



**Department of Electrical Engineering and**  
**Computer Science**

Faculty Member: Dr. Salman Ghafoor

Dated: 26/09/2022

Semester: 5<sup>th</sup>

Section: BEE 12C

**EE-232: Signals and Systems**

**Lab 3: Signal Transformations**

**Group Members**

Name	Reg. No	PL04 - CL03	PL05 - CL03	PL08 - CL04	PL09 - CL04
		Viva / Quiz / Lab Performance	Analysis of data in Lab Report	Modern Tool Usage	Ethics and Safety
		5 Marks	5 Marks	5 Marks	5 Marks
Danial Ahmad	331388				
Muhammad Umer	345834				
Syeda Fatima Zahra	334379				



## **1 Table of Contents**

<b>2</b>	<b>Signal Transformations.....</b>	<b>3</b>
2.1	Objectives .....	3
2.2	Equipment .....	3
2.3	Lab Instructions .....	3
<b>3</b>	<b>Pre-Lab .....</b>	<b>4</b>
<b>4</b>	<b>Exercise.....</b>	<b>4</b>
4.1	Transformations of the Time Index for Discrete Time Signals .....	4
4.2	Discrete Unit Step.....	6
4.3	Even and Odd Part of a Signal.....	7
4.4	Image Transformation.....	8
<b>5</b>	<b>Conclusion .....</b>	<b>9</b>



## **2 Signal Transformations**

### **2.1 Objectives**

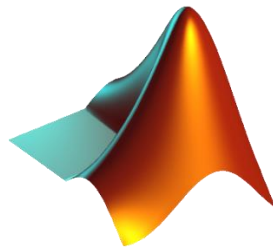
This lab experiment has been designed to introduce students to signal transformations using MATLAB.

- How to perform time shifting
- How to perform time compression/expansion
- Even and odd Signals

### **2.2 Equipment**

Software

- *MATLAB*



### **2.3 Lab Instructions**

All questions should be answered precisely to get maximum credit. Lab report must ensure following items:

- Lab objectives
- MATLAB codes
- Results (Graphs/Tables) duly commented and discussed
- Conclusion



### 3 Pre-Lab

Using MATLAB to flip, compress and expand an audio sequence in time;

1. Load the audio file titled “song.wav” using MATLAB. Play the audio file.
2. Flip the audio sequence and play it back again.
3. Next perform time compression on the flipped signal by a factor of 2. What does the audio sequence sound like?

**Answer:** As the signal is compressed by a factor of 2, the audio sequence plays twice as fast with duration being reduced from 9.13 s to 4.565 s. As frequency also alters with compression, the pitch also changes and the audio sounds high-pitched.

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

As the signal is expanded by a factor of 2, the audio sequence plays twice as long with duration being increased from 9.13 s to 18.26 s. As frequency also alters with expansion, the pitch also changes and the audio sounds deeper-pitched.

Following code accomplishes all of the aforementioned pre-lab tasks in a sequential fashion:

```
[y, Fs] = audioread('test_song.wav');  
sound(y, Fs) % Plays the audio  
pause(12); % Adds a delay of n seconds  
yflip = flipud(y); % Reverses the column vector  
sound(yflip, Fs)  
pause(12);  
sound(yflip, Fs * 2) % Compressed signal  
pause(6);  
sound(yflip, Fs / 2) % Expanded signal
```

### 4 Exercise

#### 4.1 Transformations of the Time Index for Discrete Time Signals

- a) Define a MATLAB vector  $n$ , such that  $-3 \leq n < 7$ , representing time/sample indices of a Discrete Time Signal  $x[n]$  such that:

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

*Enter the MATLAB command(s) through which you defined this vector.*



```
n = -3:6;  
x = zeros(1, length(n));  
x(4) = 2; x(6) = 3; x(7) = -1; x(9) = 2; x(10) = -3;  
stem(n, x)
```

