**Introduction to Causal and Non-Causal Signal**

**LAB # 08**

****

**Spring 2021**

**CSE301L-Signal $ System**

Submitted by: **Ashfaq Ahmad**

Registration No: **19PWCSE1795**

Class Section: **B**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Engr Durr-e-Nayab**

July 2, 2021

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**OBJECTIVES OF THE LAB**

‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐

This lab aims at the understanding of:

* *Making Signals Causal and Non‐Causal*
* *Convolution*
* *Properties of Convolution*

‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐‐

**-------------------------TASK 01--------------------------**

* Sample the signal given in above example to get its discrete‐time counterpart (take 10 samples/sec as sampling rate). Make the resultant signal causal. Display the lollipop plot of each signal.

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*Task No 01\*\*\*\*\*\*');

n=-10:0.4:10;

x1=[zeros(1,25) ones(1,26)];%Unite Step Sequence

%x2=[-2 3 4 5 -7 3 -2 -9 4 0 -5 -1 -3 5 3 2 -7 8 -2 9 -3];

%in general we can also take x2 like above but sample space will be like -10:10.

x2=sin(2\*pi\*2\*n);

x3=x1.\*x2; %length of x1 and x2 must be same.

%any signal like x2 multiplying with Unite Step signal like x1 will alway

%give Causal signal.

subplot(3,1,1)

stem(n,x1,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Unite step signal');

subplot(3,1,2)

stem(n,x2,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('General signal');

subplot(3,1,3)

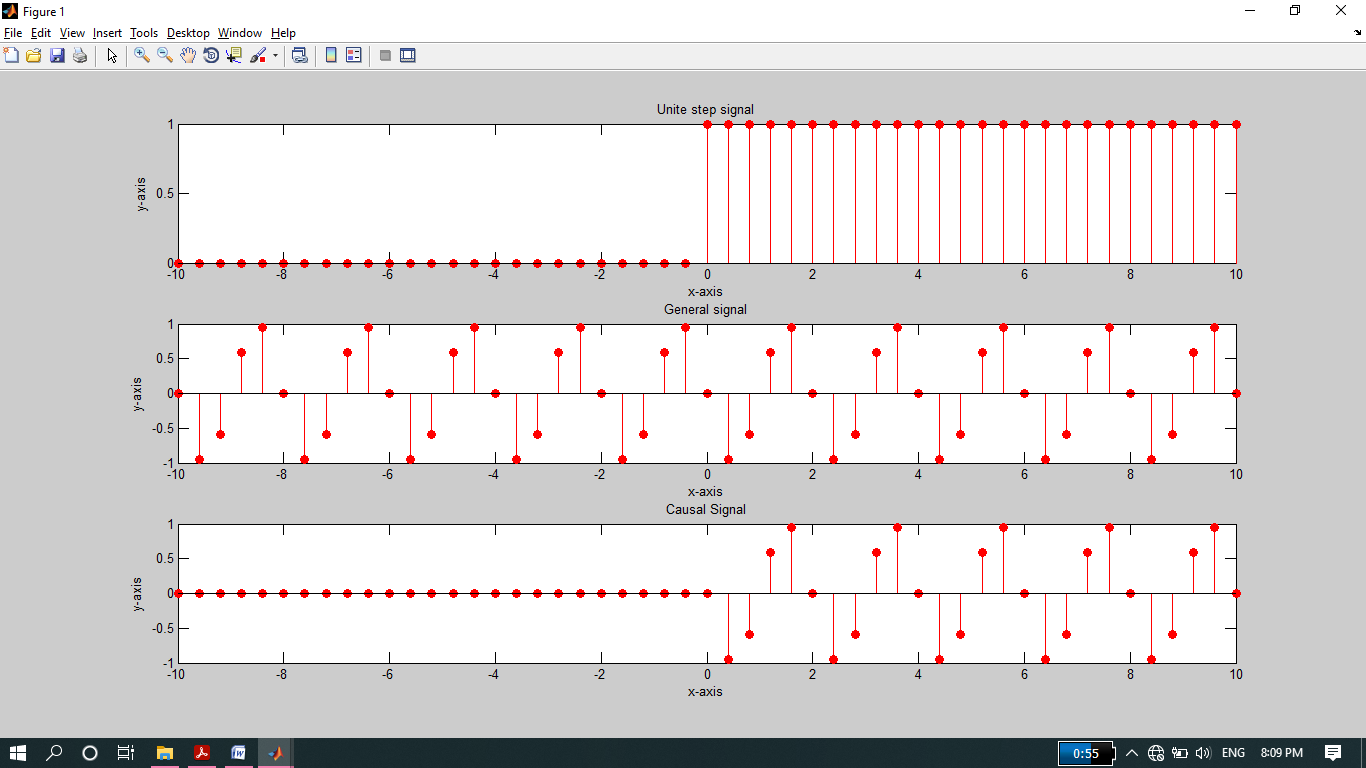
stem(n,x3,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Causal Signal');

**OUTPUT:**

****

**-----------------------TASK 02--------------------------**

* A signal is said to be anti‐causal if it exists for values of n<0. Make the signal given in above example anti‐causal.

