BINGHAMTON UNIVERSITY
State University of New York

EECE 301
Signals & Systems

Prof. Mark Fowler

Note Set #31

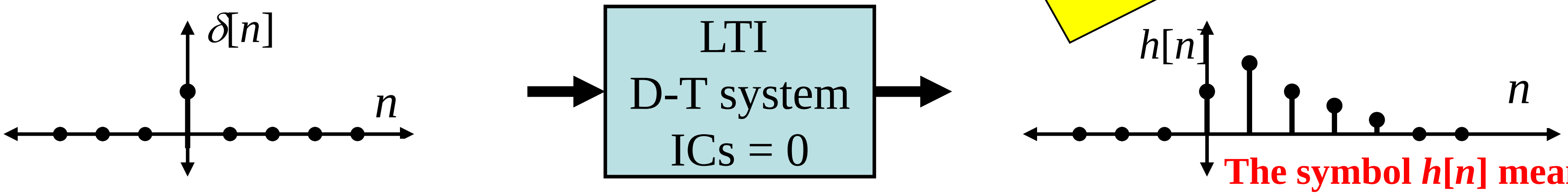
- D-T Convolution: The Tool for Finding the Zero-State Response

Recall: Impulse Response

Earlier we introduced the concept of impulse response…

…what comes out of a system when the input is an impulse (delta sequence)

Note: If system is causal, then $h[n] = 0$ for $n < 0$



The symbol $h[n]$ means "the impulse response".

Noting that the ZT of $\delta[n] = 1$ and using the properties of the transfer function and the Z transform we said that

$$h[n] = Z^{-1}\{H(z)Z\{\delta[n]\}\}$$

$$h[n] = Z^{-1}\{H(z)\}$$

$$h[n] = IDTFT\{H(\Omega)\}$$

So…once we have either H(z) or $H(\Omega)$ we can get the impulse response $h[n]$

Since H(z) & $H(\Omega)$ describe the system so must the impulse response $h[n]$

How???

Convolution Property and System Output

$x[n] \leftrightarrow X(\Omega)$

$x[n] \leftrightarrow X(z)$

Let $x[n]$ be a signal with DTFT $X(\Omega)$ and ZT of $X(z)$

$h[n] \leftrightarrow H(\Omega)$

$h[n] \leftrightarrow H(z)$

Consider a system w/ freq resp $H(\Omega)$ & trans func $H(z)$

We've spent much time using these tools to analyze system outputs this way:

$$Y(\Omega) = H(\Omega)X(\Omega) \quad \leftrightarrow \quad y[n] = DTFT^{-1}\left\{H(\Omega)X(\Omega)\right\}$$

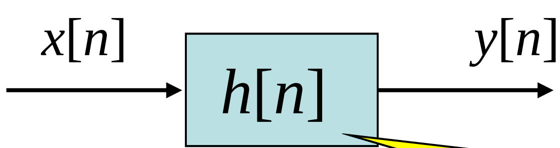
$$Y(z) = H(z)X(z) \quad \leftrightarrow \quad y[n] = Z^{-1}\left\{H(z)X(z)\right\}$$

The convolution property of the DTFT and ZT gives an alternate way to find $y[n]$:

$$DTFT^{-1}\left\{X(\Omega)H(\Omega)\right\} = x[n] * h[n]$$

$$Z^{-1}\left\{X(z)H(z)\right\} = x[n] * h[n]$$

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



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

"Convolving" input $x[n]$ with the impulse response $h[n]$ gives the output $y[n]$!

LTI System with impulse response $h[n]$

<u>Convolution for *Causal* System & with *Causal* Input</u>

An arbitrary LTI system's output can be found using the general convolution form:

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

General LTI System

If the system is causal then $h[n] = 0$ for $n < 0$… Thus $h[n - m] = 0$ for $m > n$ … so:

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

Causal LTI System

If the input is causal then $x[n] = 0$ for $n < 0$…. so:

$$y[n] = \sum_{m=0}^{\infty} x[m]h[n-m]$$

Causal Input & General LTI System

If the system & signal are both causal then

$$y[n] = \sum_{m=0}^{n} x[m]h[n-m]$$

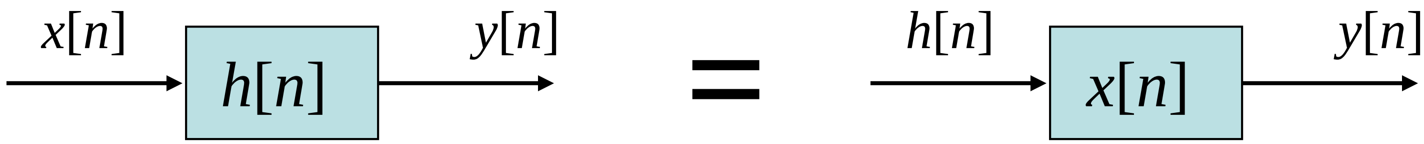
Causal Input & Causal LTI System

Convolution Properties (can sometimes exploit to make things easier)

1. Commutativity

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

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



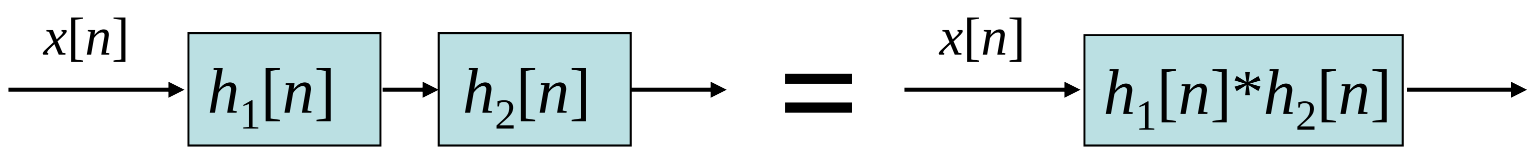
This is obvious from the frequency domain (or z domain) viewpoint:

$$x[n]*h[n]=h[n]*x[n] \Rightarrow X(\Omega)H(\Omega)=H(\Omega)X(\Omega)$$

2. Associativity

$$(x[n]*h_1[n])*h_2[n]=x[n]*(h_1[n]*h_2[n])$$

$\Rightarrow$ Can combine cascade into single equivalent system

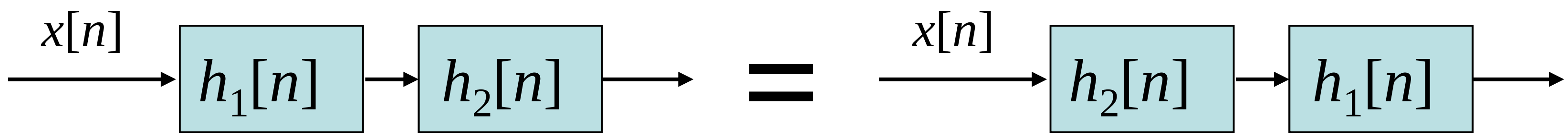


This is obvious from the frequency domain (or z domain) viewpoint:

$$[X(\Omega)H_1(\Omega)]H_2(\Omega)=X(\Omega)[H_1(\Omega)H_2(\Omega)]$$

Tells us what the Freq Resp is for a cascade

Associativity together with commutativity says we **can interchange the order of two cascaded systems**:

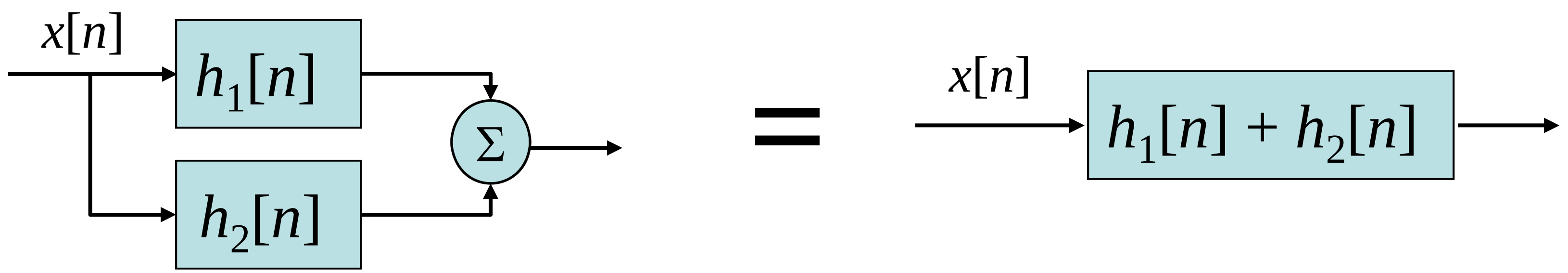


Warning: This holds in theory but in practice there may be physical issues that prevent this!!!

3. Distributivity

$$x[n]*(h_1[n]+h_2[n]) = x[n]*h_1[n] + x[n]*h_2[n]$$

$\Rightarrow$ can combine sum of two outputs into a single system (or vice versa)



With commutativity this says we can split a complicated input into sum of simple ones… which is nothing more than “linearity”!!

<u>Graphical Convolution – To Visualize & Test Real Systems</u>

Can do convolution this way when signals are know numerically or by equation

- Convolution involves the sum of a product of two signals: $x[i]h[n-i]$

- At each output index *n*, the product changes

"Commutativity" says we can flip either $x[i]$ or $h[i]$ and get the same answer

Step 1: Write both as functions of *i*: $x[i]$ & $h[i]$

Step 2: Flip $h[i]$ to get $h[-i]$ (The book calls this "fold")

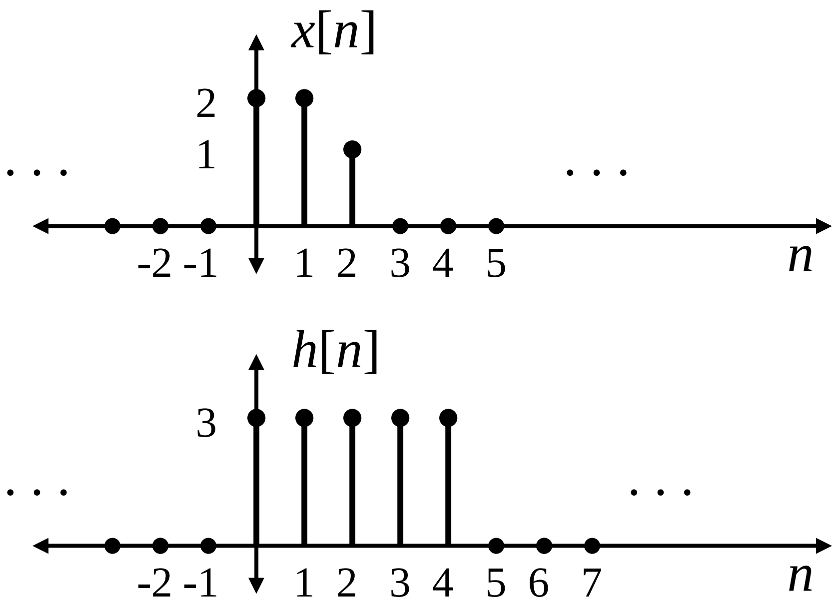
Repeat for each *n*

Step 3: For each output index *n* value of interest, shift by *n* to get $h[n - i]$

(Note: positive *n* gives right shift!!!!)

