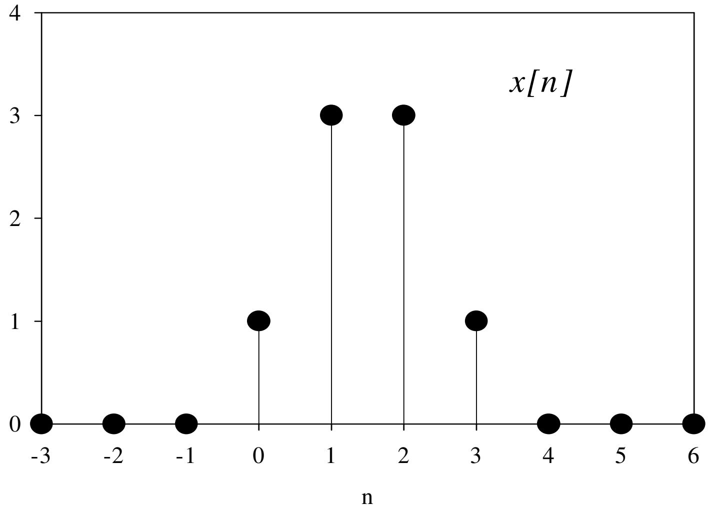
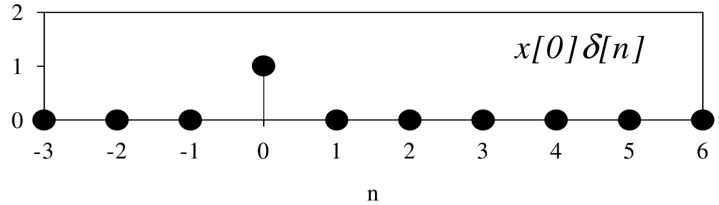
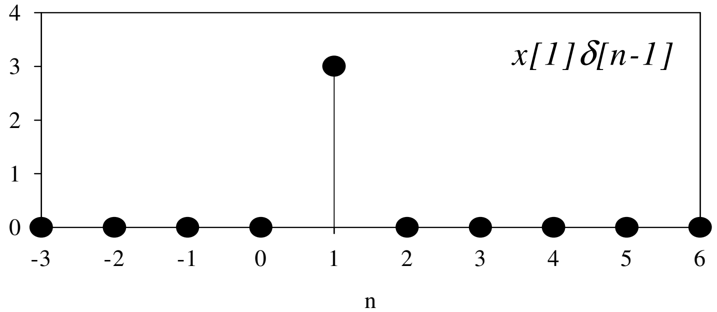
Lecture content

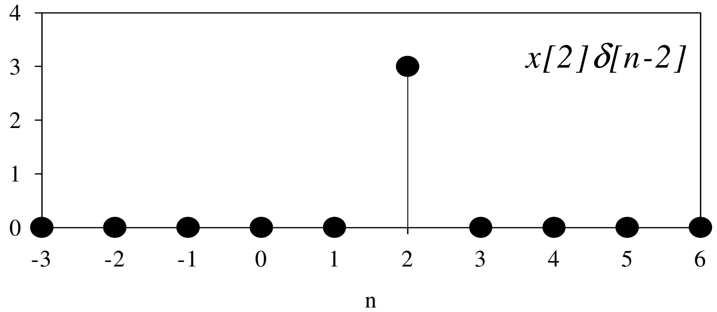
- Discrete Time Convolution
- Derivation of convolution sum for DT signals
- Convolution procedures for DT signals

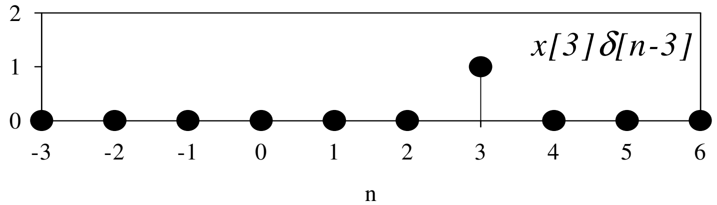
Convolution of discrete time signals











$$x[n] = \ldots + x[0]\delta[n] + x[1]\delta[n-1] + x[2]\delta[n-2] + x[3]\delta[n-3] + \ldots\ldots$$

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

This property is the ***sifting property*** of DT unit impulse.