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*Task No 02\*\*\*\*\*\*\*');

n=-10:0.4:10;

x1=[ones(1,25) zeros(1,26)];

%x2=[-2 3 4 5 -7 3 -2 -9 4 0 -5 -1 -3 5 3 2 -7 8 -2 9 -3];

x2=sin(2\*pi\*2\*n);

x3=x1.\*x2;

subplot(3,1,1)

stem(n,x1,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Unite step signal');

subplot(3,1,2)

stem(n,x2,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('General signal');

subplot(3,1,3)

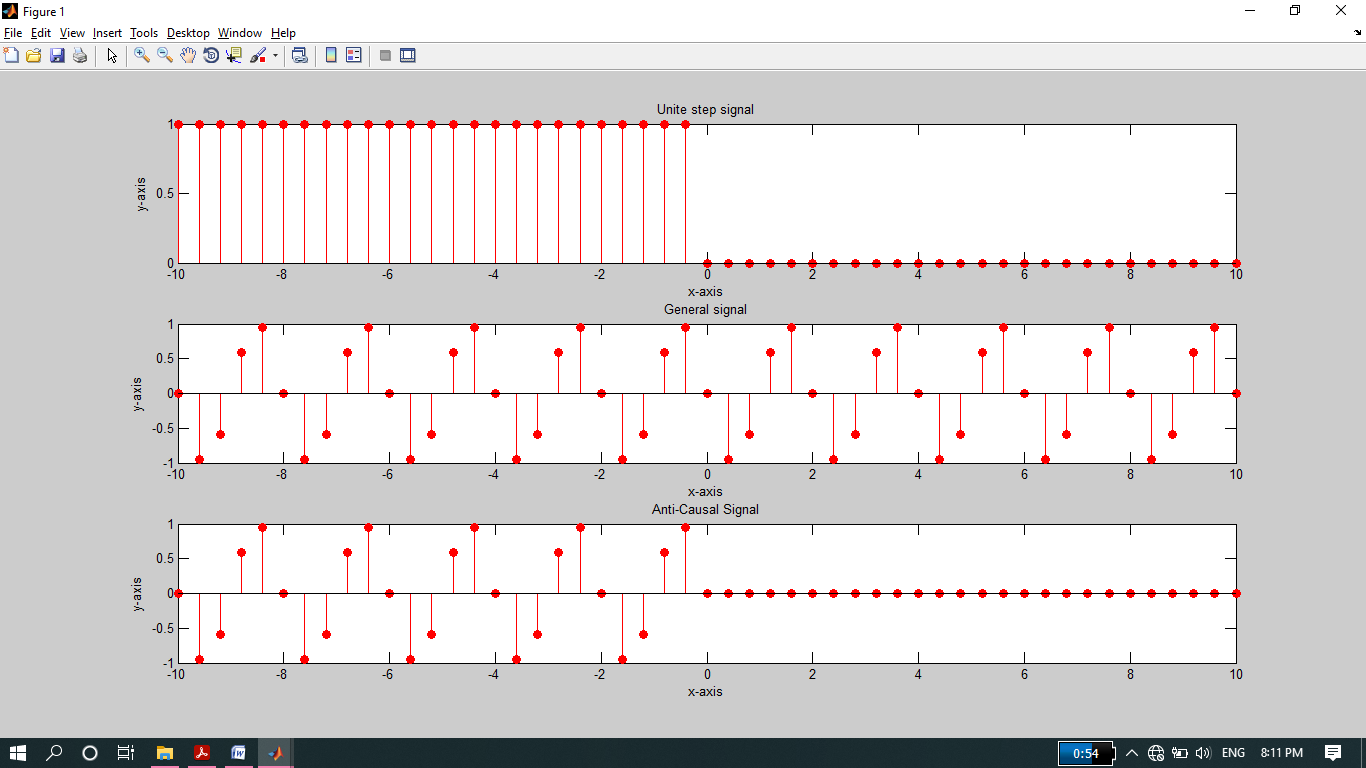
stem(n,x3,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Anti-Causal Signal');

**Output:**



**-------------------------TASK 03--------------------------**

* Create a function by name of **sig\_causal** in matlab that has two input arguments: (i) a discrete‐time signal, and (ii) a position vector. The function should make the given signal causal and return the resultant signal to the calling program.

**Source code:**

function sig\_causal

disp('\*\*\*\*\*\*Task No 03\*\*\*\*\*\*');

n=-10:10;

Signal=[-2 3 4 5 -7 3 -2 -9 4 0 8 9 -7 5 3 2 -7 8 -2 9 -3];

p\_vector=[zeros(1,10) 1 2 4 6 8 10 12 14 16 18 20];

%Position vector is a vector whose starting point is origon.

Causal\_sig=Signal.\*p\_vector;

subplot(3,1,1)

stem(n,Signal,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Discrete Signal');

grid on

subplot(3,1,2)

plot(n,p\_vector,'r','Linewidth',2);

xlabel('x-axis');

ylabel('y-axis');

title('Position Vector');

%axis([0 10 0 20]);

subplot(3,1,3)