Step 4: Form product $x[i]h[n - i]$ and sum its elements to get the number $y[n]$

Example of Graphical Convolution



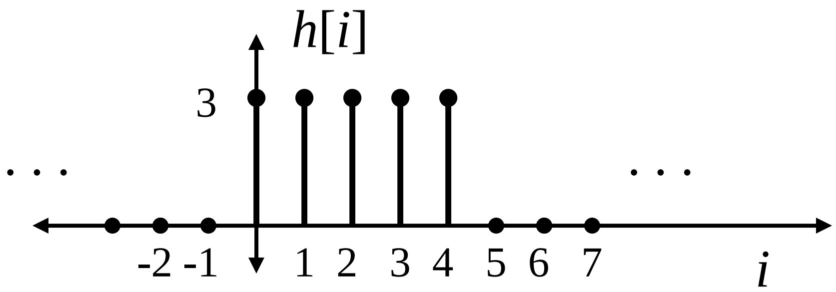
Find $y[n]=x[n]*h[n]$ for all integer values of n

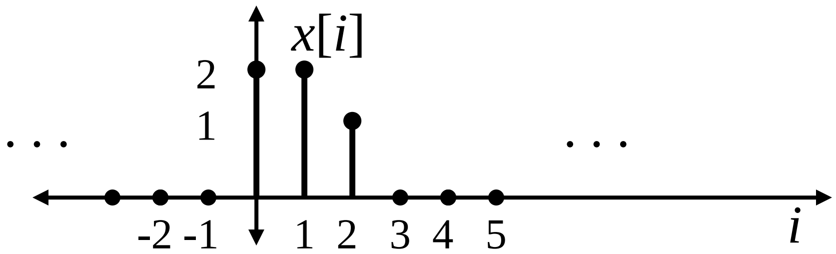
Solution

For this problem I choose to flip $x[n]$

My personal preference is to flip the shorter signal although I sometimes don't follow that "rule"… only through lots of practice can you learn how to best choose which one to flip.

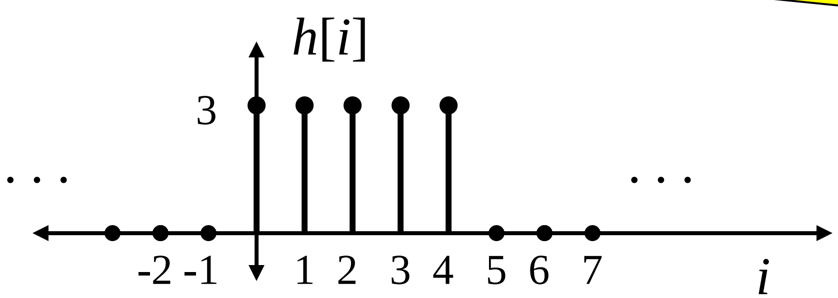
Step 1: Write both as functions of i: $x[i]$ & $h[i]$

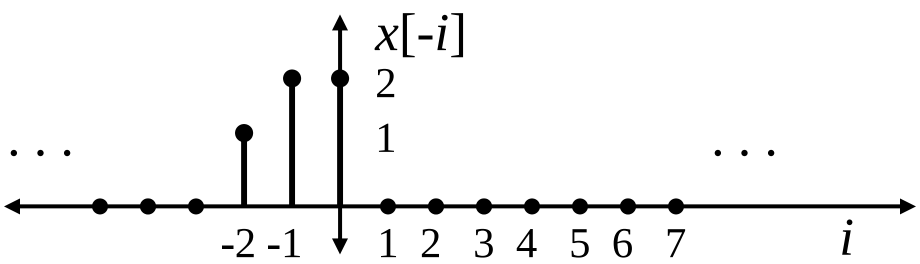




Step 2: Flip $x[i]$ to get $x[-i]$

"Commutativity" says we can flip either $x[i]$ or $h[i]$ and get the same answer…
Here I flipped $x[i]$



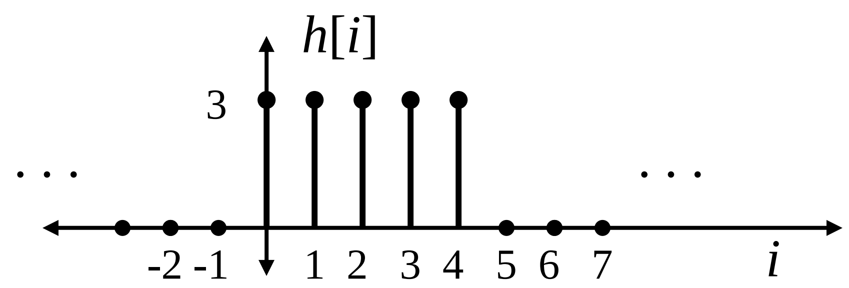


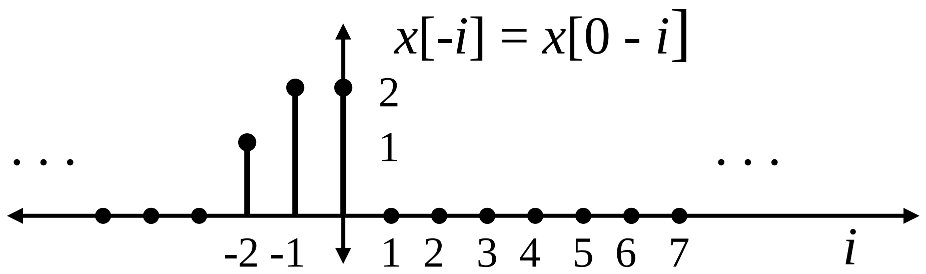
We want a solution for $n = \ldots$ -2, -1, 0, 1, 2, … so must do Steps 3&4 for all n.

But… let's first do: **Steps 3&4 for $n = 0$** and then proceed from there.