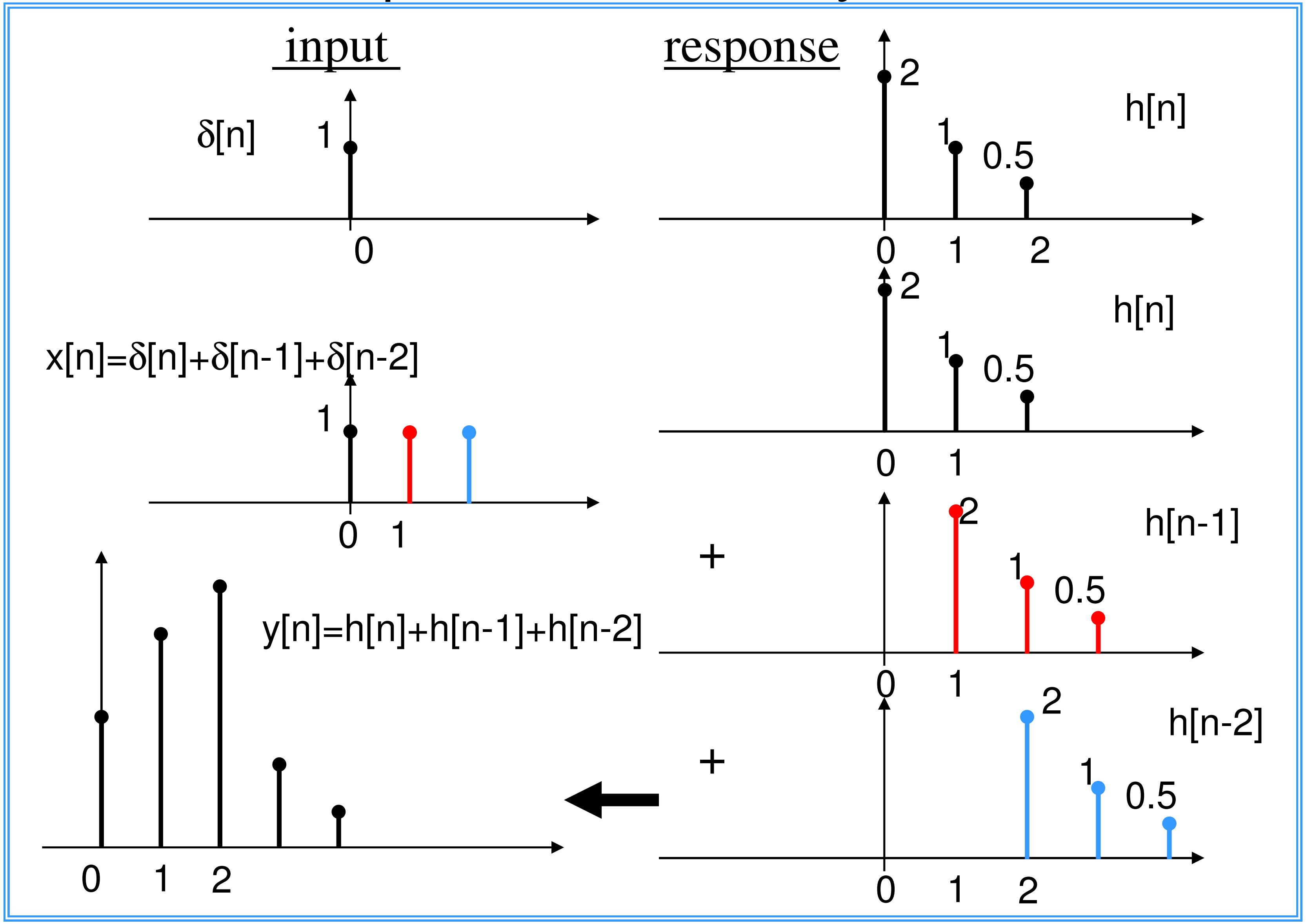
Convolution of discrete time signals

A DT signal $x[n]$ can be represented as a superposition of scaled versions of shifted impulse $\delta[n\text{-}k]$.

i.e: $x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$

We shall now show that it is possible to compute the LTI system response to any input if the impulse response is known.