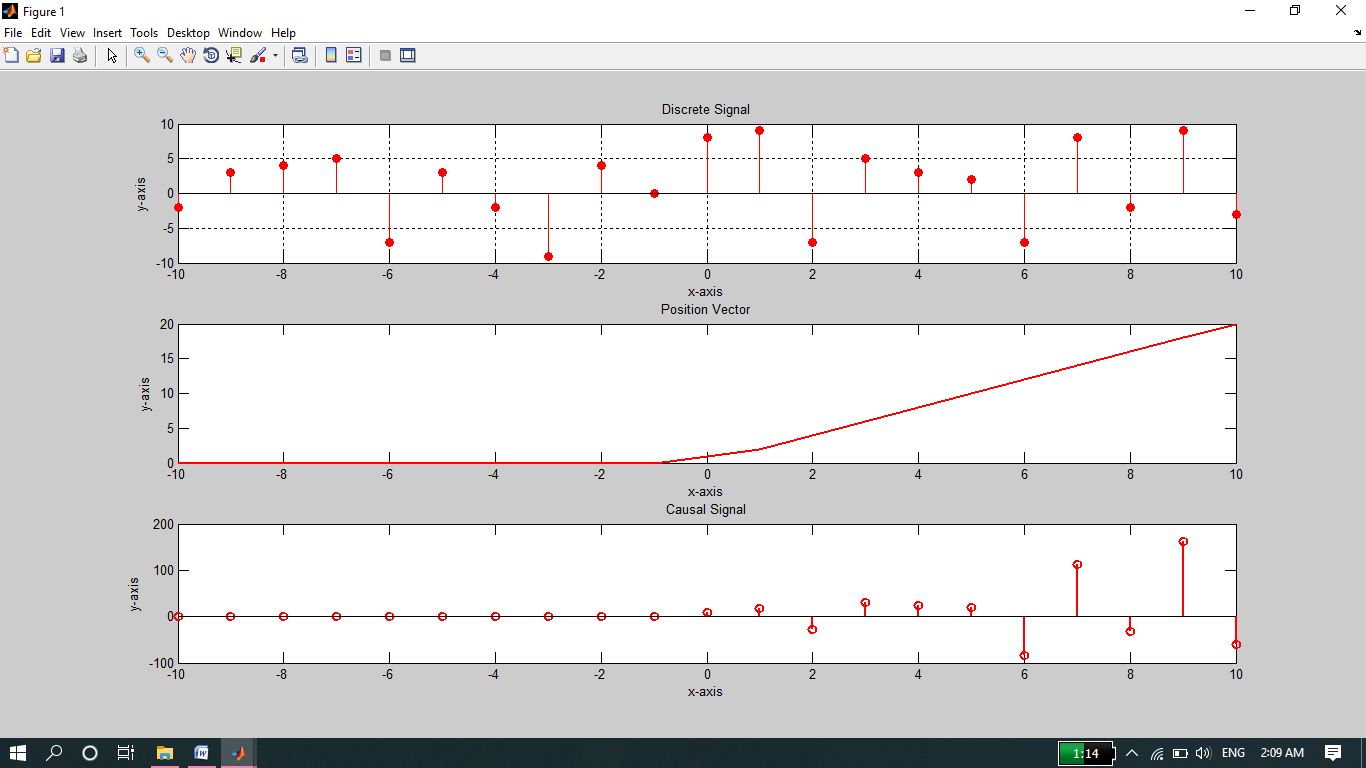
stem(n,Causal\_sig,'r','Linewidth',2);

xlabel('x-axis');

ylabel('y-axis');

title('Causal Signal');

**Output:**



**-------------------------TASK 04--------------------------**

* Convolve the following signals: x =[2 4 6 4 2];

h =[3 ‐1 2 1];

Plot the input signal as well as the output signal.

**Source Code:**

clc

clear all

close all

disp('\*\*\*\*\*\*Task No04\*\*\*\*\*\*\*\*');

x =[2 4 6 4 2];

h =[3 -1 2 1];

y=conv(h,x);

subplot(2,1,1)

stem(x,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Input Signal Signal');

grid on

subplot(2,1,2)

stem(y,'b','filled');

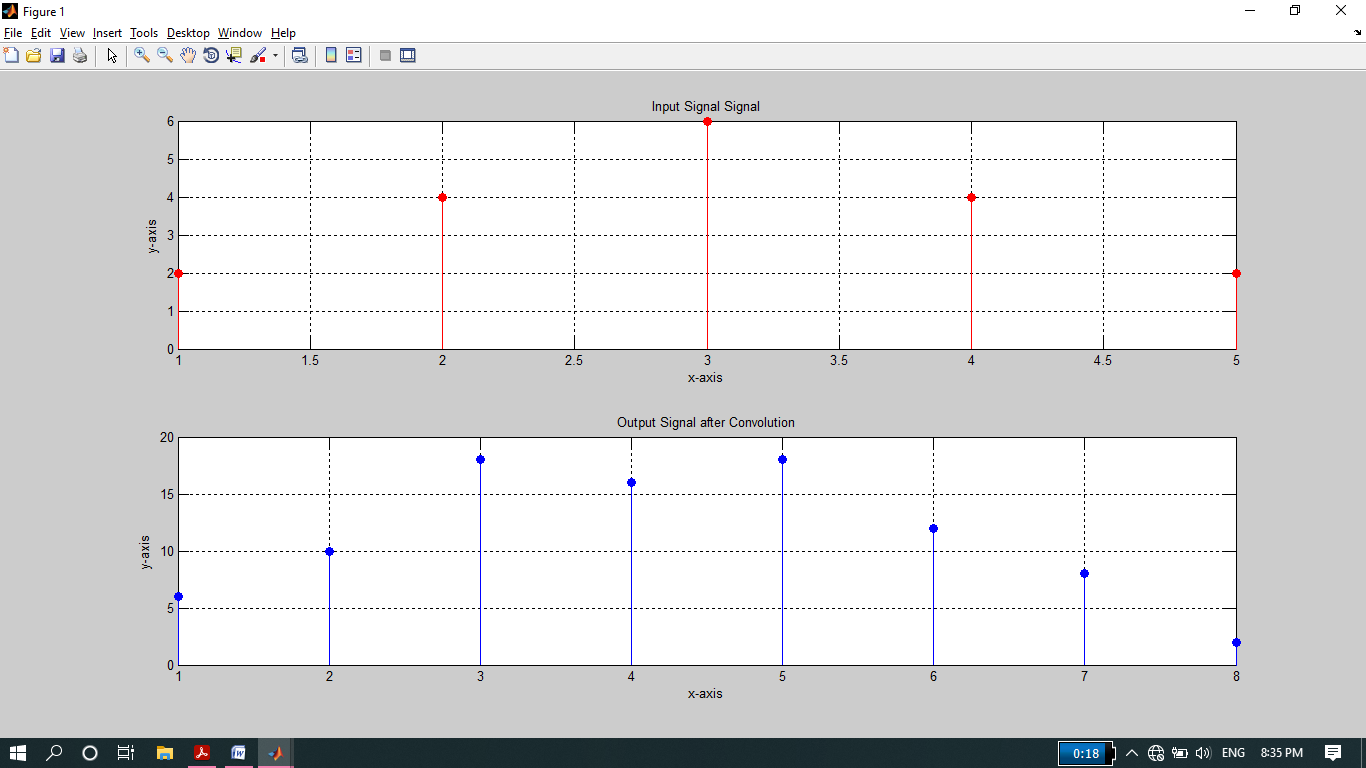
xlabel('x-axis');

ylabel('y-axis');

title('Output Signal after Convolution');

grid on

**Output:**



**-------------------------TASK 05--------------------------**

* Convolution is associative. Given the three signal x1[n], x2[n], and x3[n] as:

x1[n]= [3 1 1]

x2[n]= [4 2 1]

x3[n]=[3 2 1 2 3]

Show that (x1[n] \* x2[n]) \* x3[n] = x1[n] \* (x2[n] \* x3[n]).