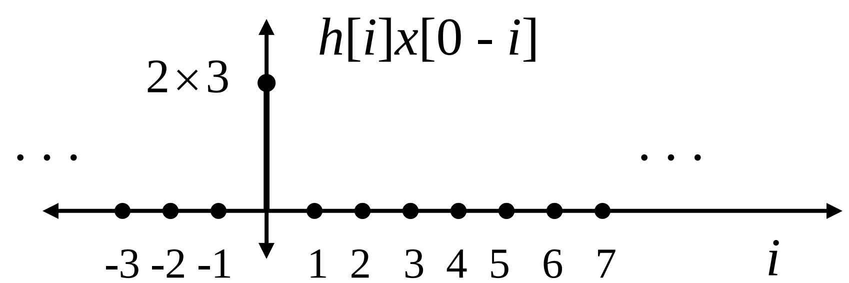
Step 3: For **$n = 0$**, shift by n to get $x[n - i]$

For $n = 0$ case there is no shift!
$x[0 - i] = x[-i]$





Step 4: For **$n = 0$**, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$



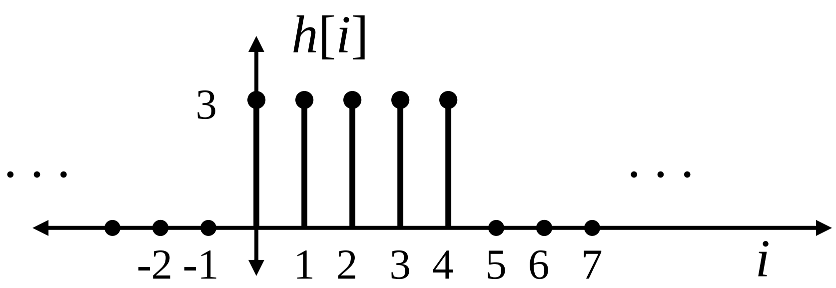
Sum over $i \Rightarrow$

$y[0] = 6$

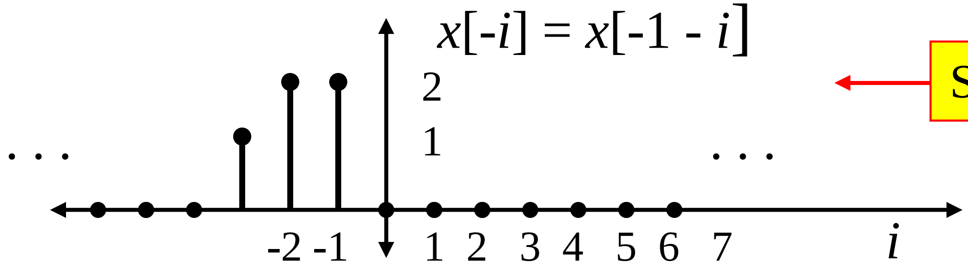
Steps 3&4 for all $n < 0$

Step 3: For $n < 0$, shift by n to get $x[n - i]$

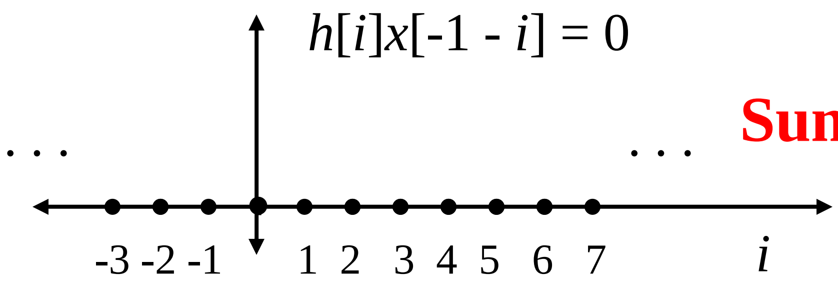
Negative n gives a left-shift



Shown here for $n = -1$



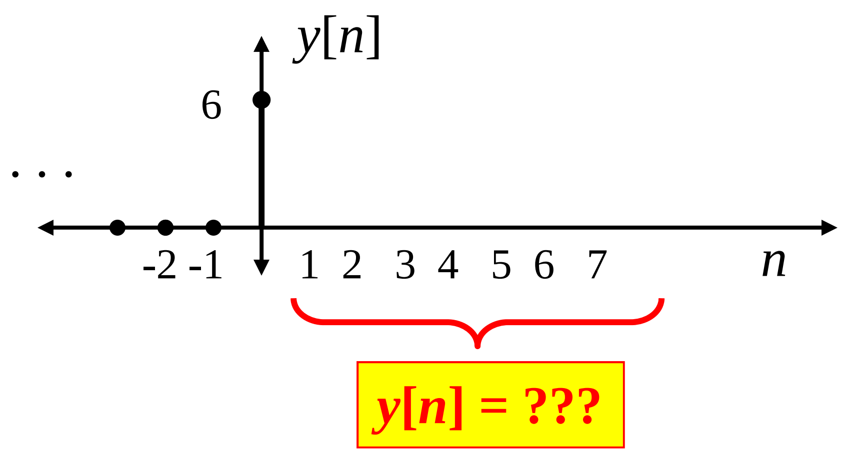
Step 4: For $n < 0$, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$



Sum over i $\Rightarrow$ $y[n] = 0 \quad \forall n < 0$

So… what we know so far is that:

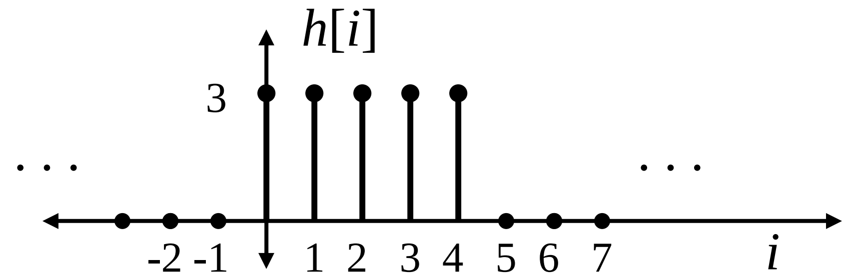
$$y[n] = \begin{cases} 0, & \forall n < 0 \\ 6, & n = 0 \end{cases}$$