Response of an LTI system



Response of an LTI system

	0	1	2	3	4	5
δ[n]	1	0	0	0	0	0
h[n]	2	1	0.5	0	0	0
x[n]	1	1	1	0	0	0
h[n]	2	1	0.5	0	0	0
h[n-1]	0	2	1	0.5	0	0
h[n-2]	0	0	2	1	0.5	0
y[n]	2	3	3.5	1.5	0.5	0

This is not efficient because all values of $h[n-k]$ need to be stored to compute $y[n]$.

Convolution sum

Input		**Response**	
$\delta[n]$	$\rightarrow$	$h[n]$	(definition),
$\delta[n\text{-}k]$	$\rightarrow$	$h[n\text{-}k]$	(time shifting),
$x[k]\delta[n\text{-}k]$	$\rightarrow$	$x[k]h[n\text{-}k]$	(homogeneity),
$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$	$\rightarrow$	$\sum_{k=-\infty}^{\infty} x[k]h[n-k]$	(additivity)

Thus, the response of the LTI system to an input $x[k]$ is

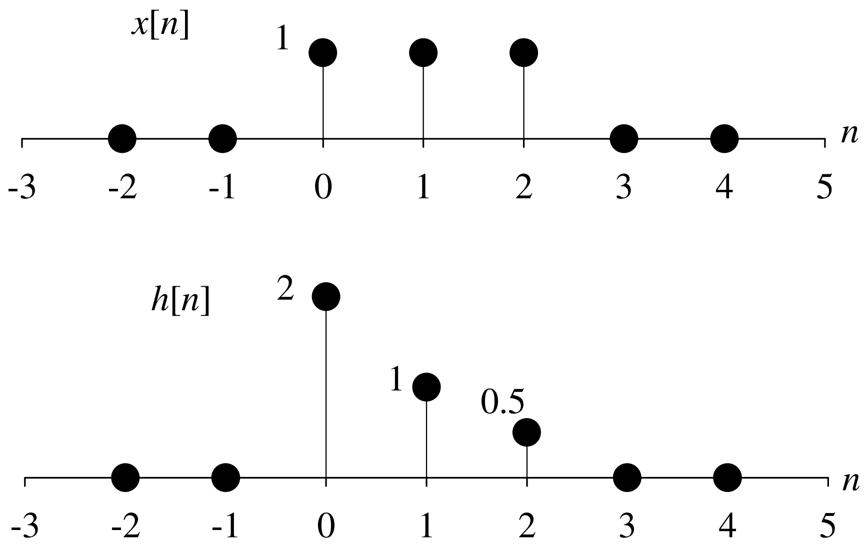
$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] .$$

This result is referred to as the ***convolution sum*** and the operation on the right hand side is called the ***discrete convolution*** of the sequences $x[n]$ and $h[n]$, which is usually represented symbolically as

$$y[n] = x[n] * h[n].$$

Convolution of discrete time signals

- Consider an LTI system with impulse response $h[n]$ and input $x[n]$ shown below



The procedures to compute $y[n]$ are:

1) Replace the variable n with k.
2) Flipping $h[k]$ with respect to $k = 0$ to obtain $h[-k]$.
3) Shifting $h[-k]$ to n to give $h[n-k]$.
4) Multiply $h[n-k]$ and $x[k]$ for all k.
5) Summing all non-zero product of $h[n-k]x[k]$ to yield $y[n]$.