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*Task No 05\*\*\*\*\*\*');

x1= [3 1 1];

x2= [4 2 1];

x3=[3 2 1 2 3];

a=conv(x1,x2);

LHS=conv(a,x3);

b=conv(x2,x3);

RHS=conv(x1,b);

subplot(2,1,1)

stem(LHS,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('LHS=(x1[n] \* x2[n]) \* x3[n]');

grid on

subplot(2,1,2)

stem(RHS,'r','filled');

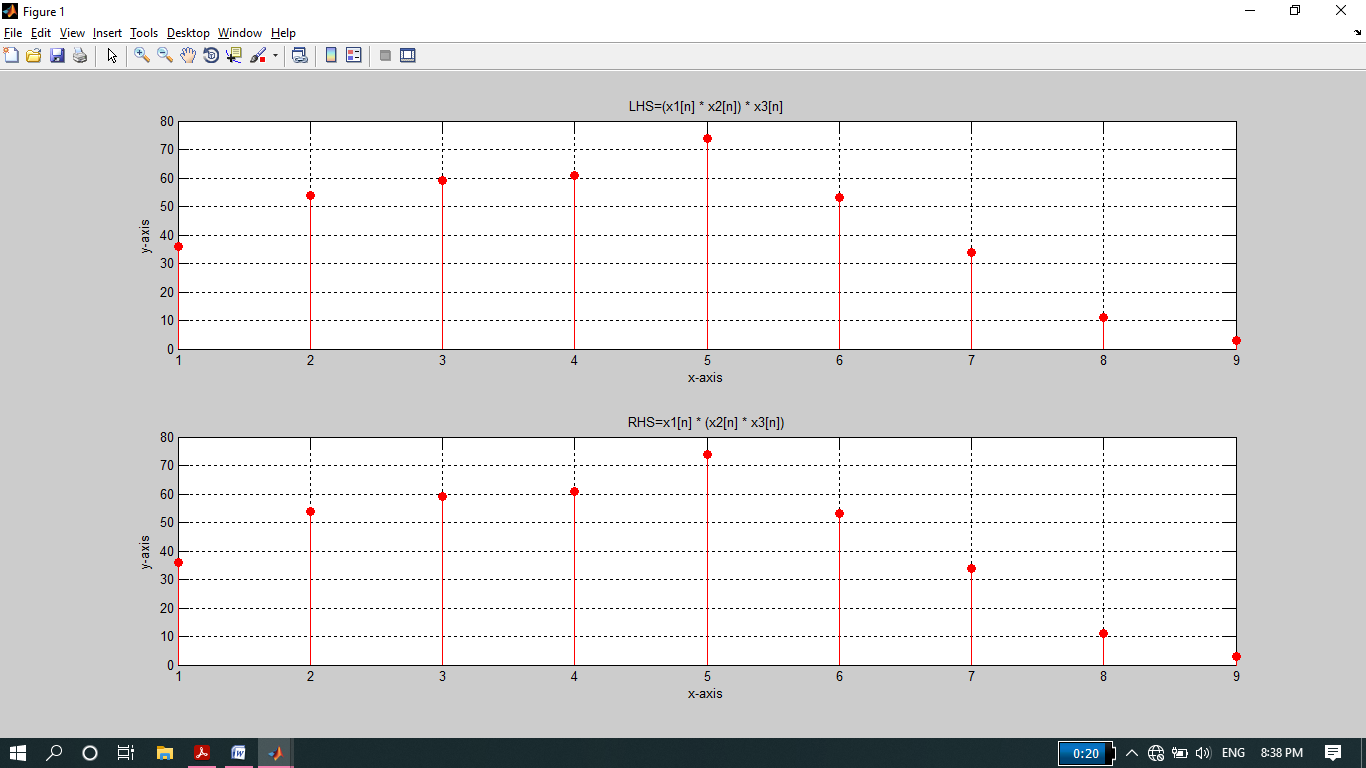
xlabel('x-axis');

ylabel('y-axis');

title('RHS=x1[n] \* (x2[n] \* x3[n])');

grid on

**Output:**



**-------------------------TASK 06--------------------------**

* Convolution is commutative. Given x[n] and h[n] as:

X[n]=[1 3 2 1]

H[n]=[1 1 2]

Show that x[n] \* h[n] = h[n] \* x[n].

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*\*Task 06\*\*\*\*\*\*\*\*\*');

X=[1 3 2 1];

H=[1 1 2];

LHS=conv(X,H);

RHS=conv(H,X);

subplot(2,1,1)

stem(LHS,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('LHS=x \* h');

grid on

subplot(2,1,2)

stem(RHS,'r','filled');

