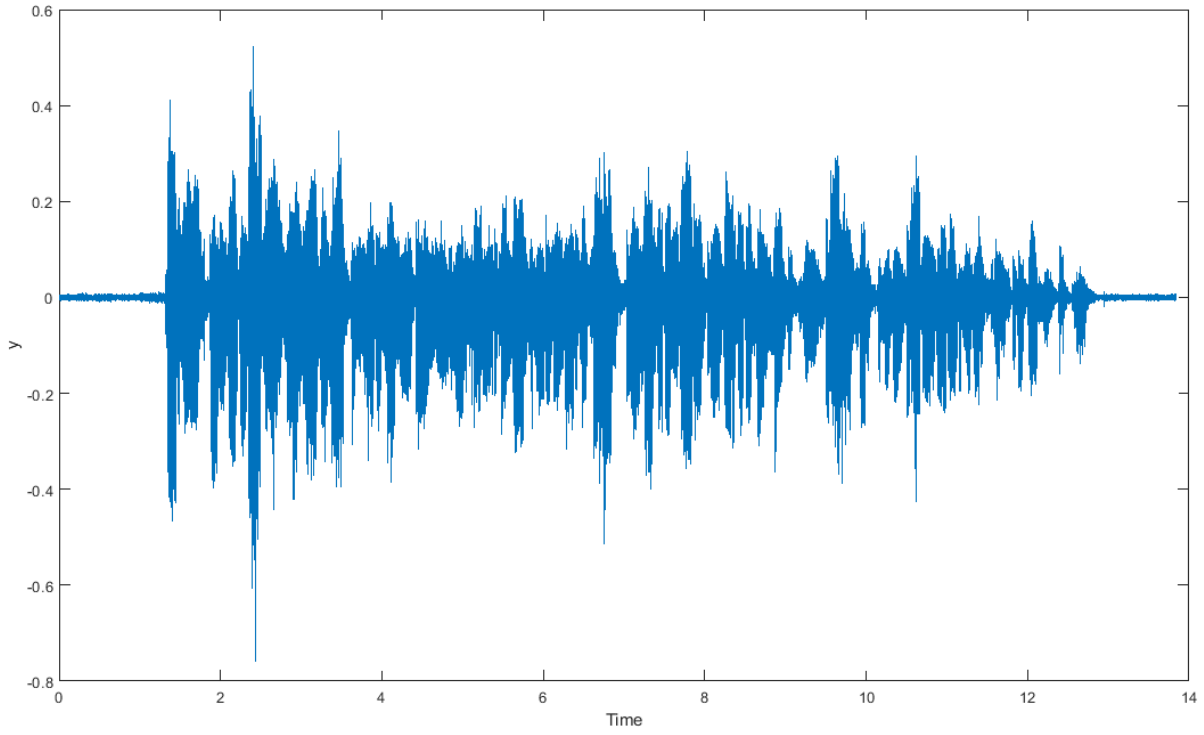
$$h[n] = \delta[n] + 0.9\delta[n-44100]$$

البته با توجه به اینکه در متلب اندیس صفر نداریم ورودی ضربه را یک واحد به راست شیفت میدهیم. پس میتوانیم پاسخ ضربه را بصورت زیر در متلب وارد نمائیم.

```
h(1)= 1;h(n0+1)=a;
```

سپس با دستور $conv$ خروجی اکودار یعنی y را به دست می‌آوریم. (تصویر ۴ خروجی y را روی محور زمان نشان میدهد.)

```
>> p4a('E:\Desktop\SS_CA1\MyTest.wav','x.wav','y.wav',n0,a);
```



تصویر۴: رسم سیگنال صوتی خروجی سوال ۴-۳

بخش ۴-۴ : پیدا کردن بهترین اکو

- با تست کردن چند نمونه بهترین اکو را با مقادیر زیر پیدا کردم:

```
>> n0=4000;
>> a=0.5;
>> p4a('E:\Desktop\SS_CA1\MyTest.wav','x.wav','y_best.wav',n0,a);
```

بخش ۴-۵ : یافتن a و n_0 از روی سیگنال ورودی $x[n]$ و خروجی $y[n]$

*** m-file مربوط به این قسمت با نام p4b.m پیوست شده است.

- روش مورد نظر این است که در ابتدا سیگنال $x[n]$ را از $y[n]$ کم کنیم تا فقط مقدار $ax[n-n_0]$ باقی بماند. سپس با استفاده از رابطه $a=\frac{R_{xy}}{R_{xx}}$ ضریب مجهول را بدست می‌آوریم و n_0 را نیز نقطه ای تعیین میکنیم که R_{xy} در آن ماکزیمم شود. (چون در حالت دستی اندیس $n-n_0$ است و n_0 از رابطه $n_0=-R_{xy}$ در متلب بدست خواهد آمد.) البته با توجه به اینکه سیگنال $x[n]$ در معادله $y[n]$ وجود دارد دیگر نیازی به حساب کردن R_{xx} نداریم و مستقیماً a را از خود R_{xy} محاسبه مینمائیم. به این صورت که مقدار ماکزیمم R_{xy} در بازه مربوط به ورودی را بر ماکزیمم R_{xy} تقسیم مینماییم.

