Heaven's Light is Our Guide

Rajshahi University of Engineering& Technology

Department of Electrical & Computer Engineering

Course No: ECE 4123
Course Name: Digital Signal Processing

<u>Submitted by:</u>
Name: Sabiha Rubiatunnesa.
Roll: 1810007

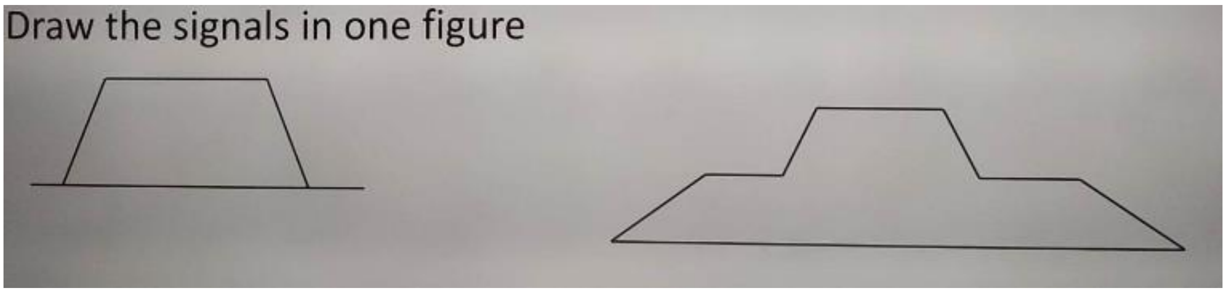
<u>Submitted to:</u>
Hafsa Binte Kibria
LECTURER
OF ECE, RUET

Experiment No: 02

Experiment Date: 03.05.2023

Experiment Name:

1. Implementation of circular convolution.
2. Addition and subtraction of 2 given discrete signals
3. Implementation of the following figure:



Theory:

Circular convolution, also known as cyclic convolution, is a special case of periodic convolution, which is the convolution of two periodic functions that have the same period. Periodic convolution arises, for example, in the context of the discrete-time Fourier transform (DTFT). Convolution via DFT is not exactly the same as linear convolution. It is called circular convolution. The convolution is circular because of the periodic nature of the DFT sequence. Recall that an N-point DFT of an aperiodic sequence is periodic with a period of N.

Addition and subtraction of two discrete signals result also discrete signal which we have implemented on the laboratory.

We have also implemented some codes to have some predefined figure.

Required software: Matlab

Code 1: Implementation of circular convolution

Code:

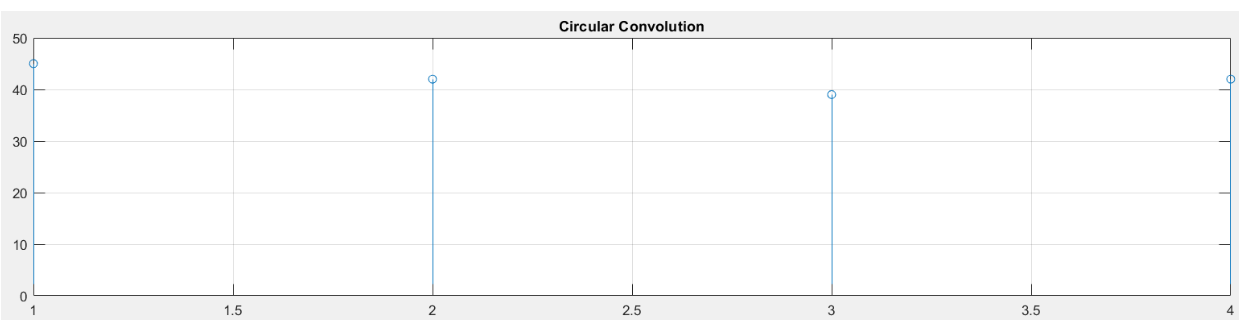
```
clc;
clear all;
close all;
x = input('Enter signal 1: ');
h = input('Enter signal 2: ');
N1 = length(x);
N2 = length(h);
N = max(N1,N2);
N3 = N1 - N2;
if(N3 > 0)
```

```
    h = [h, zeros(1, N3)];
else
    x = [x, zeros(1, -N3)];
end
for n= 1:N;
    y(n) = 0;
    for i=1:N;
        j = n-i+1;
        if(j<=0)
            j = N+j;
        end
        y(n) = [y(n)+ (x(i)*h(j))];
    end
end
disp('Result ');
y
subplot(2,1,1)
stem(y);
title('Circular Convolution')
grid on;
```