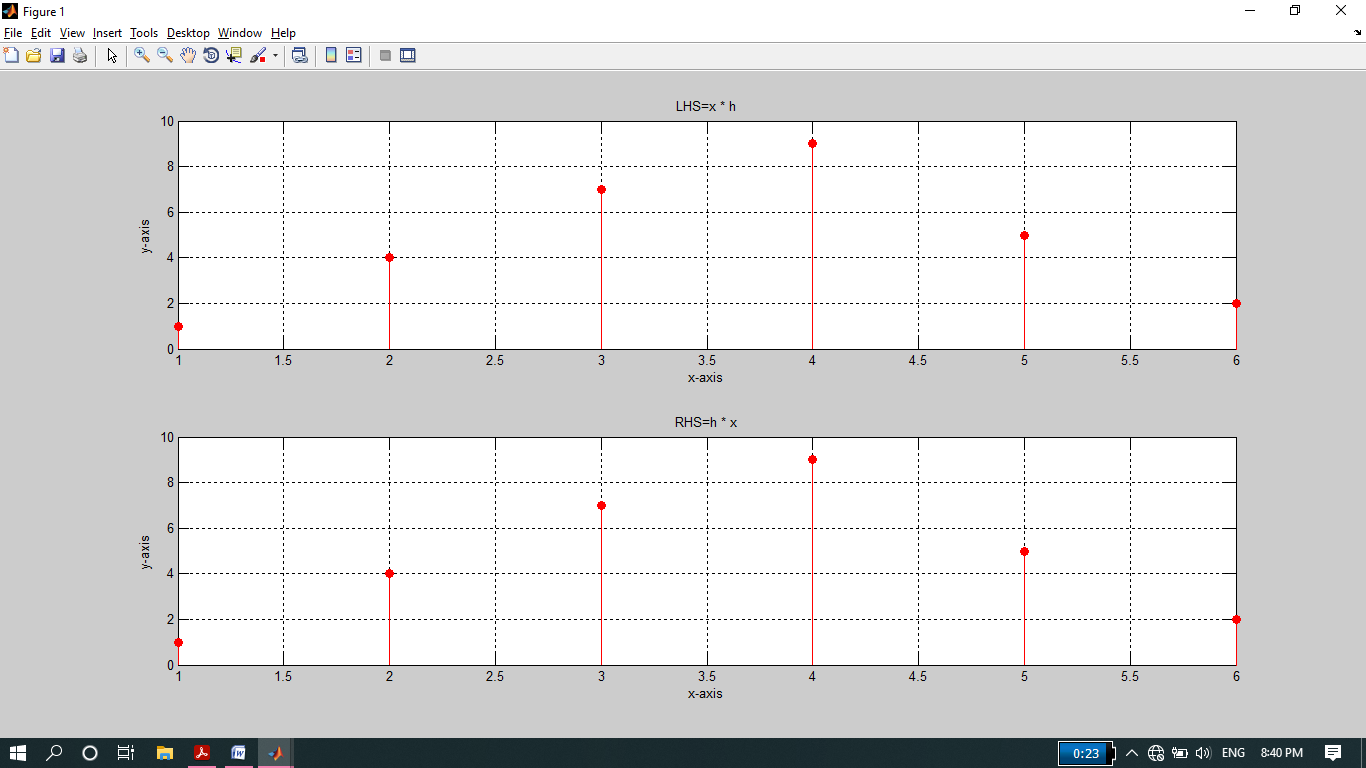
xlabel('x-axis');

ylabel('y-axis');

title('RHS=h \* x');

grid on

**Output:**

****

**-------------------------TASK 07--------------------------**

* Given the impulse response of the systems as:

h[n]= 2δ[n] + δ[n‐1]+ 2δ[n‐2]+ 4δ[n‐3]+ 3δ[n‐4]

If the input x[n] = δ[n]+ 4δ[n‐1] +3δ[n‐2] + 2δ[n‐3] is applied to the system, determine the output of the system.

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*\*Task 07\*\*\*\*\*\*\*');

h=[2 1 2 4 3];

(h is the summation of five unite impulse signals having d/f position of 1)

x=[1 4 3 2];

y=conv(h,x);

subplot(2,1,1)

stem(x,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Input Signal');

grid on

subplot(2,1,2)

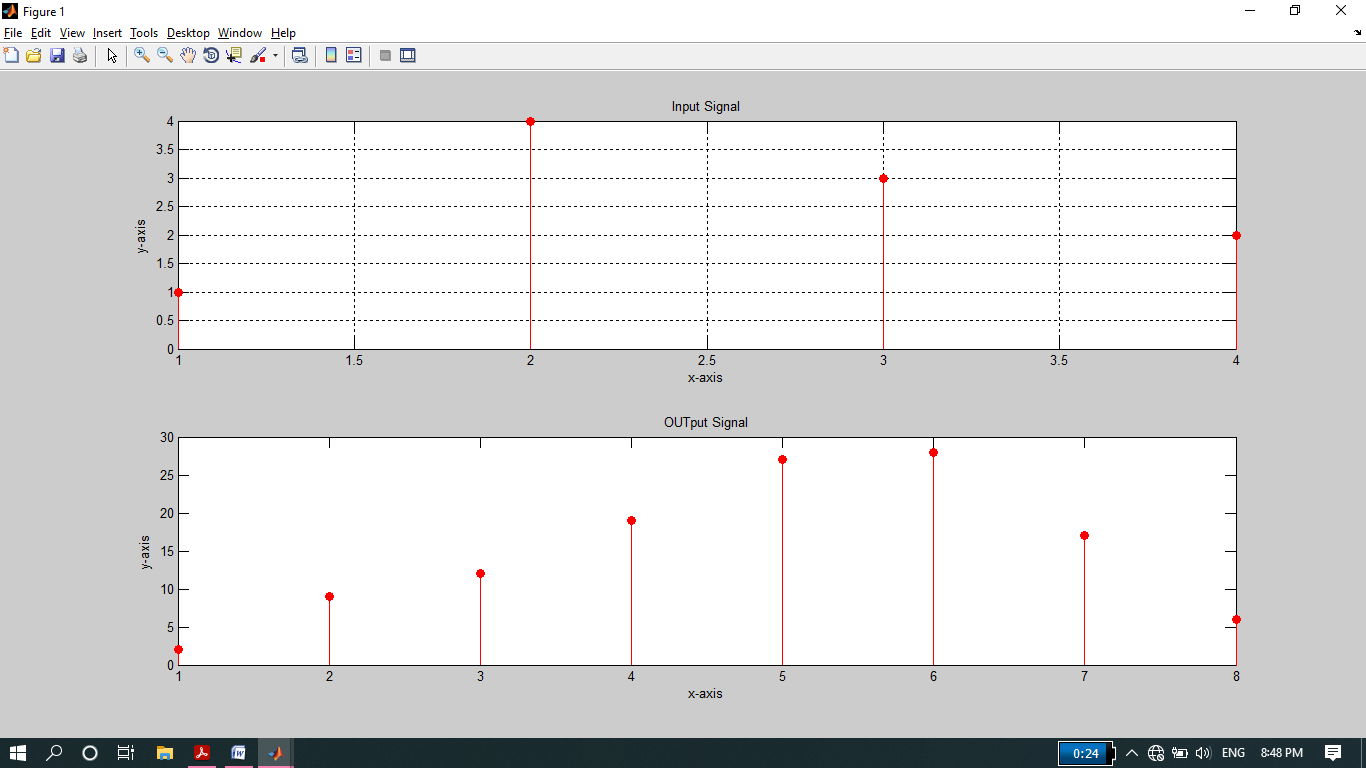
stem(y,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

