DSP First, 2/e

MODIFIED TLH

Lecture 12
Linearity & Time-Invariance
Convolution

READING ASSIGNMENTS

- This Lecture:
  - Chapter 5, Sections 5-4, 5-6, 5-7 & 5-8
    - Section 5-5 is covered, but not “in depth”
    - Convolution in Section 5-7 is important

LECTURE OBJECTIVES

- GENERAL PROPERTIES of FILTERS
  - LINEARITY
  - TIME-INVARIANCE
  - ==> **CONVOLUTION**

LTI SYSTEMS

- BLOCK DIAGRAM REPRESENTATION
  - Components for Hardware
  - Connect Simple Filters Together to Build More Complicated Systems

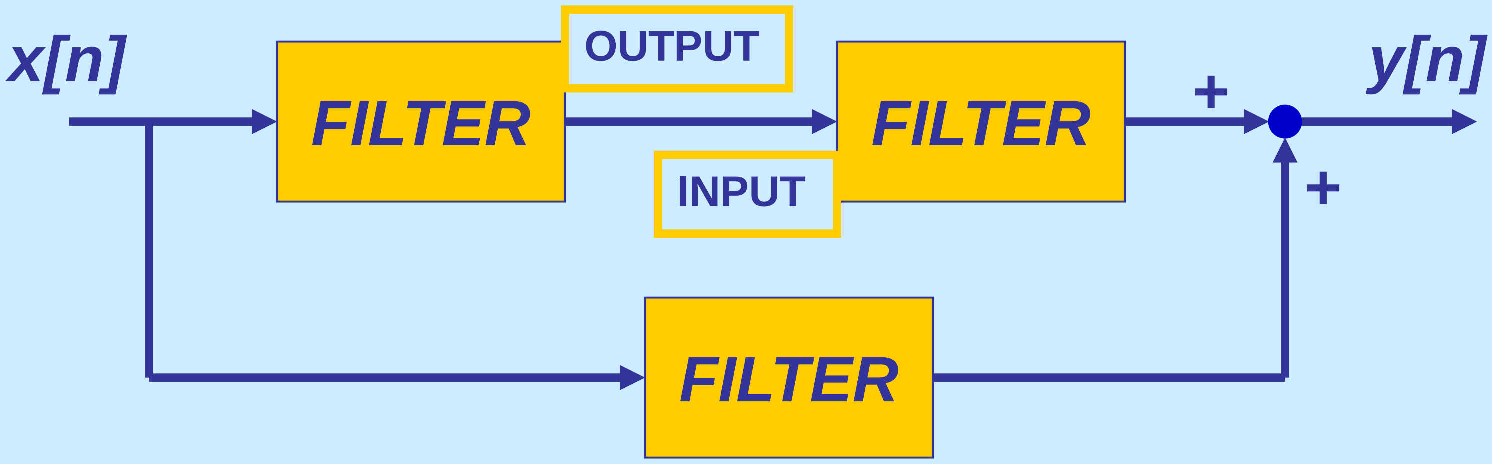
OVERVIEW

- IMPULSE RESPONSE, $h[n]$
  - FIR case: same as $\{b_k\}$
- CONVOLUTION
  - GENERAL: $y[n] = h[n] * x[n]$
  - GENERAL CLASS of SYSTEMS
  - LINEAR and TIME-INVARIANT
- ALL **LTI** systems have $h[n]$ & use convolution

DIGITAL FILTERING

- CONCENTRATE on the FILTER (DSP)
- DISCRETE-TIME SIGNALS
  - FUNCTIONS of n, the "time index"
  - INPUT $x[n]$
  - OUTPUT $y[n]$

BUILDING BLOCKS



- BUILD UP COMPLICATED FILTERS
  - FROM SIMPLE MODULES
  - Ex: FILTER MODULE MIGHT BE 3-pt FIR

GENERAL FIR FILTER

- FILTER COEFFICIENTS $\{b_k\}$
  - DEFINE THE FILTER

$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$

  - For example, $\boxed{b_k = \{3, -1, 2, 1\}}$

$$y[n] = \sum_{k=0}^{3} b_k x[n-k]$$
$$= 3x[n] - x[n-1] + 2x[n-2] + x[n-3]$$

MATLAB for FIR FILTER

- `yy = conv(bb,xx)`
  - VECTOR **bb** contains Filter Coefficients
  - DSP-First: `yy = firfilt(bb,xx)`
- FILTER COEFFICIENTS $\{b_k\}$