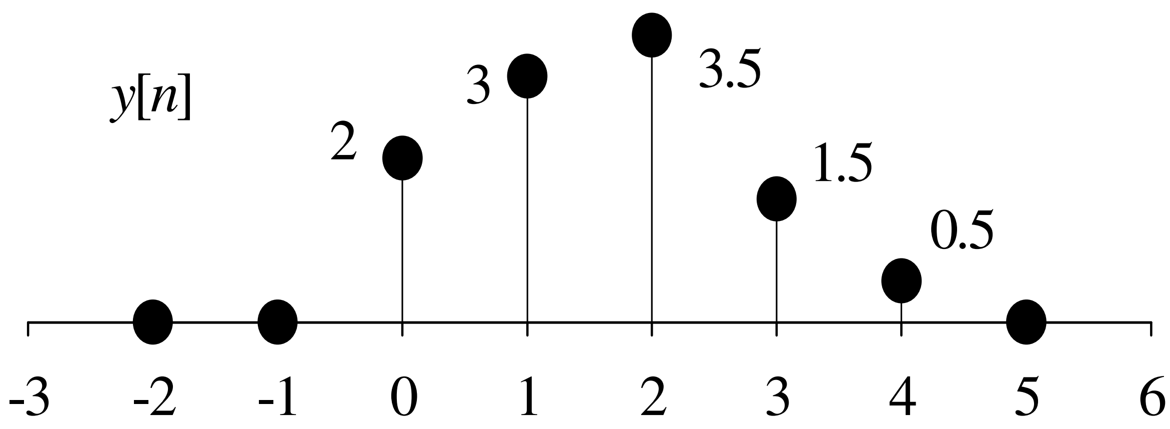
Convolution of discrete time signals

- The procedures can be outlined in a table form as:

	k	-2	-1	0	1	2	3	4	$y[n]=\Sigma h[n-k]x[k]$
	$x[k]$	0	0	1	1	1	0	0	
	$h[k]$	0	0	2	1	0.5	0	0	
$n = 0$	$h[-k]$								
$n = 1$	$h[1-k]$								
$n = 2$	$h[2-k]$								
$n = 3$	$h[3-k]$								
$n = 4$	$h[4-k]$								

Convolution of discrete time signals

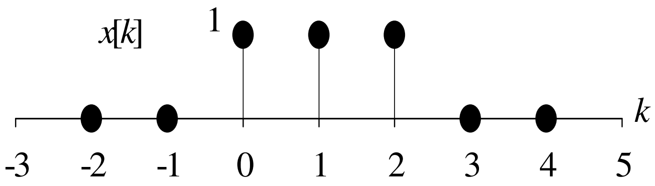
$y[n] = x[n] * h[n]$

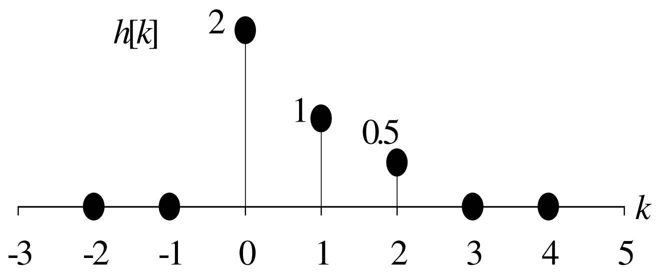


Discrete convolution: graphical method

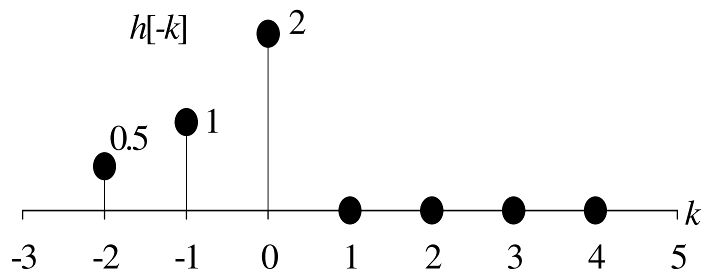
We can also perform this discreet time convolution graphically.

Step 1: Replace the variable n with k.





Step 2: Flipping $h[k]$ with respect to $k = 0$ to obtain $h[-k]$.



Discrete convolution: graphical method

- Step 3: Shifting $h[-k]$ to n to give $h[n-k]$.
- Step 4 and 5:

Find the products $h[n-k]x[k]$ for all k and summing all the non-zero products .