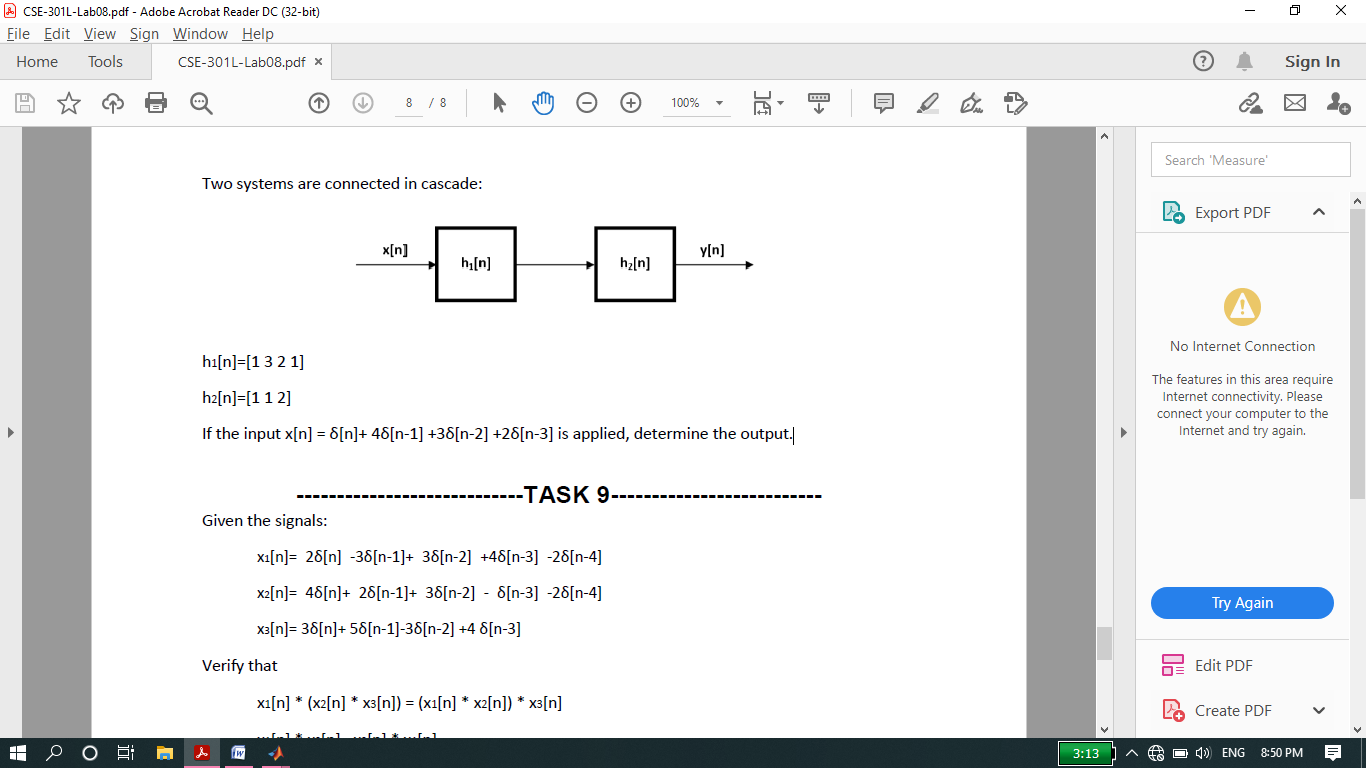
title('OUTput Signal');

**Output:**



**-------------------------TASK 08--------------------------**

* Two systems are connected in cascade:



h1[n]=[1 3 2 1]

h2[n]=[1 1 2]

If the input x[n] = δ[n]+ 4δ[n‐1] +3δ[n‐2] +2δ[n‐3] is applied, determine the output.

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*\*Task 08\*\*\*\*\*\*\*');

x=[1 4 3 2];

h1=[1 3 2 1];

h2=[1 1 2];

y1=conv(h1,x);

y2=conv(h2,y1);

subplot(3,1,1)

stem(x,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('Input Signal');

grid on

subplot(3,1,2)

stem(y1,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('OUtput signal when the input signal passes the first system');

grid on

subplot(3,1,3)