$$n_0 = index(\max\{R_{xy}\text{در مربوط به ورودی}\})$$

$$a = \frac{\max\{R_{xy}\text{در مربوط به ورودی}\}}{\max(R_{xy})}$$

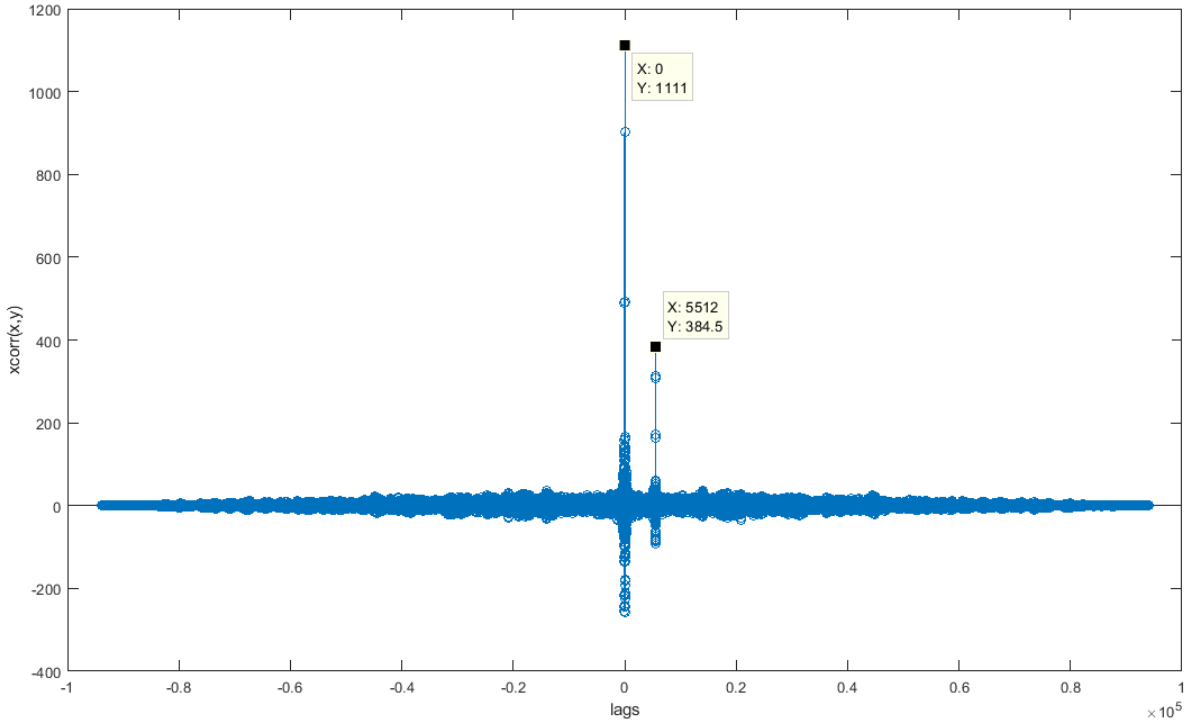
(اینکار مناسب تر است زیرا در صورتی که سیگنال $x[n]$ نویزی داشته باشد به شکل $x'[n]$ تغییر میکند و ما به آن دسترسی نداریم تا $R_{x'x'}$ را محاسبه نمائیم. پس مستقیماً از همبستگی ورودی و خروجی استفاده میکنیم.)

بخش ۴-۶ : تست کردن روش گفته شده در بخش ۴-۵

- حال با تابع $p4b$ مقادیر مجهول سیستم با ورودی $x\_test.wav$ و خروجی $y\_test.wav$ را بدست می‌آوریم:

```
>> [a,n0]=p4b('E:\Desktop\SS_CA1\x_test.wav','E:\Desktop\SS_CA1\y_test.wav')

a =

    0.3462

n0 =
```

```
5512
```



تصویر۵: نمودار همبستگی سیگنالهای ورودی و خروجی سوال ۴-۶

- همانطور که در نمودار بالا (تصویر۵) نیز مشاهده می‌شود پارامترهای مجهول سیستم بصورت زیر خواهد بود:

$$a = \frac{384.5}{1111} = 0.346$$

$$n_0 = 5512$$

با توجه به اینکه فرکانس نمونه برداری ۱۱۰۲۵ است، میتوان گفت اکو تقریباً دارای شیفت ۰.۵ ثانیه میباشد.