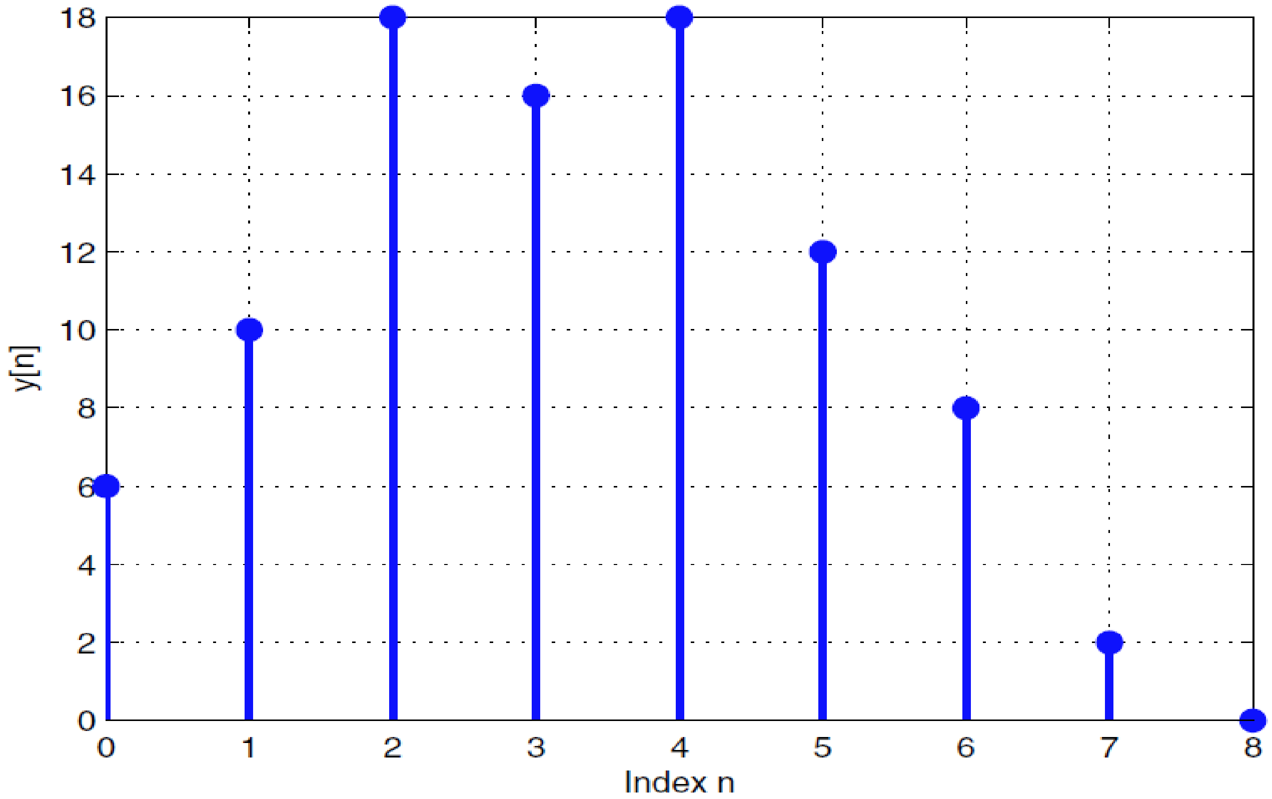
$$\boxed{y[n] = \sum_{k=0}^{M} b_k x[n-k]}$$

conv2()
for images

- We can check the answers using MATLAB's filter function

```
>> n = 0:8;
>> x = [2 4 6 4 2 0 0 0 0];
>> h = [3 -1 2 1];
>> y = filter(h,1,x);
>> y

y = 6     10     18     16     18     12     8     2     0
```



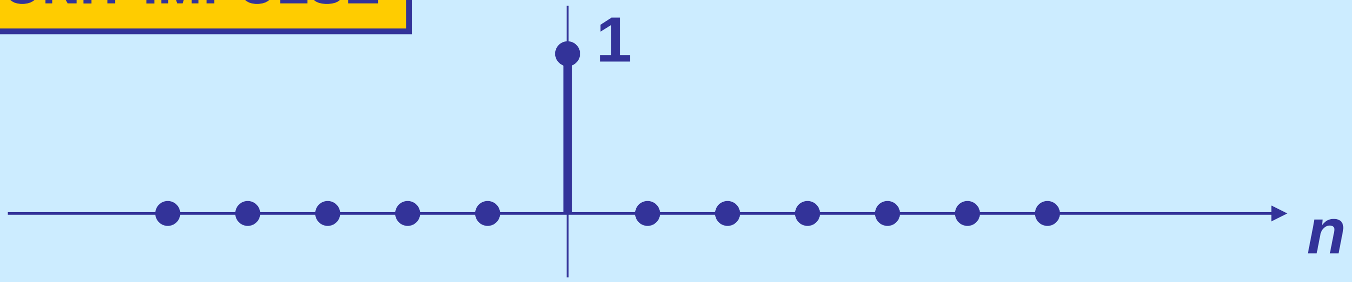
SPECIAL INPUT SIGNALS

- x[n] = SINUSOID
- x[n] has only one NON-ZERO VALUE

Later, sinusoid leads to the FREQUENCY RESPONSE

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

UNIT-IMPULSE



UNIT IMPULSE RESPONSE

- FIR filter DIFFERENCE EQUATION is specified by the filter coefficients b_k

$$\boxed{y[n] = \sum_{k=0}^{M} b_k x[n-k]}$$

- EQUIVALENCE: can we describe the filter using a SIGNAL instead?