$y[0] = h[0]x[0] = 2\times1 = 2$

$y[1] = h[1]x[0] + h[0]x[1] = (1\times1)+(2\times1) = 3$

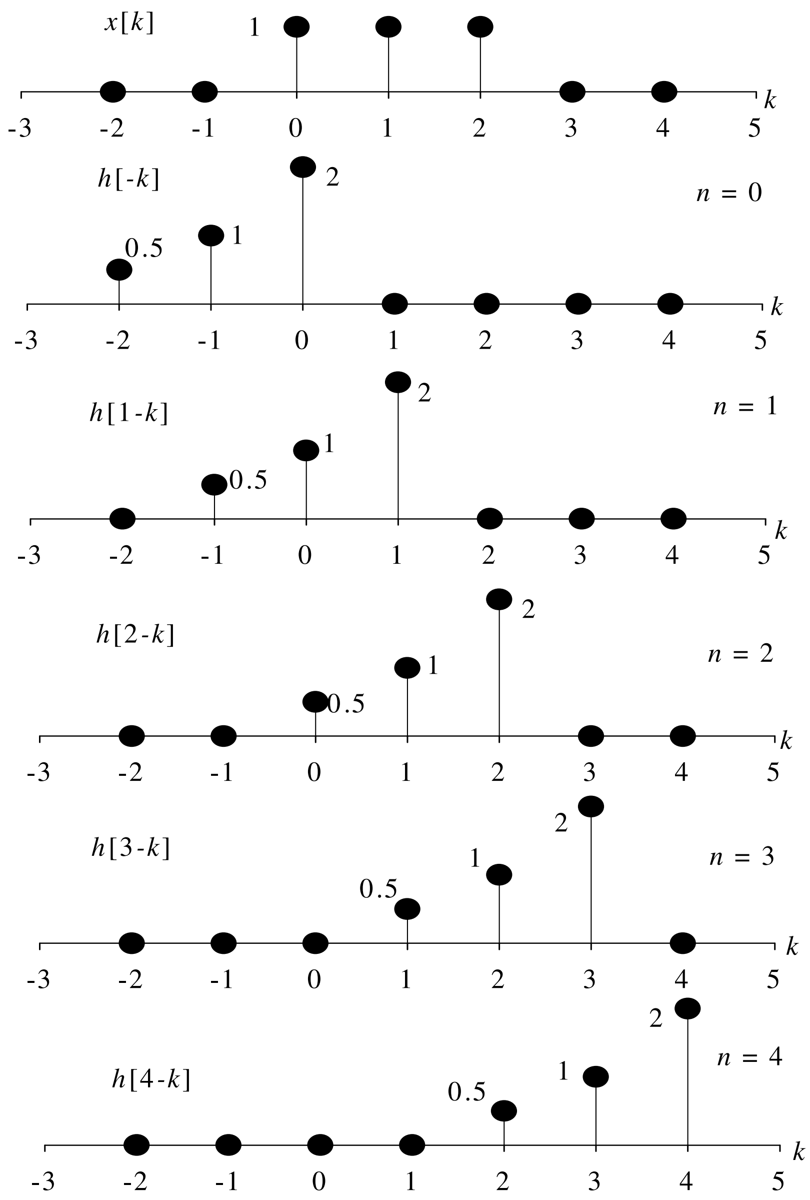
$y[2] = h[2]x[0] + h[1]x[1] + h[0]x[2]$

$\quad = (0.5\times1) + (1\times1) + (2\times1) = 3.5$

$y[3] = h[2]x[1] + h[1]x[2]$

$\quad = (0.5\times1) + (1\times1) = 1.5$

$y[4] = h[3]x[2] = (0.5\times1) = 0.5$