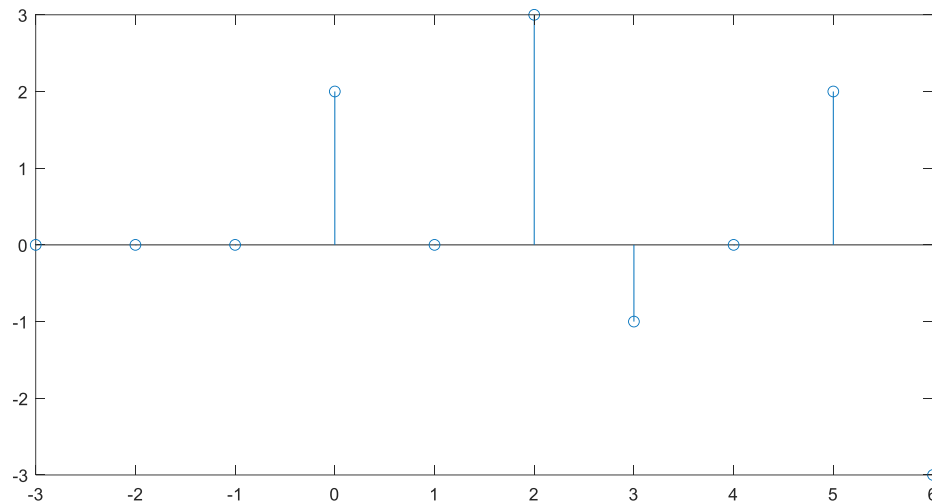


Figure 4.1.a DT Signal

- b) Make a generalized function that can shift the signal  $x[n]$  by delaying or advancing it by a specified amount.

### Function Definition

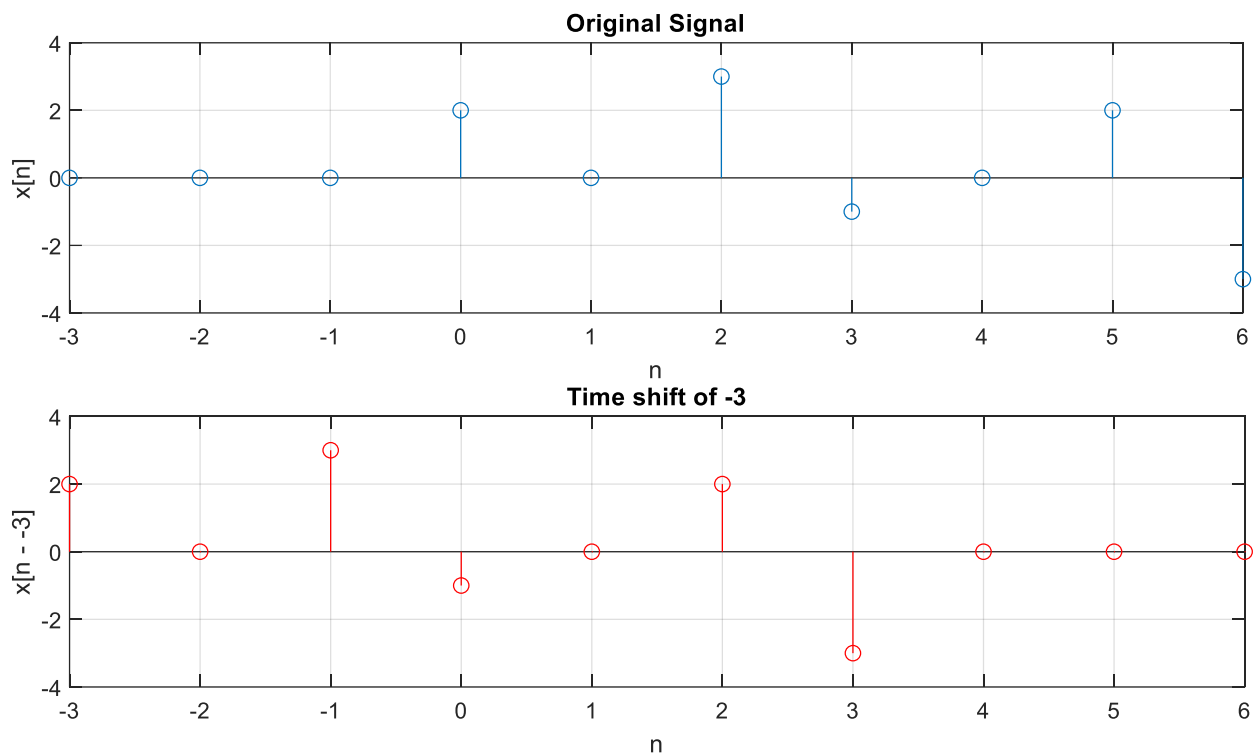
```
function y = timeshift(data, shift)  
    x = data(1, :);  
    n = data(2, :);  
  
    % Different slicing for negative  
    % / positive shifts  
    if shift < 0  
        y = x((abs(shift) + 1):length(x));  
        y = [y, zeros(1, abs(shift))];  
    else  
        for k = 1:length(n)  
            y(k + shift) = x(k);  
        end  
        y = y(1:(length(y) - shift));  
    end  
end
```