Output:

```
Enter signal 1: [1 2 2 1]
Enter signal 2: [5 6 8 9]
Result

y =

    45    42    39    42
```

Figure:



Code 2 : Addition and subtraction of 2 given discrete signals

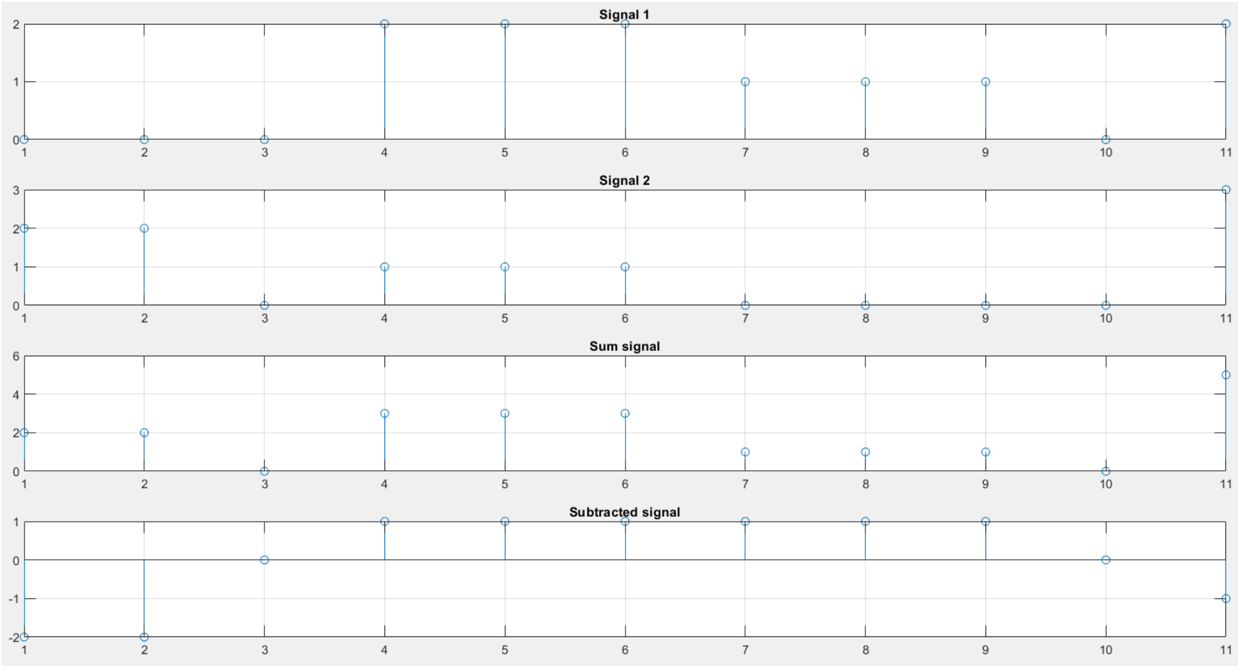
Code:

```
clc;
clear all;
close all;
t = 0:.1:12;
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
figure(1)
subplot(4,1,1);
stem(n1);
title('Signal 1')
grid on;

subplot(4,1,2);
stem(n2);
title('Signal 2')
grid on;

subplot(4,1,3);
stem(sum);
title('Sum signal')
grid on;
subplot(4,1,4);
stem(sub);
title('Subtracted signal')
grid on;
```