FIR IMPULSE RESPONSE

$$y[n]=\sum_{k=0}^{M} b_k x[n-k]$$

$$h[n]=\sum_{k=0}^{M} b_k \delta[n-k]$$

n	$n<0$	0	1	2	3	$\ldots$	M	$M+1$	$n>M+1$
$x[n]=\delta[n]$	0	1	0	0	0	0	0	0	0
$y[n]=h[n]$	0	b_0	b_1	b_2	b_3	$\ldots$	b_M	0	0

- Impulse response $h[k]=b_k$ is, in fact, a SIGNAL description of filter coefficients
- Allows us to write CONVOLUTION sum

LTI: Convolution Sum

- **Output = Convolution of x[n] & h[n]**
  - NOTATION: $y[n] = h[n] * x[n]$
  - FIR case:

FINITE LIMITS

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

Same as $\mathbf{b_k}$

FINITE LIMITS

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

LTI: Convolution Sum

- Delay the signal x[n] & then multiply by filter coefficients that come from h[n]

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

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

CONVOLUTION Exa

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

$$h[n] = \delta[n] - \delta[n-1] + 2\delta[n-2] - \delta[n-3] + \delta[n-4]$$

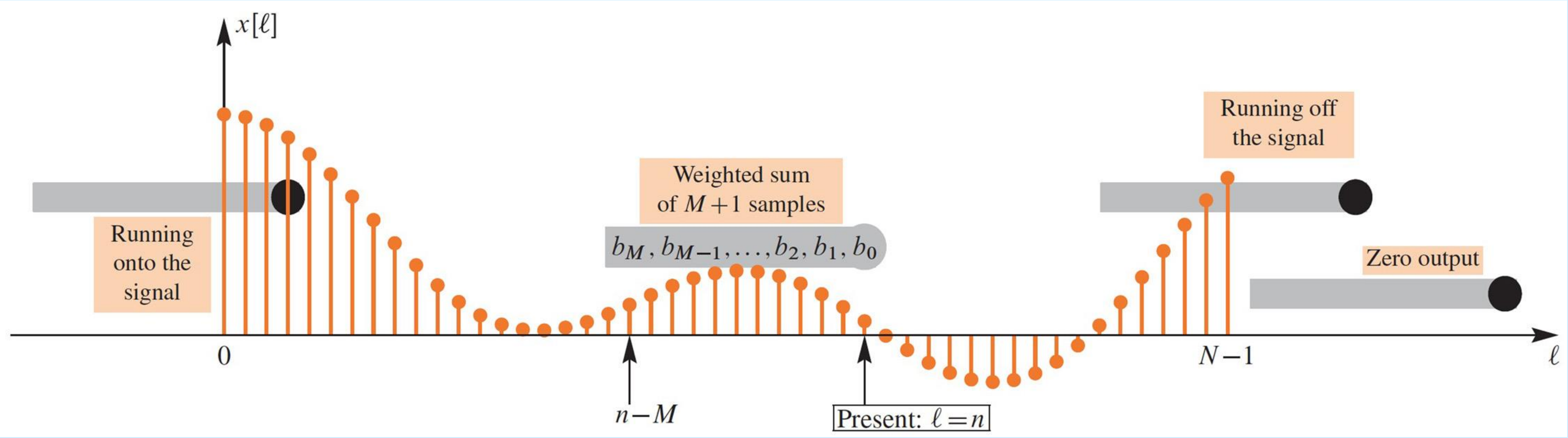
$$x[n] = u[n]$$

n	-1	0	1	2	3	4	5	6	7
$x[n]$	0	1	1	1	1	1	1	1	...
$h[n]$	0	1	-1	2	-1	1	0	0	0
$h[0]x[n]$	0	1	1	1	1	1	1	1	1
$h[1]x[n-1]$	0	0	-1	-1	-1	-1	-1	-1	-1
$h[2]x[n-2]$	0	0	0	2	2	2	2	2	2
$h[3]x[n-3]$	0	0	0	0	-1	-1	-1	-1	-1
$h[4]x[n-4]$	0	0	0	0	0	1	1	1	1
$y[n]$	0	1	0	2	1	2	2	2	...

GENERAL CAUSAL FIR FILTER

- SLIDE a Length-L WINDOW across x[n]

$$y[n] = \sum_{k=0}^{M} b_k x[n-k] = b_0 x[n] + b_1 x[n-1] + \cdots + b_M x[n-M]$$



$x[n-M]$ **$x[n]$**

POP QUIZ

- FIR Filter is “FIRST DIFFERENCE”
  - $y[n] = x[n] - x[n-1]$
- Write output as a convolution
  - Need impulse response