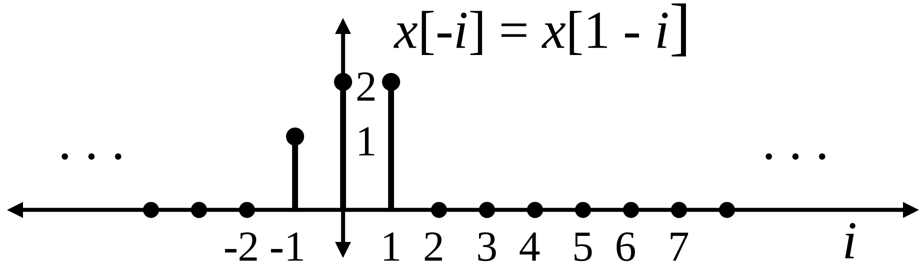


So now we have to do Steps 3&4 for $n > 0$…

Steps 3&4 for $n = 1$

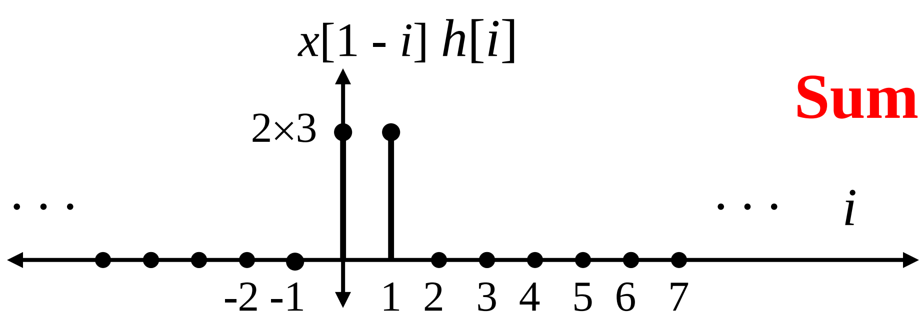
Step 3: For $n = 1$, shift by n to get $x[n - i]$ ← Positive n gives a Right-shift





