

DEPARTMENT OF
ROBOTICS AND MECHATRONICS ENGINEERING

LAB REPORT

DIGITAL SIGNAL PROCESSING
(CSE-401)

Submitted By:

Md. Tahmeed Abdullah

Roll: SH-092-002

4th year 1st semester

Submitted To:

Mr. Sujan Sarker

Lecturer

Dept. of RME

Experiment no. 3

Name of the experiment

Convolution of a digital signal with a given filter signal.

Objectives

- To learn how to use linear filters.
- To understand the basics of convolution.
- implementing linear convolution using MATLAB

Theory

In mathematics convolution is a mathematical operation on two functions (f and g) to produce a third function that expresses how the shape of one is modified by the other.

An input signal, $x[n]$, enters a linear system with an impulse response, $h[n]$, resulting in an output signal, $y[n]$. In equation form: $y[n] = x[n] * h[n]$. Expressed in words, the input signal convolved with the impulse response is equal to the output signal.

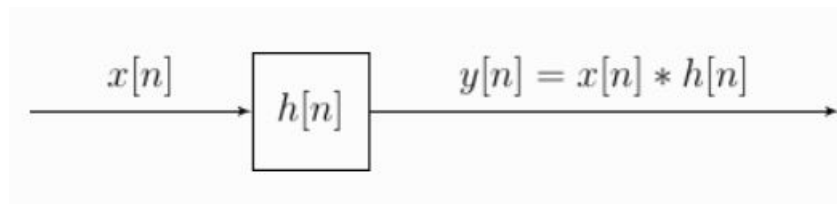


Figure 1: Convolution Block Diagram

The mathematical definition of convolution is:

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m]$$

To perform discrete time linear convolutions we need to follow the following steps:

- Select the filter signal and flip the signal

- Pad the signal to be convolved, with zeros to sufficient lengths
- Align the right most element of the filter with the left most element of the signal to be convolved.
- Perform cross multiplication.
- Apply 1 unit right shift on the filter signal.
- Perform previous two steps until no element of the original signal is left.

Implementation Code

main.m

```

1  clc; clear; close all;
2  h = [ 1 2 1 -1 ]; ho = 2;
3  x = [ 1 2 3 1 ];  xo = 1;
4
5  [out,outOrigin] = convolution(x,h,xo,ho);

```

Functions Used:

convolution.m

```

1  function [out,outOrigin] = convolution(input, h, ino
   , ho)
2      hneg = fliplr(h);
3      L = size(input,2);
4      ho = L-ho+1;
5      X = zeros(1, 3*L);
6      H=X;
7      out = X;
8      for i=1:size(input,2)
9          H(i) = hneg(i);
10     end
11     for i =L+1:2*L
12         X(i) = input(i-L);
13     end
14     for i =1:size(X,2)
15         out(1,i) = Ws(X,H);
16         H = rightShift(H);
17     end
18     out = out(2:size(out,2)-L);
19     outOrigin = L-ho;
20 end

```

rightShift.m

```
1 function [outSignal] = rightShift(inSignal,inOrigin
2 )
3     l = size(inSignal,2);
4     outSignal = zeros(1,l);
5
6     for i = 2:l
7         outSignal(i) = inSignal(i-1);
8     end
9 end
```

Ws.m

```
1 function out = Ws(a,b)
2     s=0;
3     for j = 1:size(b,2)
4         s = s+ a(1,j)*b(1,j);
5     end
6     out=s;
7 end
```