Department of Electrical Engineering

Faculty Member: Usman Ilyas **Dated: 18-2-2016**

Course/Section: BEE-6B **Semester: 4th semester**

EE-232 Signals and Systems
Lab #3Signal Transformations

Name	Reg. no.	Report Marks / 10	Viva Marks / 5	Total/15
Saad Iqbal	**32903**			

3.0 Pre-Lab Tasks

Use of Matlabto Flip, compress and expand an audio sequence in time:

i) Load the audio file titled "song.wav" using MATLAB. Play the audio file.

Audio can load in matlab using audioread Command and play using sound command.

CODE:

```
clear all;
clc;
[Y,Fs]=audioread('test_song.wav');%audio file in the open
directory
sound(Y,Fs);%play sound at sample rate Fs
```

ii) Flip the audio sequence and play it back again. (Hint: The operation is time reversal).

Audio sequence flips by invert up to down a matrix

CODE:

```
A=flipud(Y);
sound(A,Fs);%play sound at sample rate Fs
```

iii) Next perform time compression on the flipped signal by a factor of 2. What does the audio sequence sound like?

Compress audio by reducing samples by factor of 2, taking value with interval of 2.

CODE:

```
B=A(1:2:end,:);
sound(B,Fs);%play sound at sample rate Fs
```

Its played Faster.

iv) Perform time expansion by a factor of 2. What does the flipped audio sequence sound like?

v)

Expansion of audio is done by repeating every sample by two while same sample frequency while playing sound.

CODE:

```
C=B(floor(1:0.5:end),:);
sound(C,Fs);%play sound at sample rate Fs
```

Its played slower.

Hints: You may use the commands 'fliplr' 'flipud' for flipping the signal. Also the same can be achieved using a for loop.

3.1 Basic trasformation of signals

a) Transformations of the Time Index for Discrete time signals:

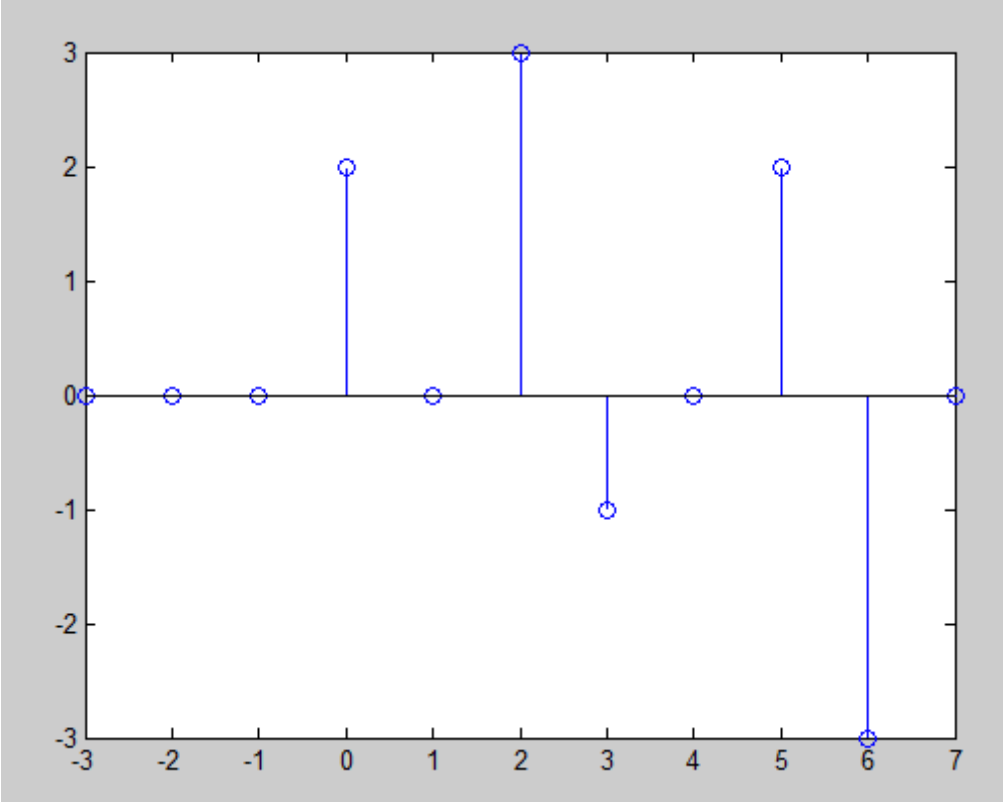
i) Define a Matlab vector ***n,***such that -3 ≤n < 7, representing time/sample indices of a Discrete Time Signal **x[n]** such that :