$$h[n] = \delta[n] - \delta[n-1]$$

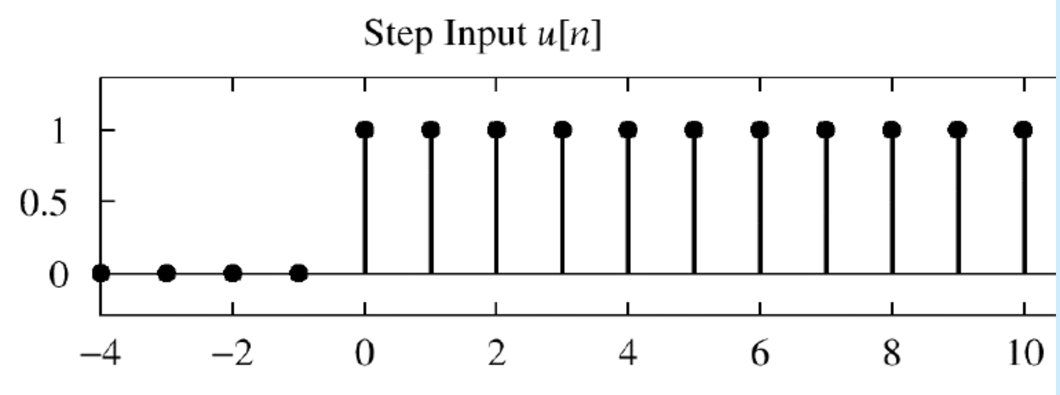
  - Then, an equivalent way to compute the output:

$$\begin{aligned} y[n] = h[n] * x[n] &= \big(\delta[n] - \delta[n-1]\big) * x[n] \\ &= \big(\delta[n] * x[n]\big) - \big(\delta[n-1] * x[n]\big) \\ &= x[n] - x[n-1] \end{aligned}$$

POP QUIZ

- FIR Filter is “FIRST DIFFERENCE”
  - $y[n] = x[n] - x[n-1]$
- INPUT is “UNIT STEP”

$$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$



- Find $y[n]$?

$$y[n] = u[n] - u[n-1] = \delta[n]$$

HARDWARE STRUCTURES

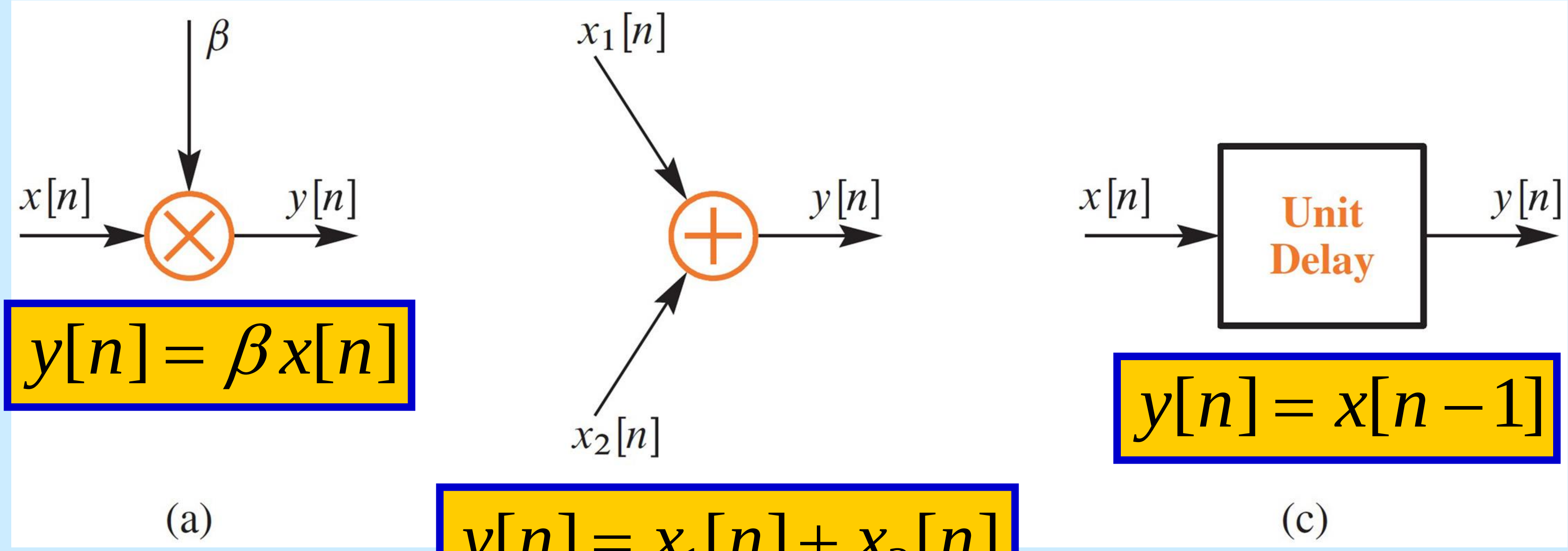
$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$

- INTERNAL STRUCTURE of "FILTER"
  - WHAT COMPONENTS ARE NEEDED?
  - HOW DO WE "HOOK" THEM TOGETHER?
- SIGNAL FLOW GRAPH NOTATION