1. Consider the two sequences:

$$x[n] = \begin{cases} 1, & 0 \le n \le 3 \\ 0, & otherwise \end{cases}$$

and

$$h[n] = \begin{cases} n, & 0 \le n \le 4 \\ 0, & otherwise \end{cases}$$

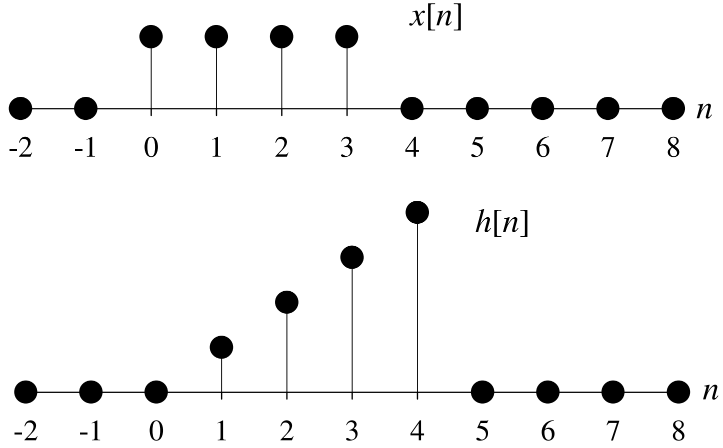
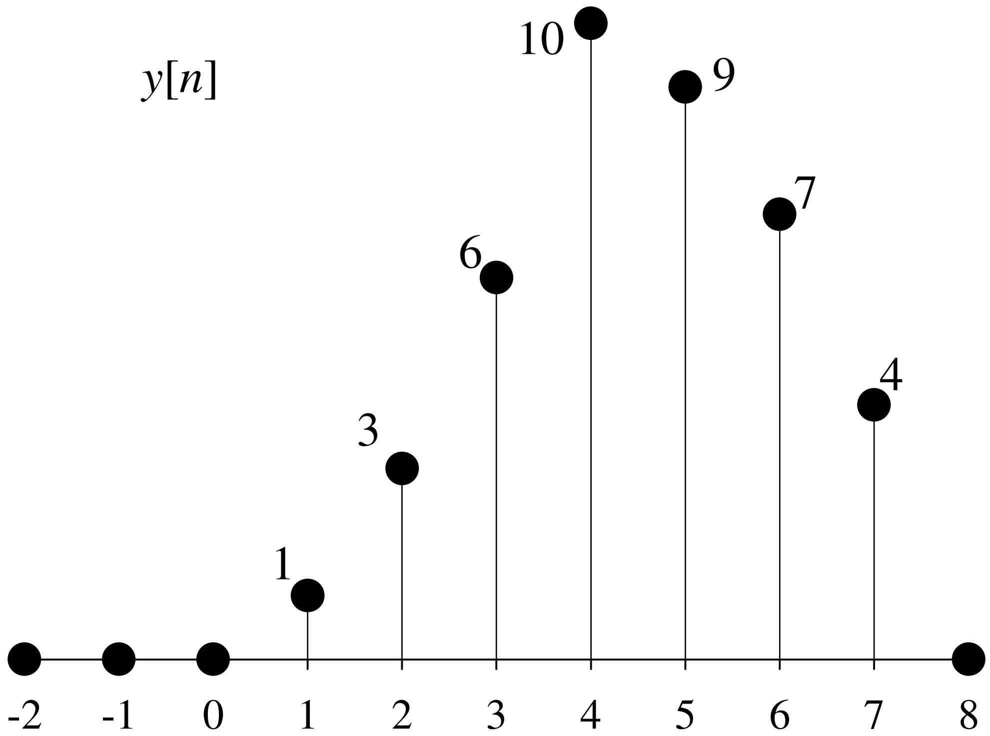


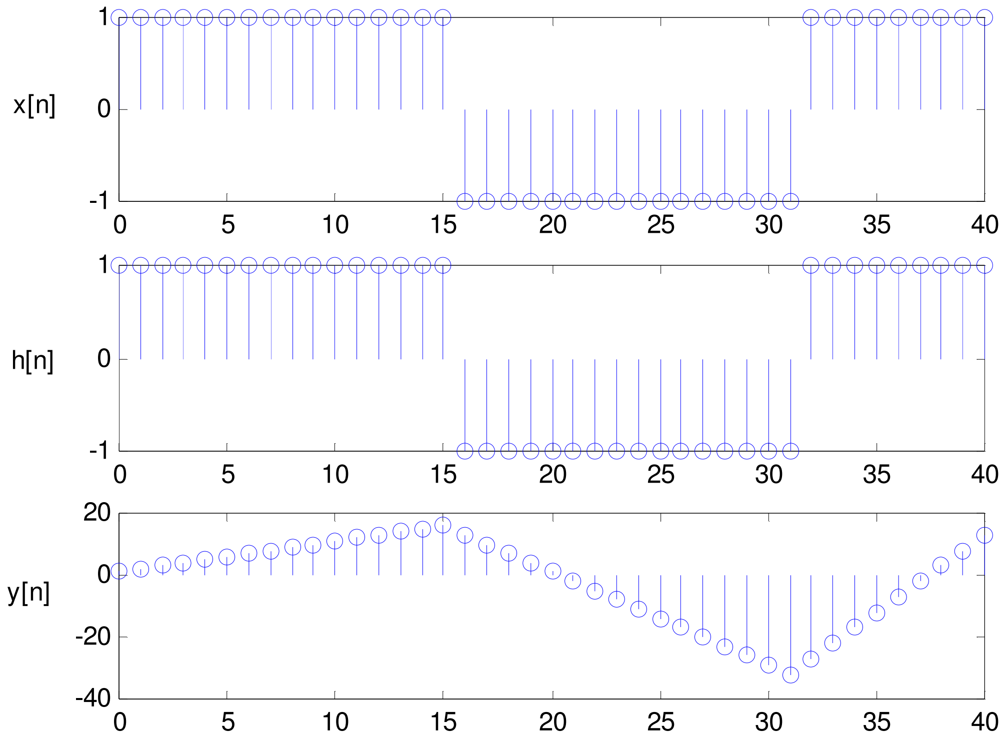
Figure 3.7: The signals x[n] and h[n] to be convolved.