shifted to the right by one

Step 4: For $n = 1$, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$

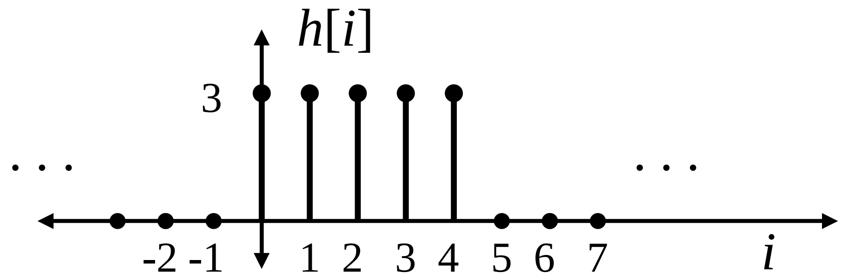


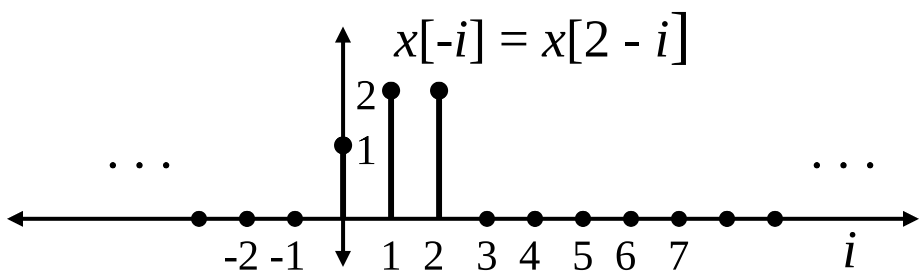
Sum over i ⇒ $y[1] = 6 + 6 = 12$

Steps 3&4 for $n = 2$

Step 3: For $n = 2$, shift by n to get $x[n - i]$

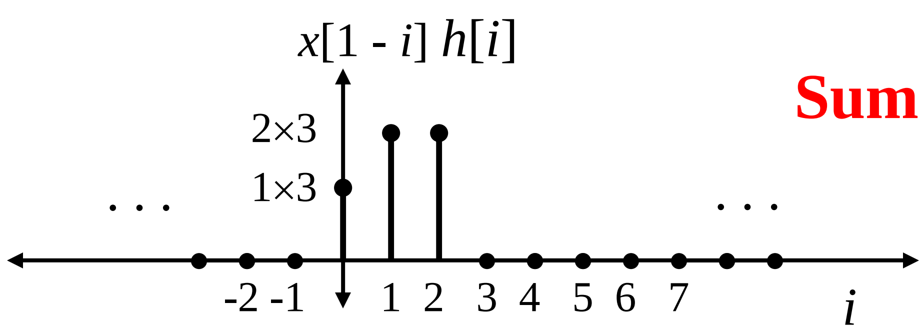
Positive n gives a Right-shift





shifted to the right by two

Step 4: For $n = 2$, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$

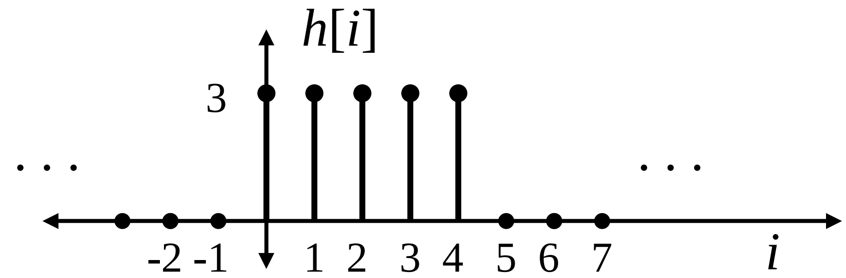


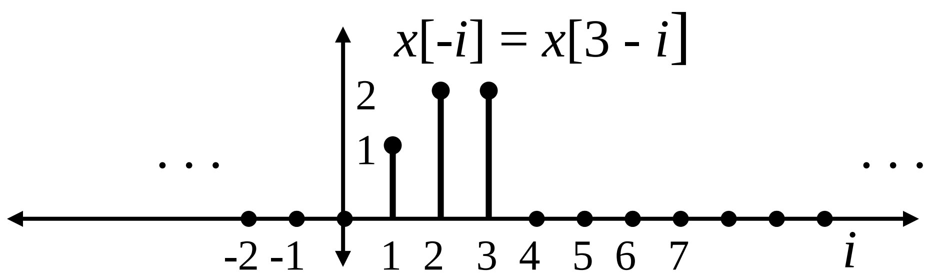
Sum over i ⇒ $y[2] = 3 + 6 + 6 = 15$

Steps 3&4 for $n = 3$

Step 3: For $n = 3$, shift by n to get $x[n - i]$

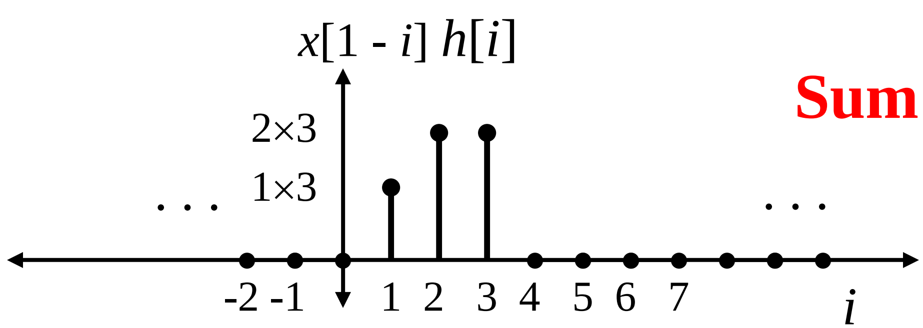
Positive n gives a Right-shift





shifted to the right by three

Step 4: For $n = 3$, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$

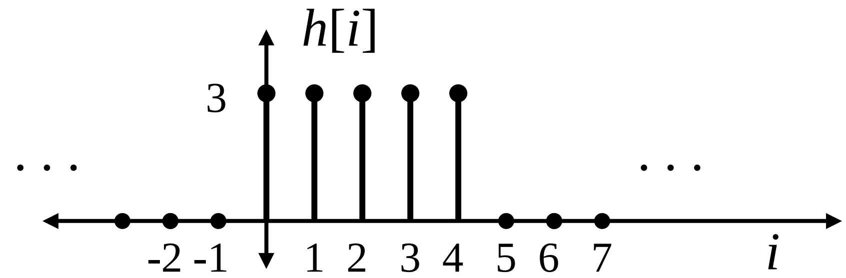


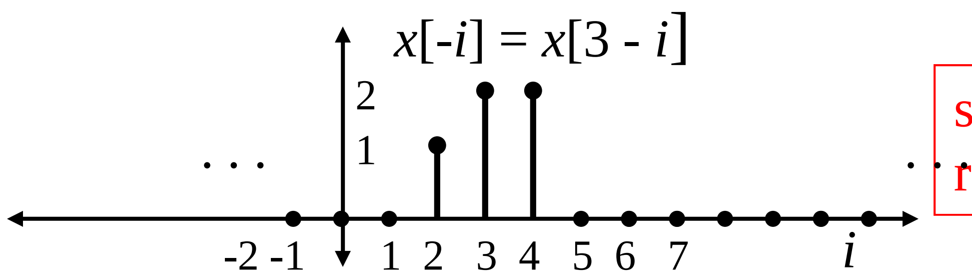
Sum over i $\Rightarrow$ $y[3] = 3 + 6 + 6 = 15$