HARDWARE ATOMS

- Add, Multiply & Store

$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$



$$y[n] = \beta x[n]$$

$$y[n] = x_1[n] + x_2[n]$$

$$y[n] = x[n-1]$$

FIR STRUCTURE

- Direct Form

$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$

SIGNAL FLOW GRAPH

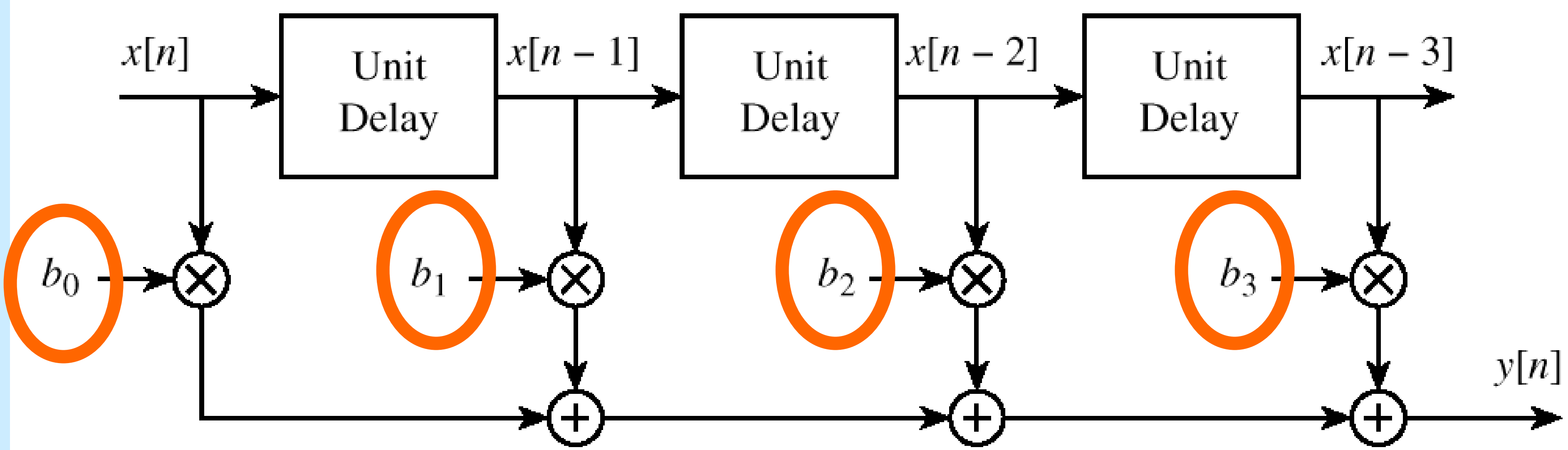


Figure 5.13 Block-diagram structure for the Mth order FIR filter.