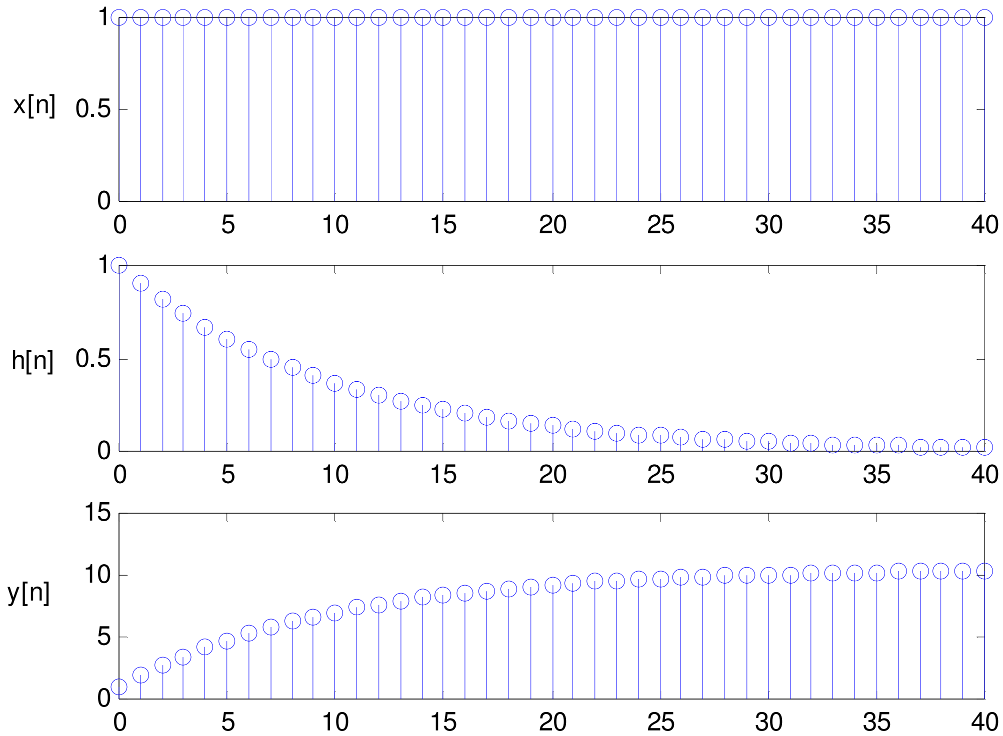
	k	-2	-1	0	1	2	3	4	5	6	7
	$x[k]$	0	0	1	1	1	1	0	0	0	0
	$h[k]$	0	0	0	1	2	3	4	0	0	0
$n=0$	$h[-k]$										
$n=1$	$h[1-k]$										
$n=2$	$h[2-k]$										
$n=3$	$h[3-k]$										
$n=4$	$h[4-k]$										
$n=5$	$h[5-k]$										
$n=6$	$h[6-k]$										
$n=7$	$h[7-k]$										

	$\Sigma h[n-k]x[k]$
$n=0$	
$n=1$	
$n=2$	
$n=3$	
$n=4$	
$n=5$	
$n=6$	
$n=7$	



$y[n]$
10
9
7
6
4
3
1
-2
-1
0
1
2
3
4
5
6
7
8


x[n]
h[n]
y[n]


x[n]
h[n]
y[n]
0
0.5
1
5
10
15
0
5
10
15
20
25
30
35
40