### Plotting Script

```
subplot(211)  
stem(n, x)  
grid on
```



```
xlabel('n')
ylabel('x[n]')
title('Original Signal')
subplot(212)
stem(n, y, 'r')
xlabel('n')
ylabel(['x[n - ' num2str(shift) ']''])
title(['Time shift of ' num2str(shift)])
grid on
```



## 4.2 Discrete Unit Step

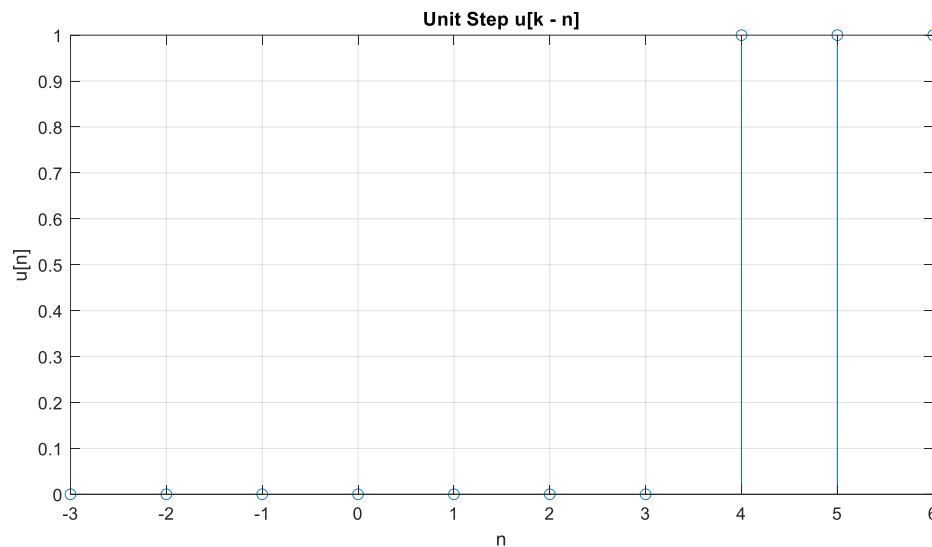
Implement the following function in MATLAB. Your function should take  $k$  and  $n$  as input from user.

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

```
function u = unitstep(k, n)
    u = zeros(1, length(n));
    indexer = find(n == k);
    u(indexer:length(u)) = 1;
    stem(n, u);
    grid on
    title('Unit Step u[k - n]')
    xlabel('n')
    ylabel('u[n]')
end
```



Attach a screenshot of the figure generated by your function for the input values of  $k = 4$ .



### 4.3 Even and Odd Part of a Signal

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

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

$$\text{Even: } x_e[n] = \frac{1}{2} [x[n] + x[-n]] \quad \text{Odd: } x_o[n] = \frac{1}{2} [x[n] - x[-n]]$$

Write a function that will produce three figures. The first figure shall show the signal  $x[n]$  given above. The second figure shall show the signal  $x_e[n]$ . The third figure shall show the signal  $x_o[n]$ .

