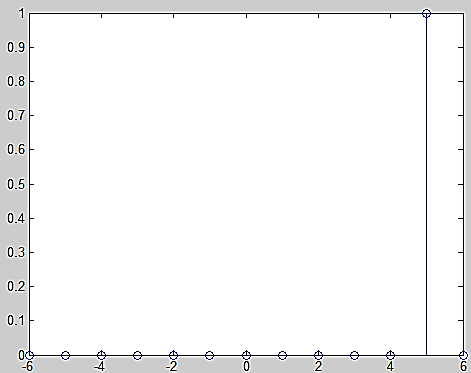
**LAB 5**

**OBJECT: To understand manipulation of this parameter (Time-delay, Time-scaling and Time-Reversal) of both analog & discrete time signal.**

**TASK 01:** Write a script that the delays the unit impulse by 5 samples.

**OUTPUT:**



**CODING:**

x=-6:6;

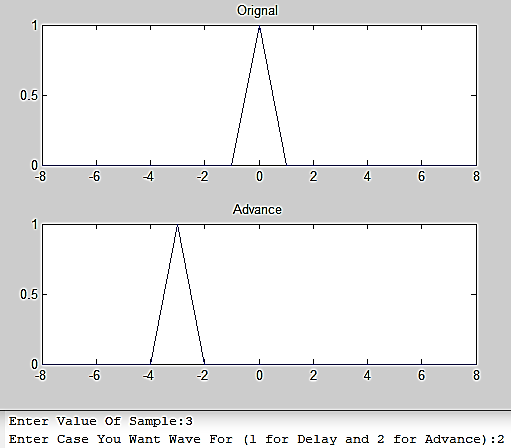
y = x==5;

stem(x,y)

**TASK 02:** Generate a script that uses user input to shift unit impulse right to left. Use MATLAB input command to take the number of samples to shift the signal and use switch case option and if-else statement to decide whether to shift to right (delay) or to the left (advance**).**

**CODING: (With “Case” function)**

**OUTPUT:**



x=-8:8;

s =input('Enter Value Of Sample:');

c =input('Enter Case You Want Wave For (1 for Delay and 2 for Advance):');

n= x==0;

subplot(2,1,1)

plot(x,n)

title('Orignal')

switch c

case 1

n = x==0+ s;

subplot(2,1,2)

plot(x,n)

title('Delay')

case 2

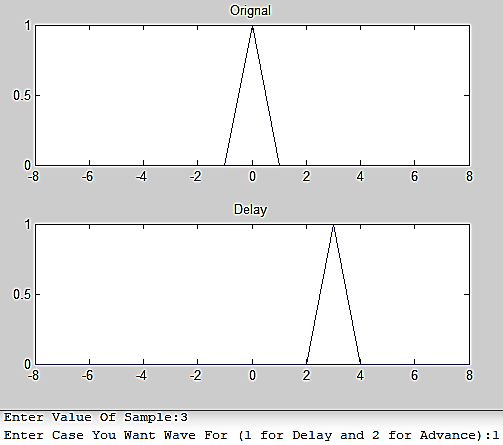
n = x==0- s;

subplot(2,1,2)

plot(x,n)

title('Advance')

end

 **CODING: (With “IF-ELSE” function)**

x=-8:8;

s =input('Enter Value Of Sample:')

c =input('Enter Case You Want Wave For (1 for Delay and 2 for Advance):')

n= x==0;

subplot(2,1,1)

plot(x,n)

title('Orignal')

if (c==1)

n = x==0+ s;

subplot(2,1,2)

plot(x,n)

title('Delay')

elseif (c==2)

n = x==0- s;

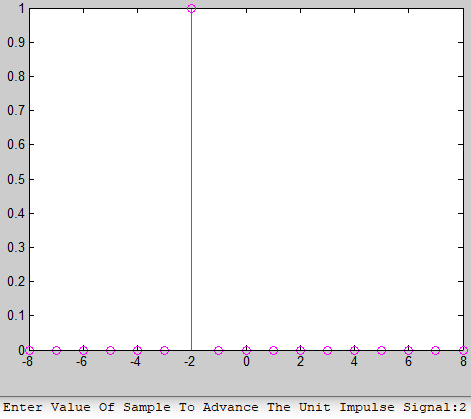
subplot(2,1,2)

plot(x,n)

title('Advance') end

**TASK 03:** Write a script that uses user input to advance the unit impulse signal.

**OUTPUT:**



**CODING:**

x=-8:8;

s =input('Enter Value Of Sample To Advance The Unit Impulse Signal:')

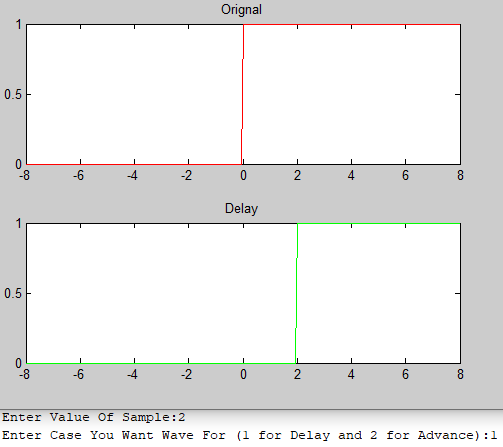
n= x==0-s;

stem(x,n,'m')

**TASK 04** **:** Repeat task # 2 with unit step response.

**CODING:**

**OUTPUT:**



x=-8:0.05:8;

s =input('Enter Value Of Sample:');

c =input('Enter Case You Want Wave For (1 for Delay and 2 for Advance):');

n= x>=0;

subplot(2,1,1)

plot(x,n,'r')

title('Orignal')

switch c

case 1

n = x>=0+ s;

subplot(2,1,2)

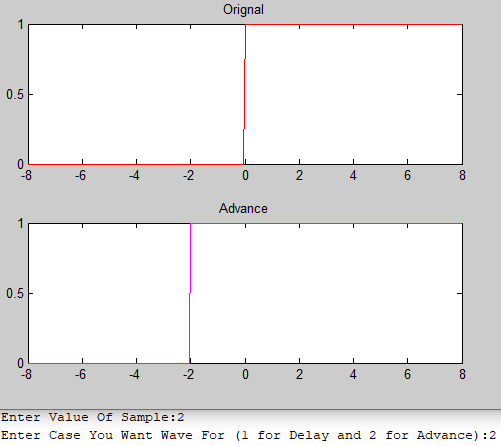
plot(x,n,'g')

title('Delay')

case 2

n = x>=0- s;

subplot(2,1,2)



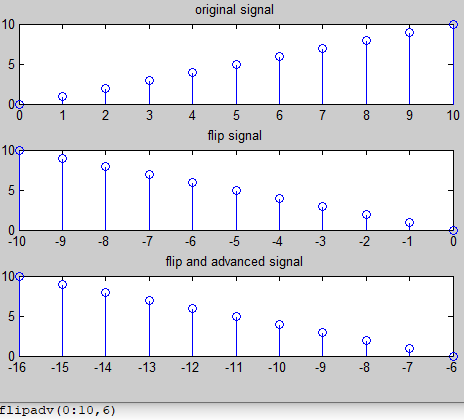
plot(x,n,'m')

title('Advance')

end

**TASK 05** **:** Create a function that flip the sequence first then advance the sequence. The function will receive two arguments as defined below. First is input signal which needs to be flipped and then advance and second is the number of samples by which the signal should be advanced [x]=flipadv (u,s).