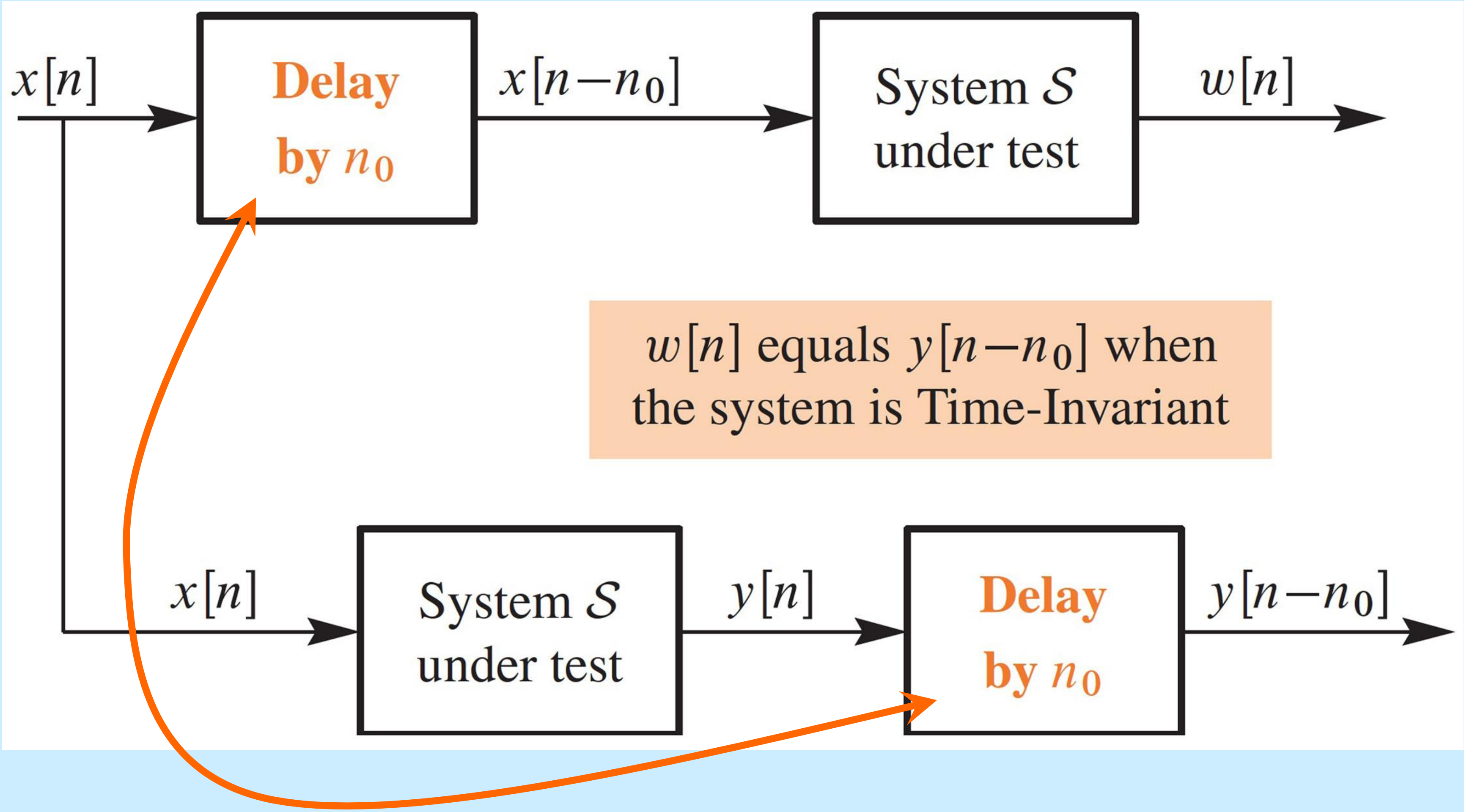
SYSTEM PROPERTIES

- MATHEMATICAL DESCRIPTION
- TIME-INVARIANCE
- LINEARITY
- CAUSALITY
  - “No output prior to input”

TIME-INVARIANCE

- IDEA:
  - "Time-Shifting the input will cause the same time-shift in the output"
- EQUIVALENTLY,
  - We can prove that
    - The time origin (n=0) is picked arbitrary

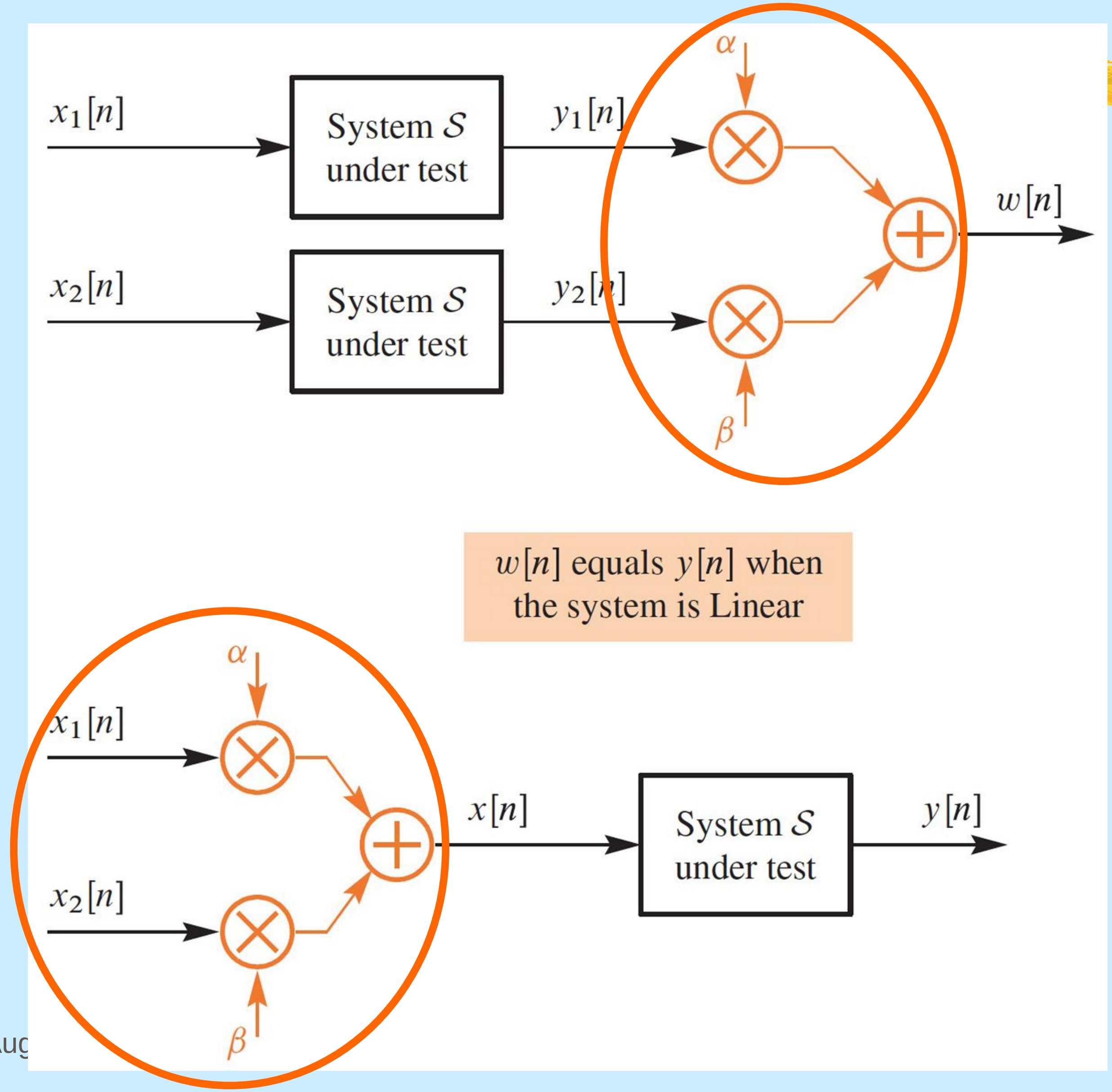
TESTING Time-Invariance



LINEAR SYSTEM

- LINEARITY = Two Properties
- SCALING
  - "Doubling x[n] will double y[n]"
- SUPERPOSITION:
  - "Adding two inputs gives an output that is the sum of the individual outputs"