$$x[n] = \begin{cases} 2, & n = \mathbf{0} \\ 3, & n = \mathbf{2} \\ -1, & n = \mathbf{3} \\ 2, & n = \mathbf{5} \\ -3, & n = \mathbf{6} \\ 0, & elsewhere \end{cases}$$

CODE:

```
n=-3:7
x=[0 0 0 2 0 3 -1 0 2 -3 0]
stem(n,x)
```

OUTPUT:



ii) Make a generalized **function** that can shift the signal **x[n]**by delaying or advancing it by a specified amount.
*HINT (You can make use of 'find' command in Matlab).
The inputs to the function should be the data and shift while output should be the shifted signal. Please make sure that the indices accompanying the shifted signal are correct.

CODE:

```
function [B,Bn]=shiftinga(A,n,a)
if size(A)==size(n)
    B=A;
    Bn=a+n(1):a+n(end)
else
    disp('Error in size of data and index vector')
    end
    stem(Bn,B)
end
```

How to use:

```
Shiftinga(A,n,a)
```

c) Use of Matlab command zeros and ones:

iii) Implement the following function in Matlab. Your function should take ***k*** and ***n*** as input from user.

$$u[k-n] = \begin{cases} 1, k \geq n \\ 0, k < n \end{cases}$$

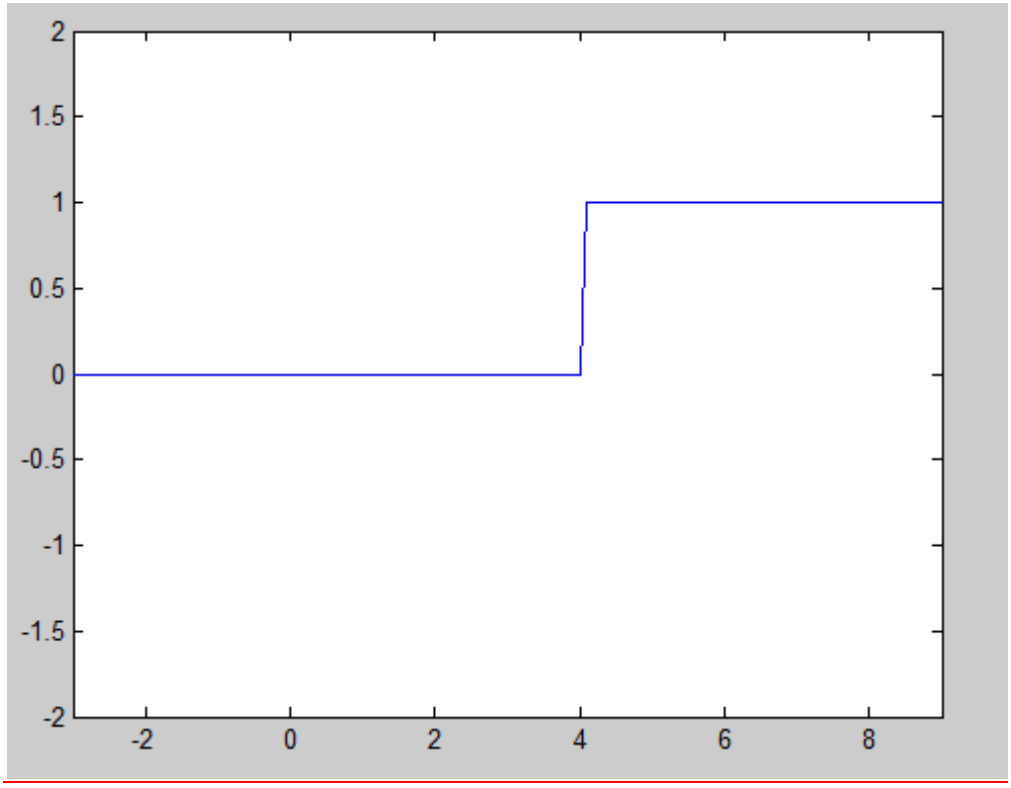
CODE:

```
function [unit]=step(k,n)
unit=[];
for i=1:length(n)%loop for every point check
if(k<n(i))
    unit(i)=ones;
else
    unit(i)=zeros;
end
i=i+1;
end
plot(n,unit)
axis([n(1) n(end) -2 2])
```