**OUTPUT**



**CODING:**

function[x]=flipadv(u,s)

n = u ;

x=fliplr(u);

nl = -fliplr(n);

subplot(3,1,1)

stem(n,u)

title('original signal')

subplot(3,1,2)

stem(nl,x)

title('flip signal')

n2=nl-s;

subplot(3,1,3)

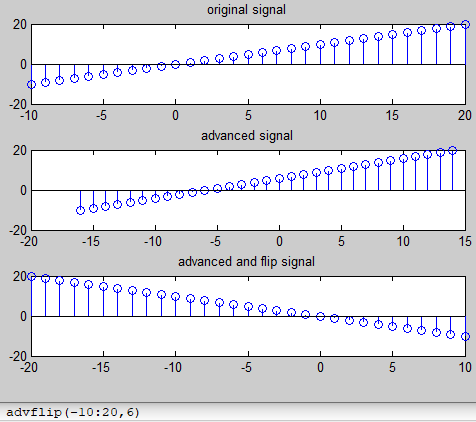
stem(n2,x)

title('flip and advanced signal')

end

**TASK 06** **:** Generate a function that advance the sequence first then flip the sequence. The function will receive two arguments as defined below. First is input signal which needs to be moved and then flipped and second is the number of samples by which the signal should be advanced [x]=advflip (u,s).

**OUTPUT**



**CODING:**

function [x] = advflip(u,s)

n =u;

x=fliplr(u);

n1=-fliplr(n)

subplot(3,1,1)

stem(n,u)

title('original signal')

n2=n-s;

subplot(3,1,2)

stem(n2,u)

title('advanced signal')

subplot(3,1,3)

stem(n1,x)

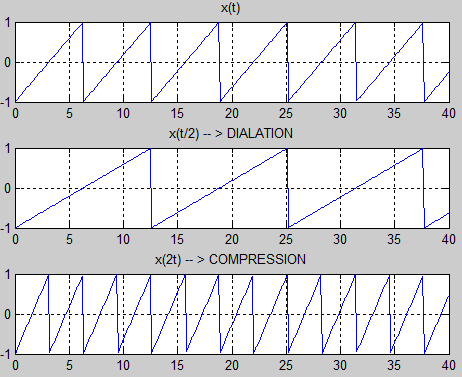
title('advanced and flip signal')

end

**LAB ASSIGNMENT 4**

**TASK 07:** Create a triangular wave with unit amplitude. Use subplot command to show the effect of compression and dialation.

**OUTPUT**



**CODING:**

t = 0:0.1:40;

x = sawtooth(t);

subplot(3,1,1)

plot(t,x)

grid on

title('x(t)')

x1 = sawtooth(t/2);

subplot(3,1,2)

plot(t,x1)

grid on

title('x(t/2) -- > DIALATION')

x2 = sawtooth(t\*2);

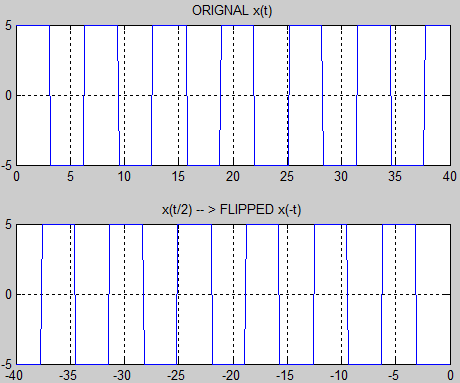
subplot(3,1,3)

plot(t,x2)

grid on

title('x(2t) -- > COMPRESSION')

**OUTPUT**



**TASK 08:** Create a square wave with unit amplitude form -5:5. Plot the signal which shows the original signal along with its flipped signal.

**CODING:**

t = 0:0.1:40;

x = 5\*square(t);

x1 = -fliplr(x);

t1 = -fliplr(t);

subplot(2,1,1)

plot(t,x)

grid on

title('ORIGNAL x(t)')

subplot(2,1,2)

plot(t1,x1)

grid on

title('x(t/2) -- > FLIPPED x(-t)')