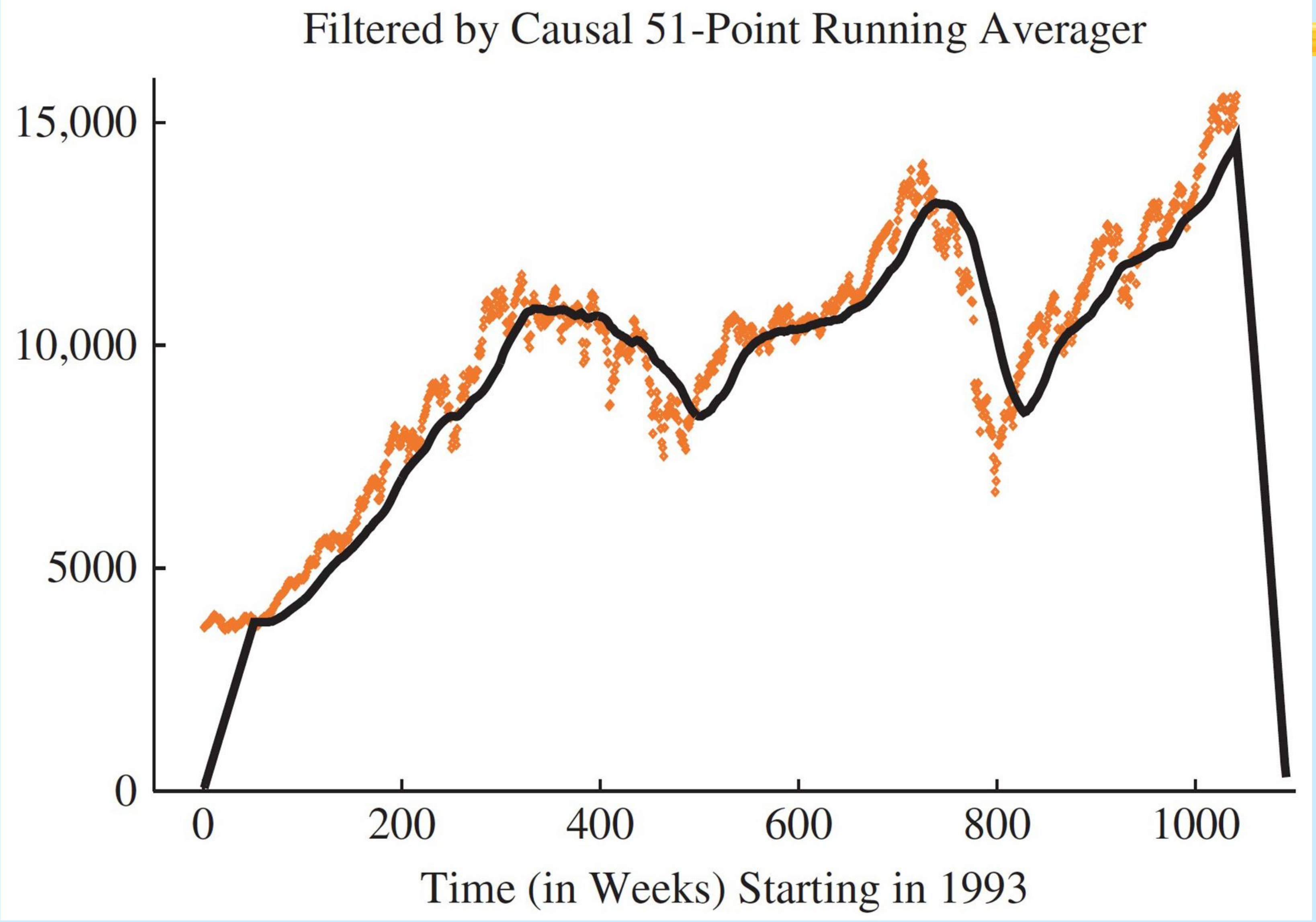
TESTING LINEARITY



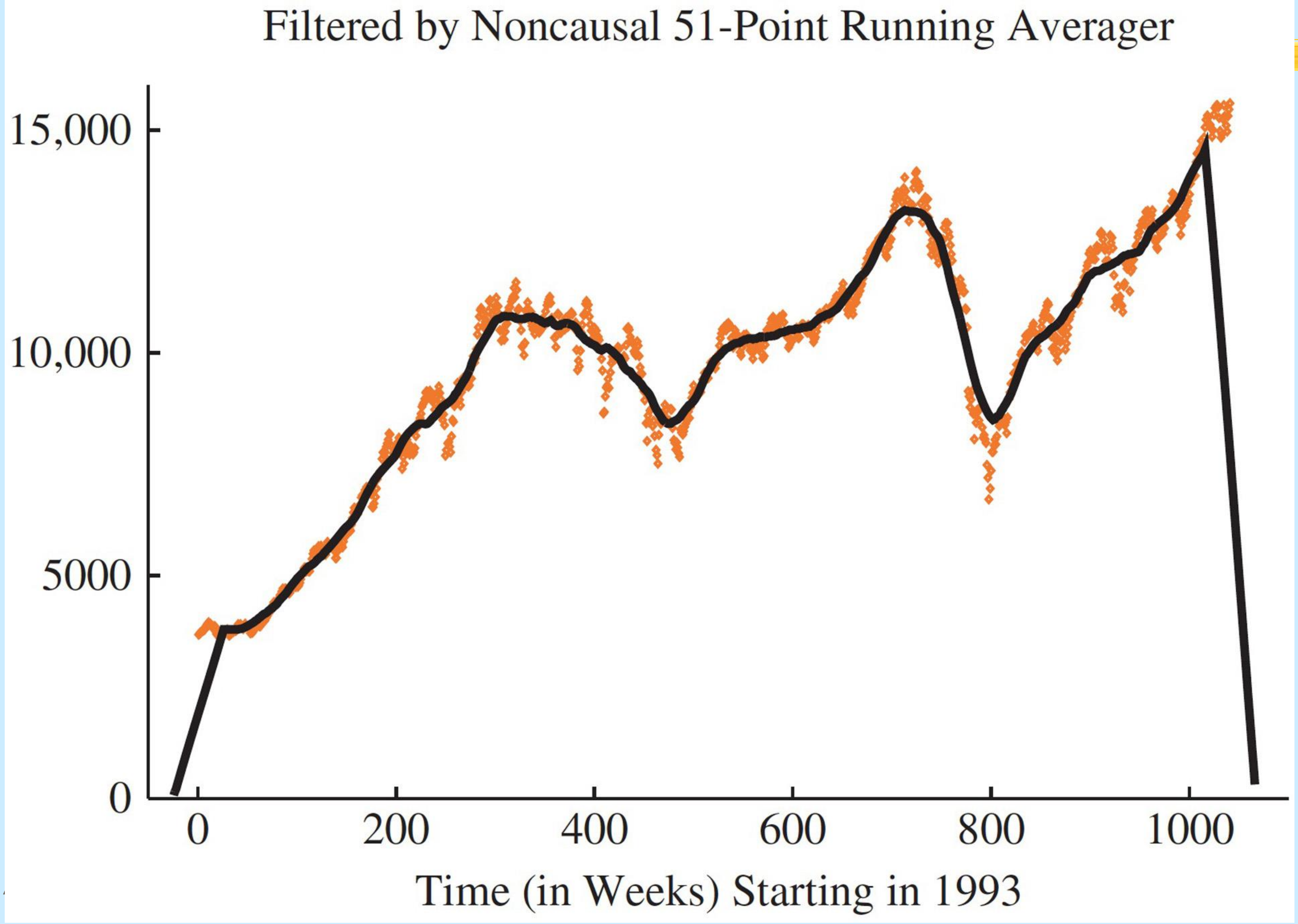
LTI SYSTEMS $y[n] = h[n] * x[n]$

- LTI: Linear & Time-Invariant
- COMPLETELY CHARACTERIZED by:
  - **IMPULSE RESPONSE** h[n]
  - **CONVOLUTION**:
    - The “rule” defining the system can ALWAYS be re-written as convolution
- FIR Example: h[n] is same as b_k

FILTER STOCK PRICES



STOCK PRICES FILTERED (2)



CASCADE SYSTEMS

- Does the order of S_1 & S_2 matter?
  - NO, LTI SYSTEMS can be rearranged !!!
  - WHAT ARE THE FILTER COEFFS? $\{b_k\}$

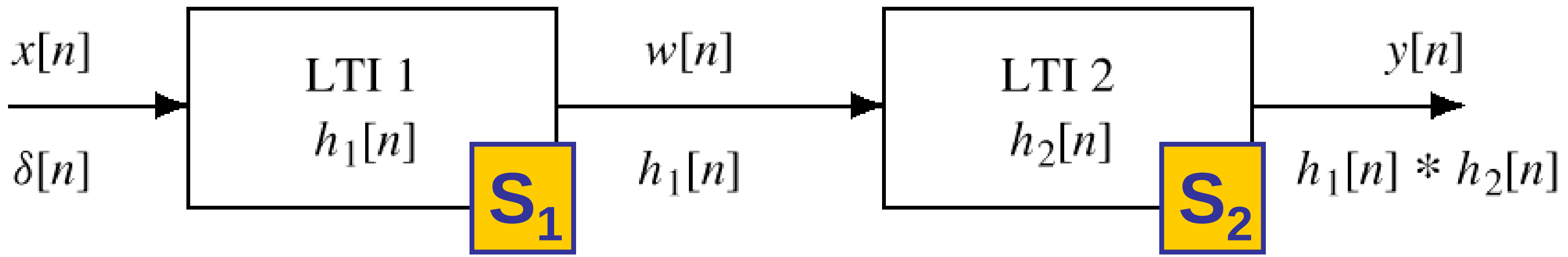


Figure 5.19 A Cascade of Two LTI Systems.