Steps 3&4 for $n = 4$

Step 3: For $n = 4$, shift by n to get $x[n - i]$

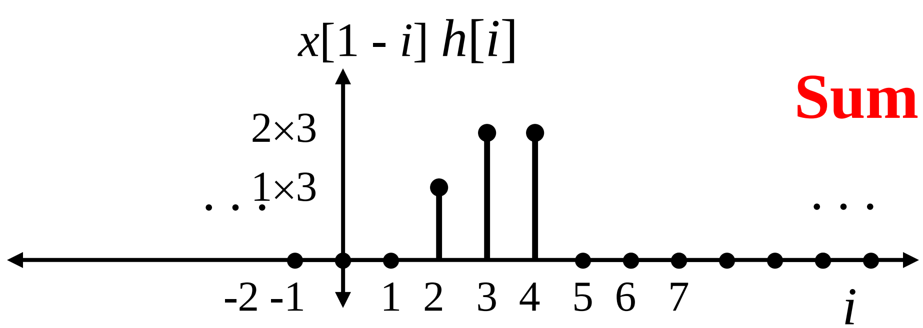
Positive n gives a Right-shift





shifted to the right by four

Step 4: For $n = 4$, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$

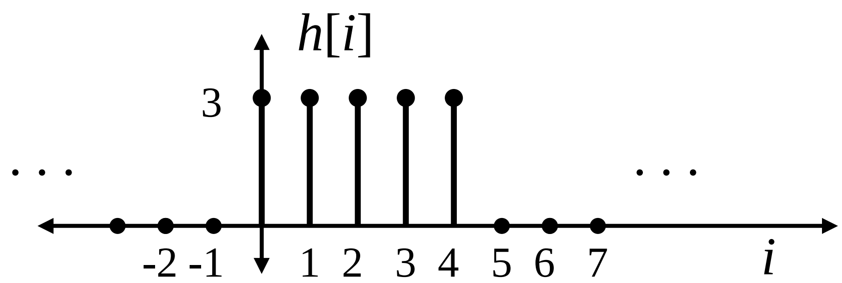


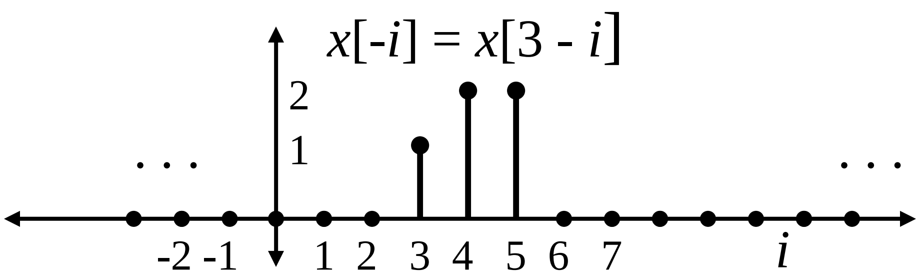
Sum over i ⇒ $y[4] = 3 + 6 + 6 = 15$

Steps 3&4 for *n* = 5

Step 3: For ***n* = 5**, shift by n to get $x[n - i]$

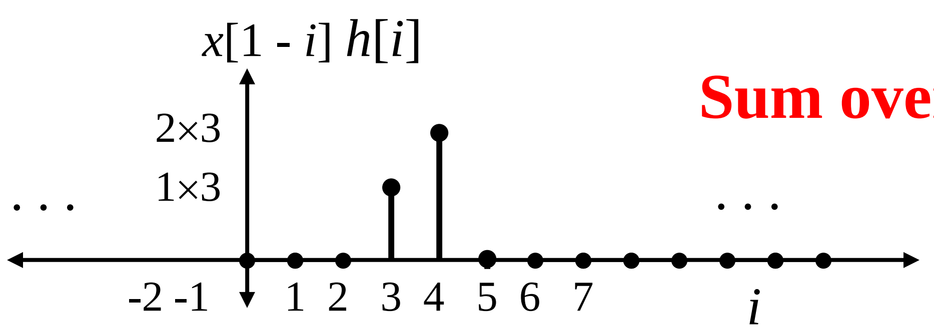
Positive n gives a Right-shift





shifted to the right by five

Step 4: For ***n* = 5**, Form the product $x[i]h[n – i]$ and sum its elements to give $y[n]$

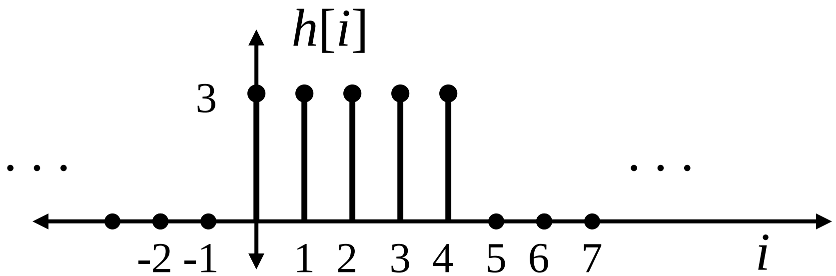


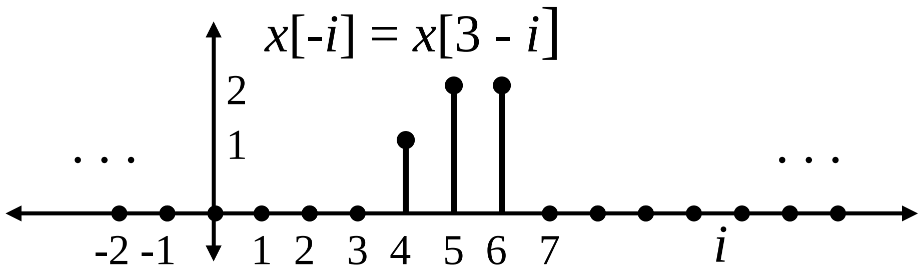
Sum over *i* $\Rightarrow$ $y[5] = 3 + 6 = 9$

<u>Steps 3&4 for n = 6</u>

<u>Step 3</u>: For **<u>n = 6</u>**, shift by n to get $x[n - i]$

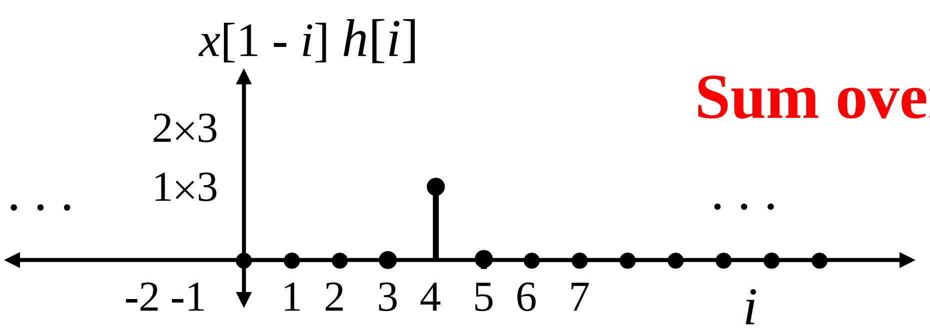
Positive n gives a Right-shift





shifted to the right by six

<u>Step 4</u>: For **<u>n = 6</u>**, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$