How to use:

Step(k,n) % k is number,n is array

OUTPUT:



d) Even and odd part of a signal:

iv) Every signal ***x[n]*** is sum of its even part and odd part. Given the signal ***x[n]*** find its even and odd part.

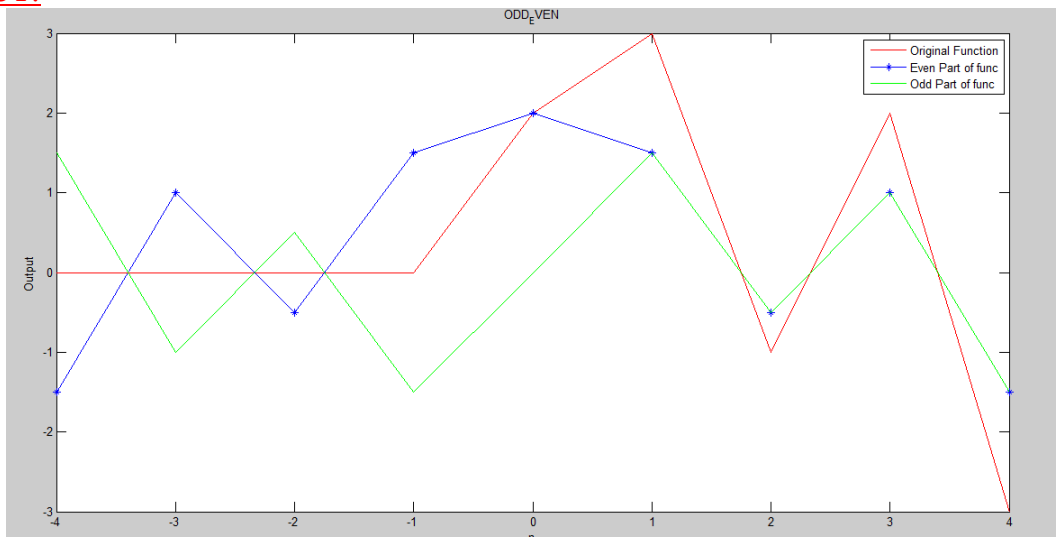
$$x[n] = \begin{cases} 2, & n = 0 \\ 3, & n = 1 \\ -1, & n = 2 \\ 2, & n = 3 \\ -3, & n = 4 \\ 0, & elsewhere \end{cases}$$

$$x_e[n] = \frac{1}{2}\,[x[n] + x[-n]],\ x_o[n] = \frac{1}{2}\,[x[n] - x[-n]]$$

CODE:

```
clear all;
clc;
n=-4:4;
x=[0,0,0,0,2,3,-1,2,-3];
for i=n+5
    xe(i)=(x(i)+x(10-i))/2;
    xo(i)=(x(i)-x(10-i))/2;
end
plot(n,x,'r',n,xe,'b-*',n,xo,'g')
title('ODD_EVEN')
xlabel('n')
ylabel('Output')
legend('Original Function','Even Part of func','Odd Part of func')
```

OUTPUT:



d) Image Transformation:

v) Load the file 'image.jpg' in Matlab. See the image. You should see a fighter throwing a punch. Assuming that each row is a separate signal performs time reversal of each row. Now see the image. The fighter is now throwing a punch with which hand?

CODE:

```
img=imread('image.jpg');%Load image
[H W C]=size(img);%obtaining dimensions of image
A(:,:,1)=fliplr(img(:,:,1));%flipping left to right image
A(:,:,2)=fliplr(img(:,:,2));
A(:,:,3)=fliplr(img(:,:,3));
imagesc(A)%display image
```

OUTPUT: