Heaven's Light is Our Guide

RAJSHAHI UNIVERSITY OF ENGINEERING & TECHNOLOGY

Department of Electrical & Computer Engineering

DIGITAL SIGNAL PROCESSING SESSIONAL

LAB REPORT

Number of Experiment: 02

Course Title: ***Digital Signal Processing Sessional***

Course No: ECE 4124

Date of Submission: May 08, 2023

Submitted by:

Marjia Afroj

Series: 18

Roll: 1810029

4th year Odd semester

Dept. of ECE, RUET

Submitted to:

Hafsa Binte Kibriya

Lecturer

Dept. of ECE, RUET

2.1 Experiment No: 02

2.2 Experiment Date: March 03, 2023

2.3 Experiment Name: Study of circular convolution, summation and subtraction of two discrete signal and drawing specific signals MATLAB

2.4 Theory:

Convolution is a mathematical tool for combining two signals to produce a third signal. Circular convolution is a mathematical operation that involves multiplying two finite-length sequences and summing the products over a circular range of indices. In other words, it is a convolution that assumes the inputs are periodic, with the period being the length of the sequences.

To perform circular convolution, we take the Discrete Fourier Transform (DFT) of the input sequences, multiply them point-wise, and then take the inverse DFT of the result. The result of the circular convolution is another finite-length sequence with the same length as the input sequences.

If x(n) and h(n) are two discrete signals and N is the number of samples of largest sequence the resultant circular convolution-

$$y(n) = \sum_{k=0}^{N-1} x(k) * h(n-k)$$

2.5 Used Platform: MATLAB

2.6 Code:

2.6.1: Code for circular convolution

```
clc;
clear all;
close all;

x=[1 2 3 4];
h=[4 4 3 2];

L1=length(x);
L2=length(h);

L=max(L1,L2);

L3=L1-L2;
if(L3>0)
    x=[x,zeros(1,L3)];
else
    h=[h,zeros(1,-L3)];
end
y = zeros(1,L);

for i=1:L

    for j=1:L
```

```
        k=i-j+1;
        if(k<=0)
            k=L+k;
        end
        y(i)=y(i)+(x(j)*h(k));
    end
end
disp('The resultant is');
y

subplot(3,1,1);
stem(x, 'r');
xlabel('n values');
ylabel('Amplitude');
title("x(n)");


subplot(3,1,2);
stem(h, 'r');
xlabel('n values');
ylabel('Amplitude');
title("h(n)");


subplot(3,1,3);
stem(y, 'r');
xlabel('n values');
ylabel('Amplitude');
title("y(n)");
```

Output:

```
The resultant is

y =

    33    30    31    36
```

Figure 2.1: Output of convolution without using cconv() function

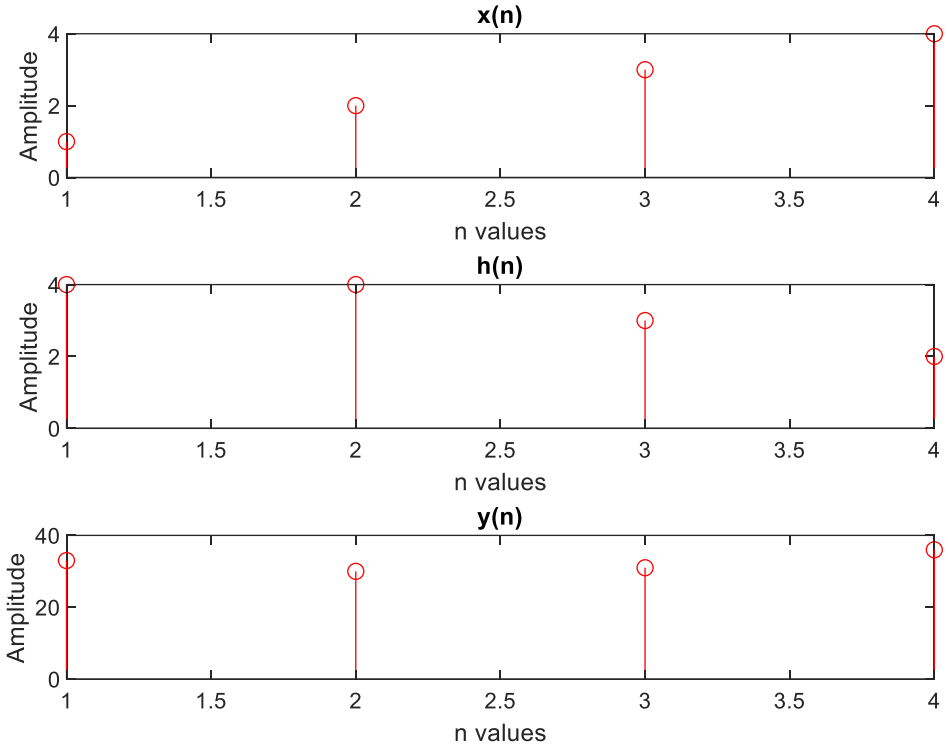

Figure 2.2: Graphical representation of the output of circular convolution without using cconv() function

2.6.2: Code for summation and subtraction of two discrete signal:

```
clc;

clear all;
close all;

n1 = [0 0 0 2 2 2 1 1 1 0 2];
n2 = [2 2 0 1 1 1 0 0 0 0 3];

sum = n1 + n2;
sum
sub = n1-n2;
sub

subplot(4,1,1);
stem(n1, 'r');
xlabel('n values');
ylabel('Amplitude');
title("Signal 1(n1)");


subplot(4,1,2);
stem(n2, 'r');
xlabel('n values');
ylabel('Amplitude');
title("Signal 2(n2)");
```

```
subplot(4,1,3);
stem(sum, 'r');
xlabel('n values');
ylabel('Amplitude');
title("Signal 3(sum)");

subplot(4,1,4);
stem(sub, 'r');
xlabel('n values');
ylabel('Amplitude');
title("Signal 4(sub)");
```

Output:

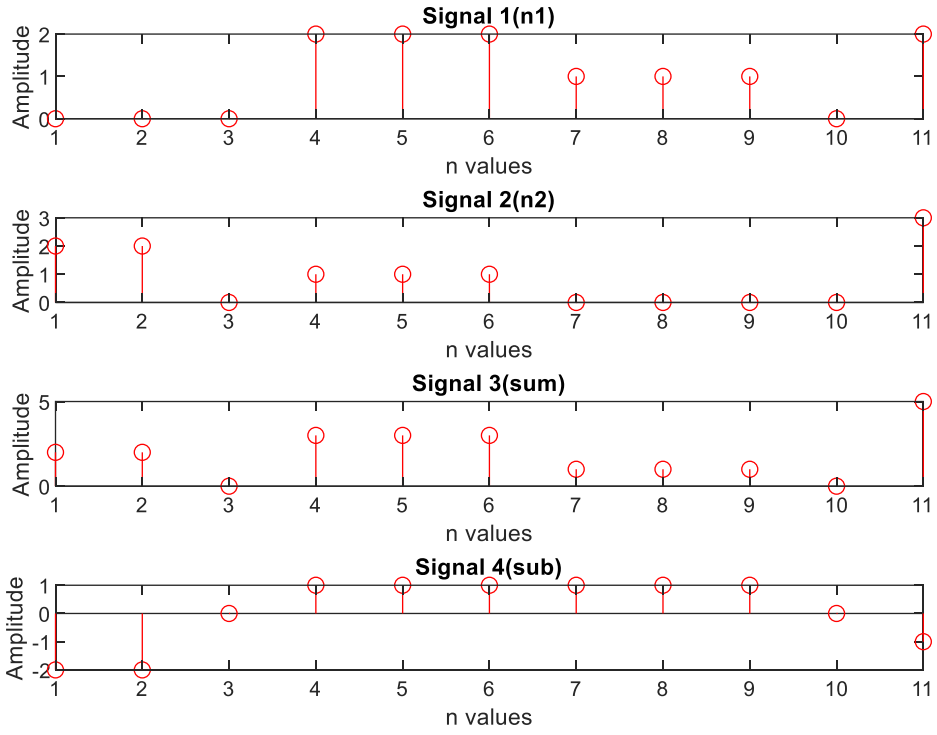

Figure 2.3: Graphical representation of summation and subtraction of two discrete signal

2.6.3 Code for drawing two given signals:

```
clc;

clear all;
close all;

x1 = [0 1 3 4];
y1 = [0 2 2 0];
subplot(2,1,1);
plot(x1, y1, 'r');

x2 = [0 1 2 3 5 6 7 8];
```

```
y2 = [0 1 1 2 2 1 1 0];
subplot(2,1,2);
plot(x2, y2, 'r');
```

Output:

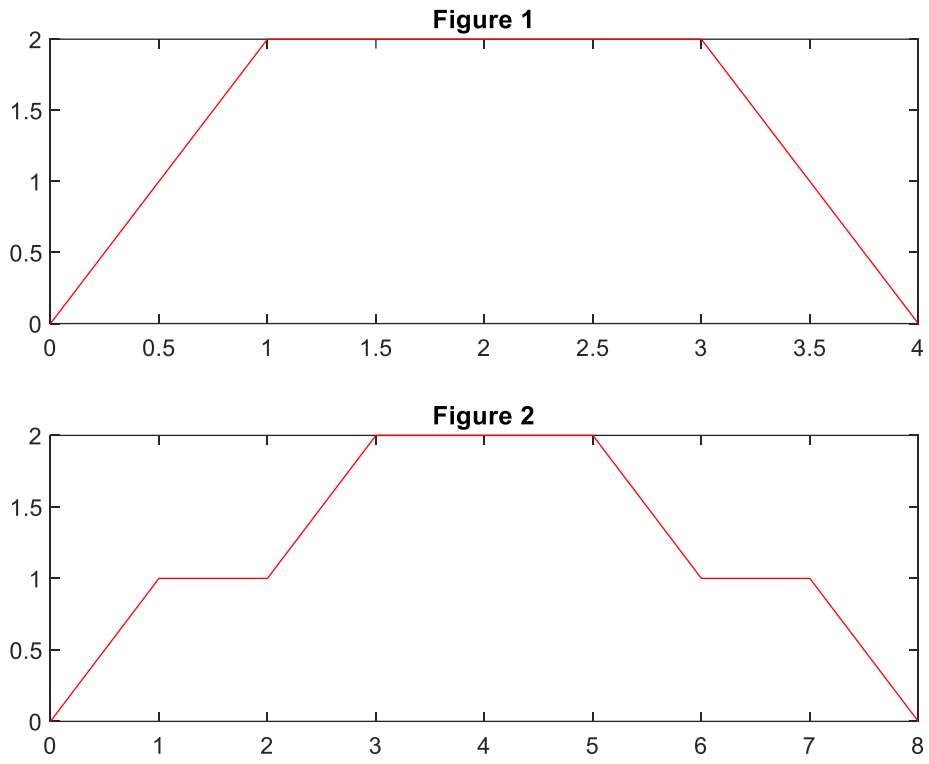

Figure 2.4: Output of two given figure

2.7 Discussion:

The experiment was about to implement circular convolution in MATLB without using built in cconv() function. It was implemented successfully using some for loop and mathematical logic.

Summation and subtraction of two discrete signal was performed simply using plus(+) and minus(-) operator and given two figures was drawn plotting sum points.

2.8 Conclusion:

The code was implemented successfully. The output of the code was similar to the theory.