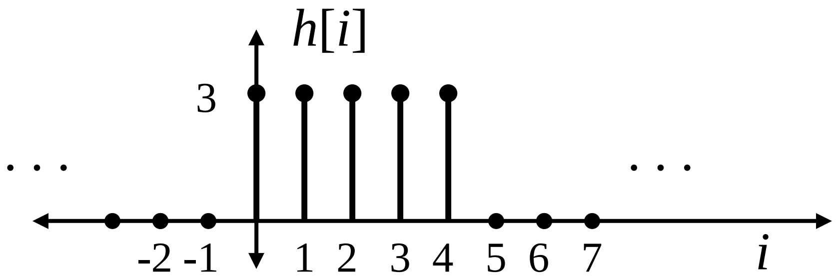


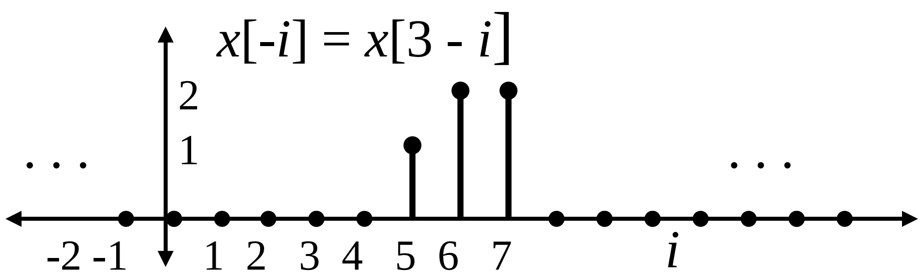
Sum over $i \Rightarrow$ $y[6] = 3$

Steps 3&4 for all $n > 6$

Step 3: For $n > 6$, shift by n to get $x[n - i]$

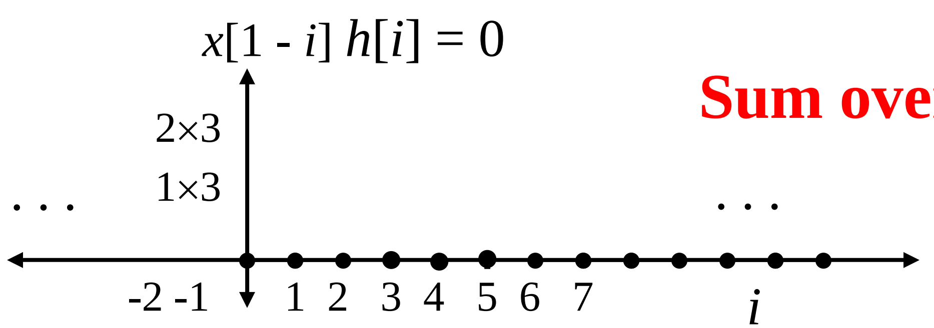
Positive n gives a Right-shift





shifted to the right by seven

Step 4: For $n > 6$, Form the product $x[i]h[n - i]$ and sum its elements to give $y[n]$

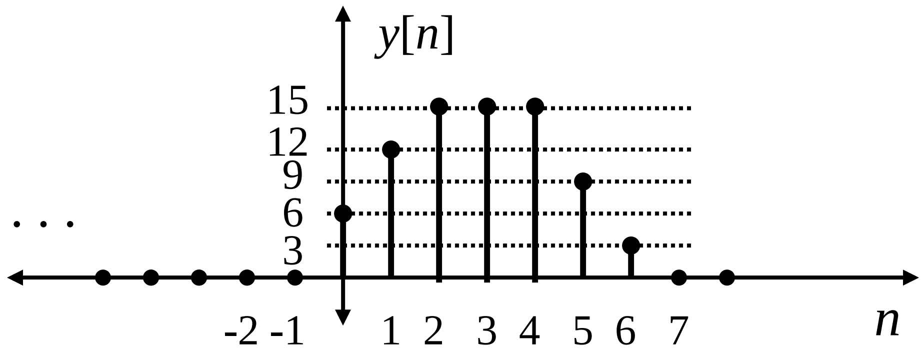


Sum over i $\Rightarrow$ $y[n] = 0 \quad \forall n > 6$

So… now we know the values of $y[n]$ for all values of n

We just need to put it all together as a function…

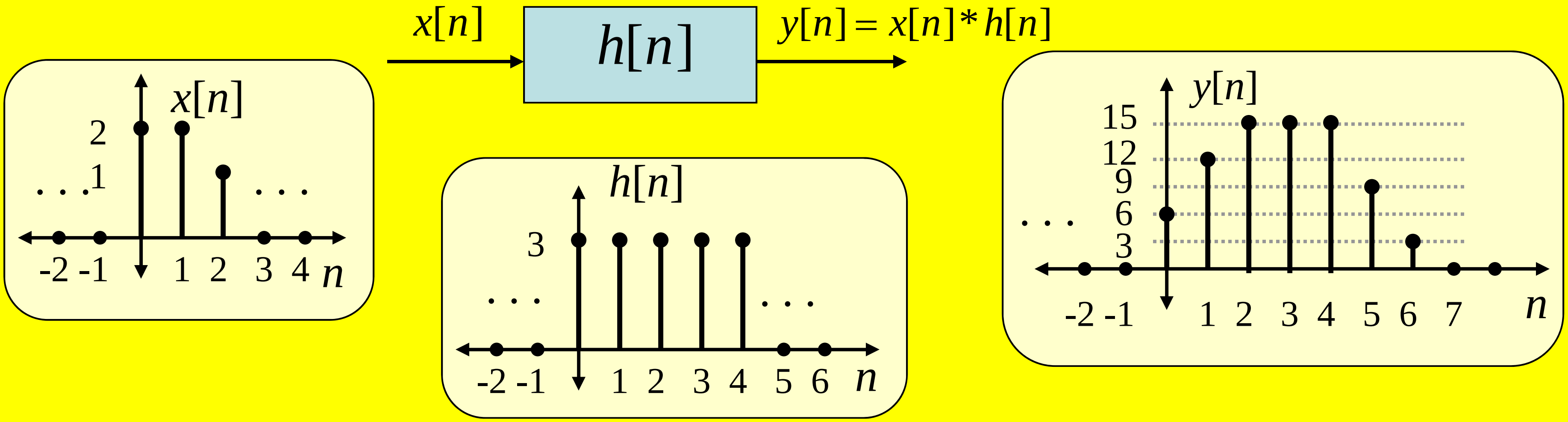
Here it is easiest to just plot it… you could also list it as a table.



Note that convolving these kinds of signals gives a "ramp-up" at the beginning and a "ramp-down" at the end.

Various kinds of "transients" at the beginning and end of a convolution are common.

<u>BIG PICTURE</u>: So… what we have just done is found the zero-state output of a system having an impulse response given by this $h[n]$ when the input is given by this $x[n]$:



<u>Link</u>: Web Demos of Graphical D-T Convolution

This is a good site that provides insight into how to visualize D-T convolution…

However, be sure you can do graphical convolution by hand without the aid of this site!!

Connection to FIR Filters

Consider a D-T system with impulse response *h*[*n*] that has <u>finite</u> duration… and set an FIR filter's coefficients equal to them: $b_0 = h[0] \quad b_1 = h[0] \ldots b_M = h[M]$