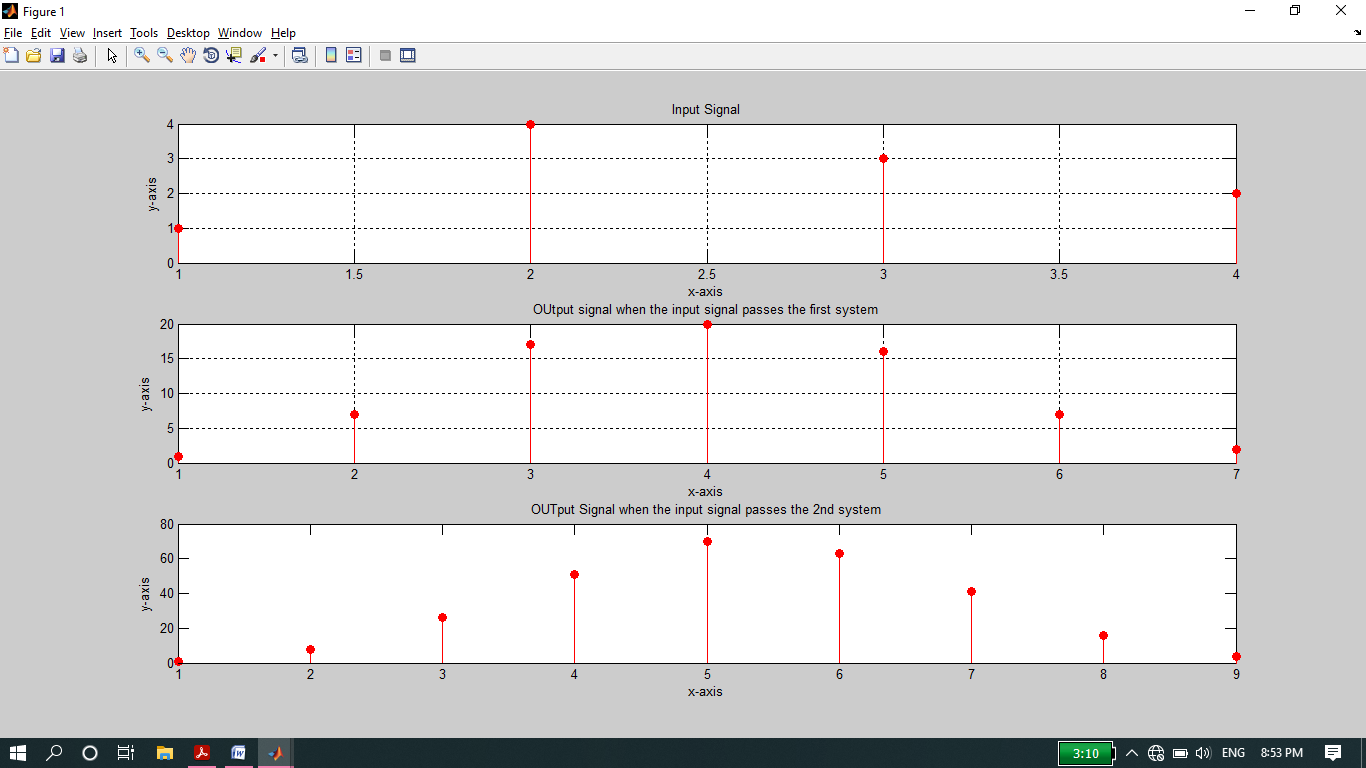
stem(y2,'r','filled');

xlabel('x-axis');

ylabel('y-axis');

title('OUTput Signal when the input signal passes the 2nd system');

**Output:**



**-------------------------TASK 09--------------------------**

Given the signals:

* x1[n]= 2δ[n] ‐3δ[n‐1]+ 3δ[n‐2] +4δ[n‐3] ‐2δ[n‐4]
* x2[n]= 4δ[n]+ 2δ[n‐1]+ 3δ[n‐2] ‐ δ[n‐3] ‐2δ[n‐4]
* x3[n]= 3δ[n]+ 5δ[n‐1]‐3δ[n‐2] +4 δ[n‐3]

Verify that:

* x1[n] \* (x2[n] \* x3[n]) = (x1[n] \* x2[n]) \* x3[n]
* x1[n] \* x2[n]= x2[n] \* x1[n]

**Source code:**

clc

clear all

close all

disp('\*\*\*\*\*\*\*\*Task 09\*\*\*\*\*\*\*');

x1=[2 -3 3 4 -2];

x2=[4 2 3 -1 -2];

x3=[3 5 -3 4];

a=conv(x2,x3);

LHS\_1=conv(x1,a);

b=conv(x1,x2);

RHS\_1=conv(b,x3);

LHS\_2=conv(x1,x2);

RHS\_2=conv(x2,x1);

figure(1)

subplot(2,1,1)

stem(LHS\_1,'filled');

xlabel('X-axis');

ylabel('Y-axis');

title('LHS=x1 \* (x2 \* x3)');

grid on

subplot(2,1,2)

stem(RHS\_1,'filled');

xlabel('X-axis');

ylabel('Y-axis');

title('RHS=(x1 \* x2) \* x3');

grid on

figure(2)

subplot(2,1,1)

stem(LHS\_2,'filled');

xlabel('X-axis');

ylabel('Y-axis');

title('LHS=(x1\*x2)');

grid on

subplot(2,1,2)

stem(RHS\_2,'filled');