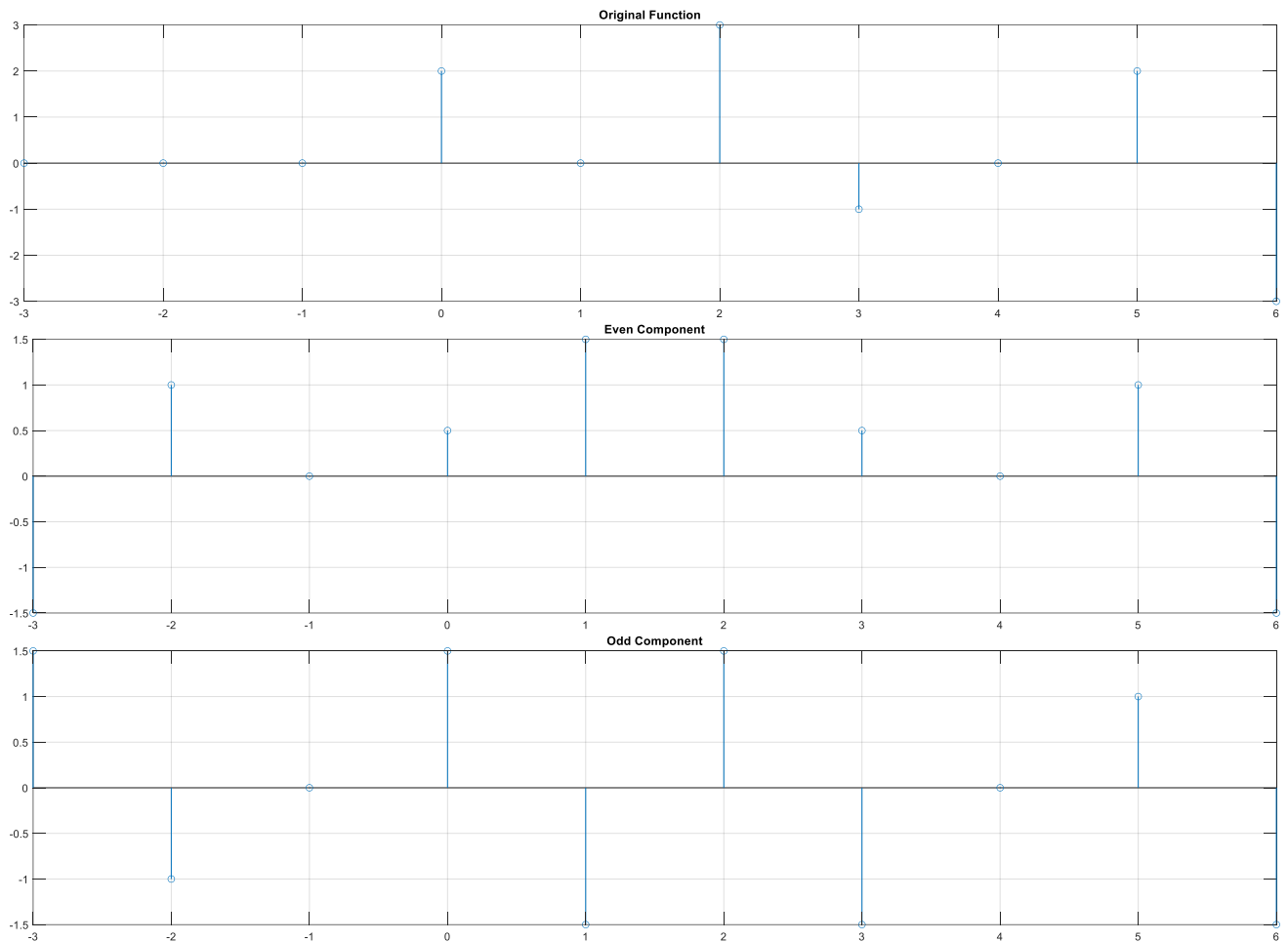
```
% Signal Definition
n = -3:6;
x = zeros(1, length(n));
x(4) = 2; x(6) = 3; x(7) = -1; x(9) = 2; x(10) = -3;

x_e = 1/2 * (x + fliplr(x)); % Even
x_o = 1/2 * (x - fliplr(x)); % Odd

figure
stem(n, x)
title('Original Function')
grid
figure
stem(n, x_e)
```



```
title('Even Component')  
grid  
figure  
stem(n, x_o)  
title('Odd Component')  
grid
```