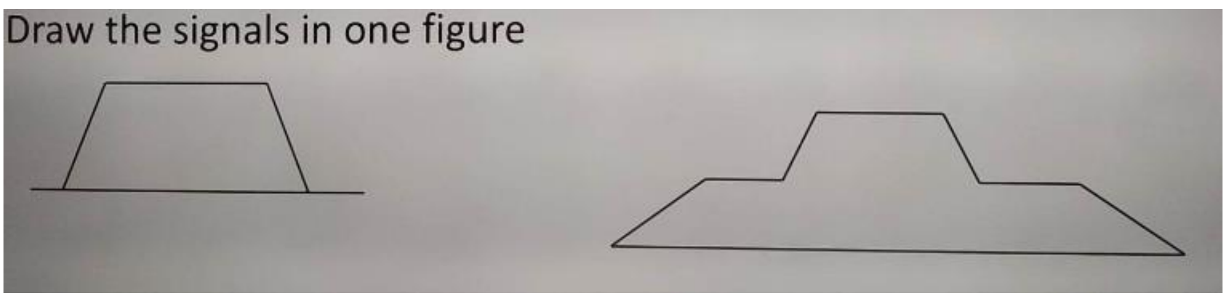
Output:

```
sum =

     2     2     0     3     3     3     1     1     1     0     5


sub =

    -2    -2     0     1     1     1     1     1     1     0    -1
```

Figure:



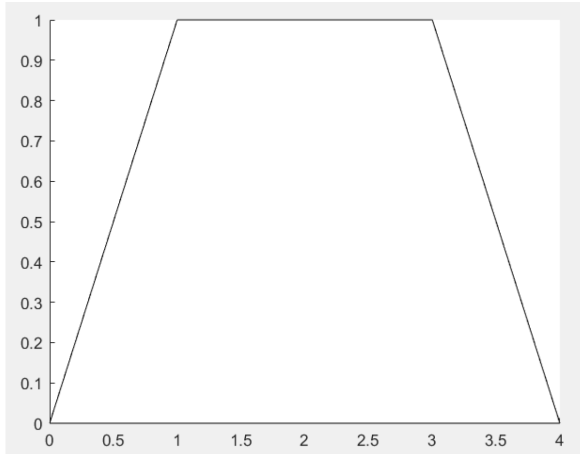
Code 3: Implementation of the following figure:



Code for 1st figure:

```
clc;
clear all;
close all;
h = 1;
a = 2;
b = 4;

A = [0 0];
B = [b 0];
C = [0.5*(b-a)+a h];
D = [0.5*(b-a) h];
corr = [A ; B ; C ; D];
patch(corr(:,1), corr(:,2), 'w')
```

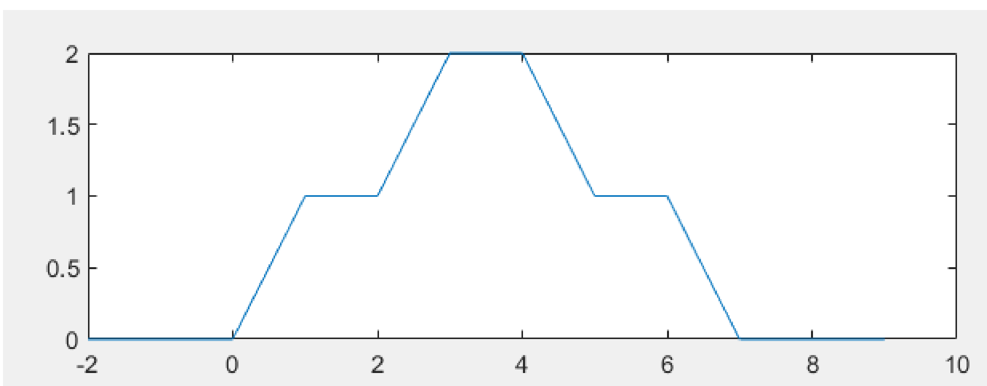
Figure:



Code for 2nd figure:

```
clc;
clear all;
close all;
u = @(t) t>=0;
x=@(t) t.*u(t)-(t-1).*u(t-1)+(t-2).*u(t-2)-(t-3).*u(t-3)-(t-
4).*u(t-4)+(t-5).*u(t-5)-(t-6).*u(t-6)+(t-7).*u(t-7);
t= linspace(-2,9,1000);
subplot(2,1,2)
plot(t,x(t))
```

Figure:



Conclusion:

The experiment was done successfully as we have achieved the expected output which matches theoretical analysis.