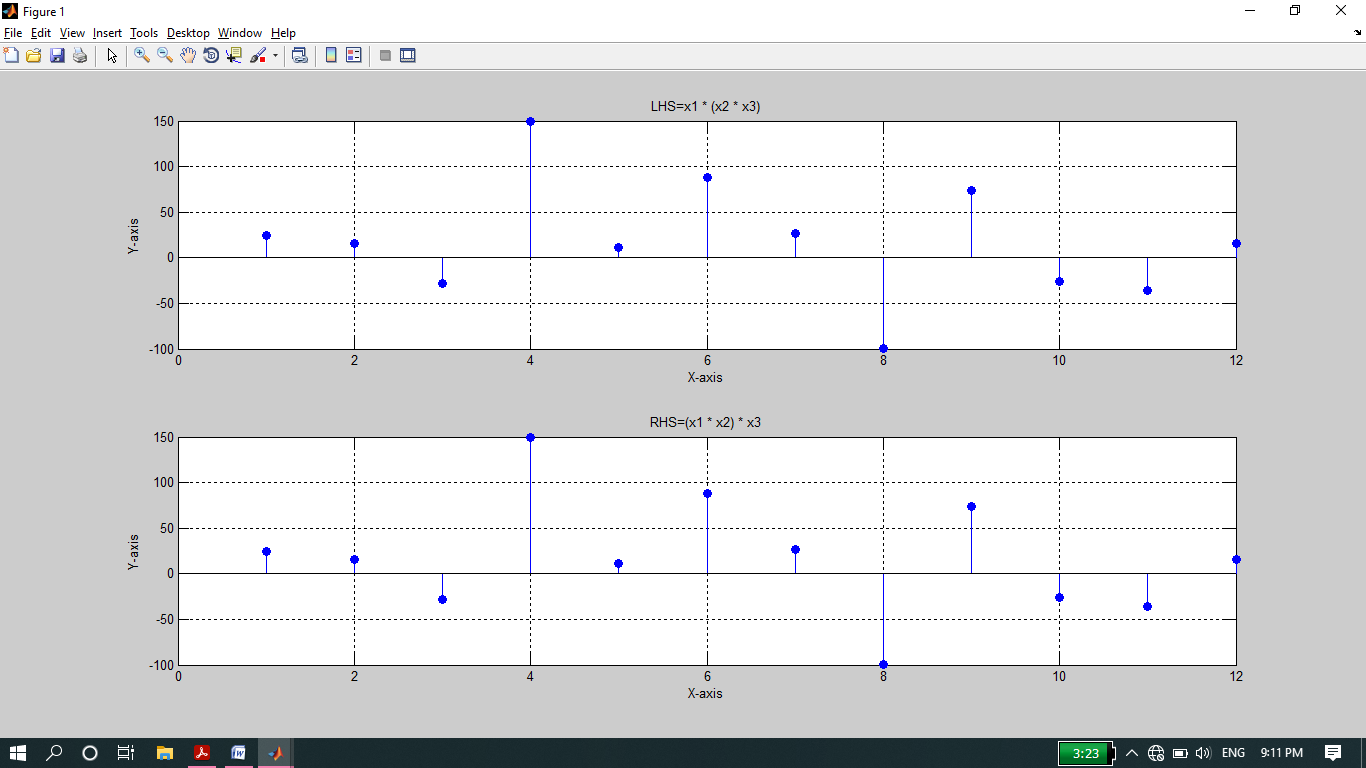
xlabel('X-axis');

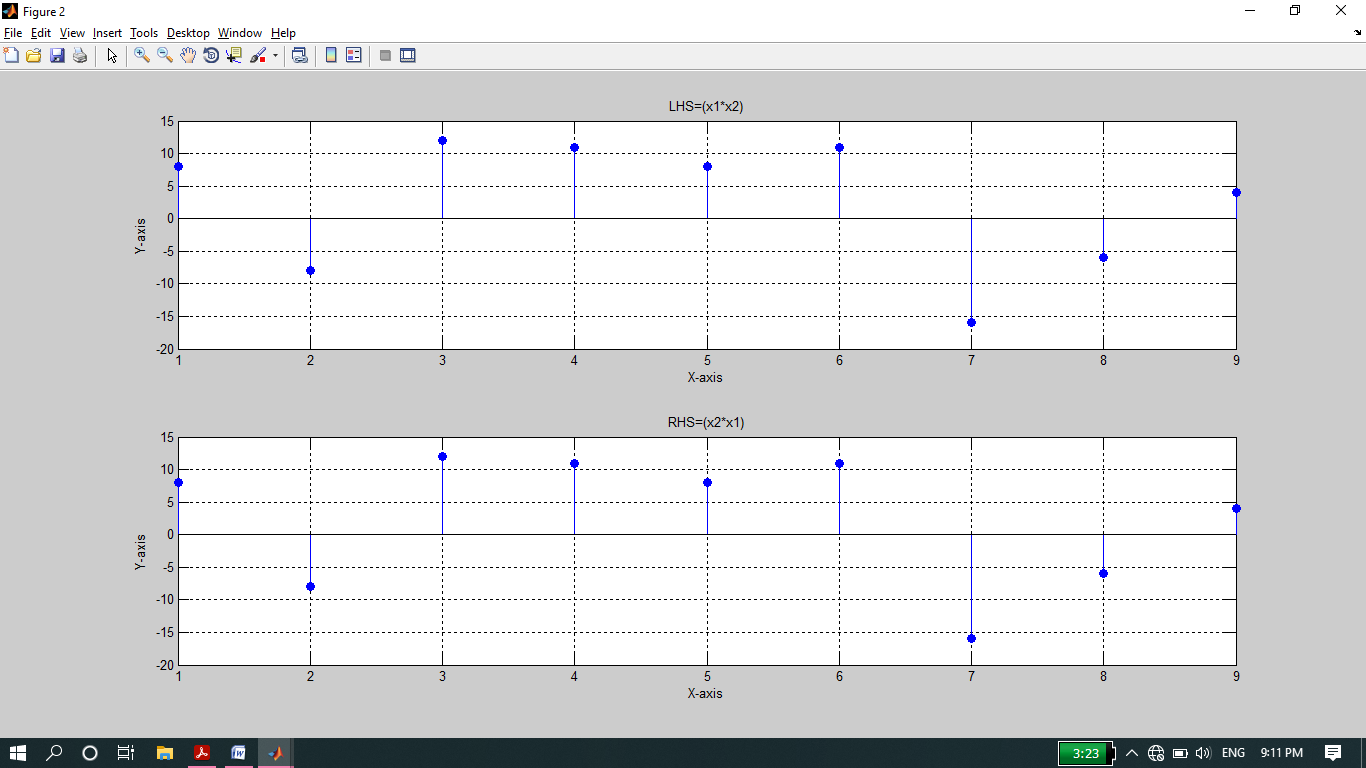
ylabel('Y-axis');

title('RHS=(x2\*x1)');

grid on

**Output:**



****

**THE END**