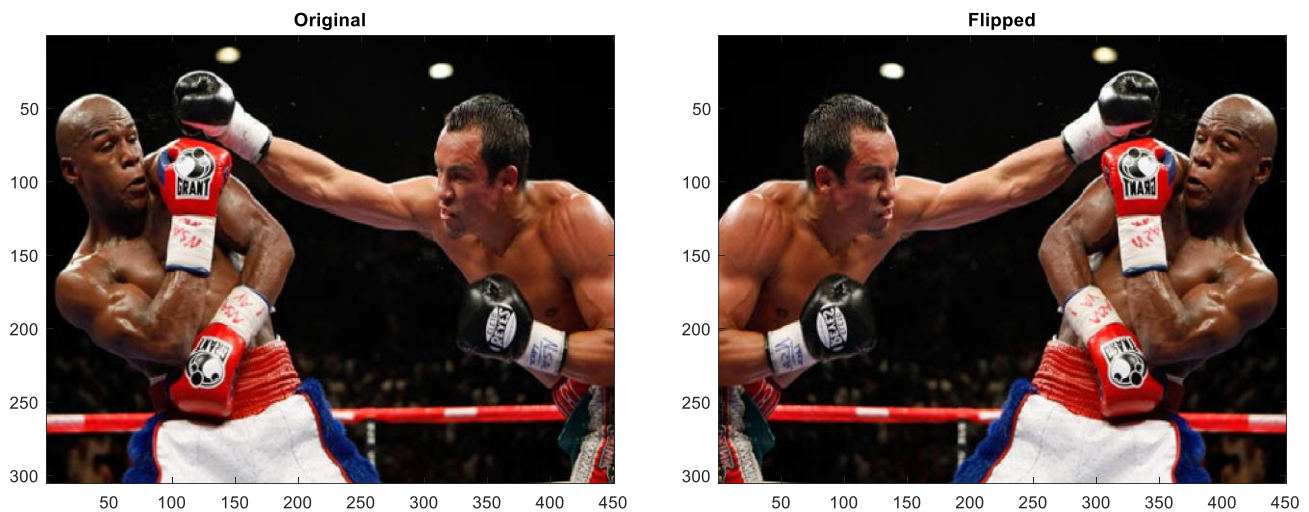


## 4.4 Image Transformation

Load the file 'image.jpg' in MATLAB. See the image. You should see a fighter throwing a punch.

```
img = imread('image.jpg');  
[height, width, channels] = size(img);  
figure  
imagesc(img)  
title('Original')  
figure  
imagesc(fliplr(img))  
title('Flipped')
```





Assuming that each row is a separate signal perform time reversal of each row. Now see the image. The fighter is now throwing a punch with which hand?

**Answer:** After time reversal (fliplr), we observe that the fighter is throwing the punch with his left-hand as opposed to the right-hand as in the original image; implying that time reversal is working as expected.

## 5 Conclusion

In this lab, we further familiarized ourselves with MATLAB, and we learned how to perform time shifting. We learned how to plot Discrete time graphs using stem(), and we got to know how to flip vectors and matrices using flipud() and fliplr(). In addition to that, we imported an audio file and performed compression and expansion on it. We also learned how to flip an image using the fliplr() and flipud() functions, and further performed operations which separated odd and even signals from a given function.