



ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE

# *Circuits and Systems I*

LECTURE #10

FIR Filtering: An Introduction

**lions@epfl**

*Prof. Dr. Volkan Cevher*

**LIONS/Laboratory for Information and Inference Systems**

# License Info for SPFirst Slides

- This work released under a [Creative Commons License](#) with the following terms:
- Attribution
  - The licensor permits others to copy, distribute, display, and perform the work. In return, licensees must give the original authors credit.
- Non-Commercial
  - The licensor permits others to copy, distribute, display, and perform the work. In return, licensees may not use the work for commercial purposes—unless they get the licensor's permission.
- Share Alike
  - The licensor permits others to distribute derivative works only under a license identical to the one that governs the licensor's work.
- [Full Text of the License](#)
- *This (hidden) page should be kept with the presentation*

# Outline - Today

- Today <> Section 5-1  
Section 5-2  
Section 5-3
- Next week <> Section 5-4  
Section 5-5  
Section 5-6  
Section 5-7

**CSI**  
Progress  
Level:



# Lecture Objectives

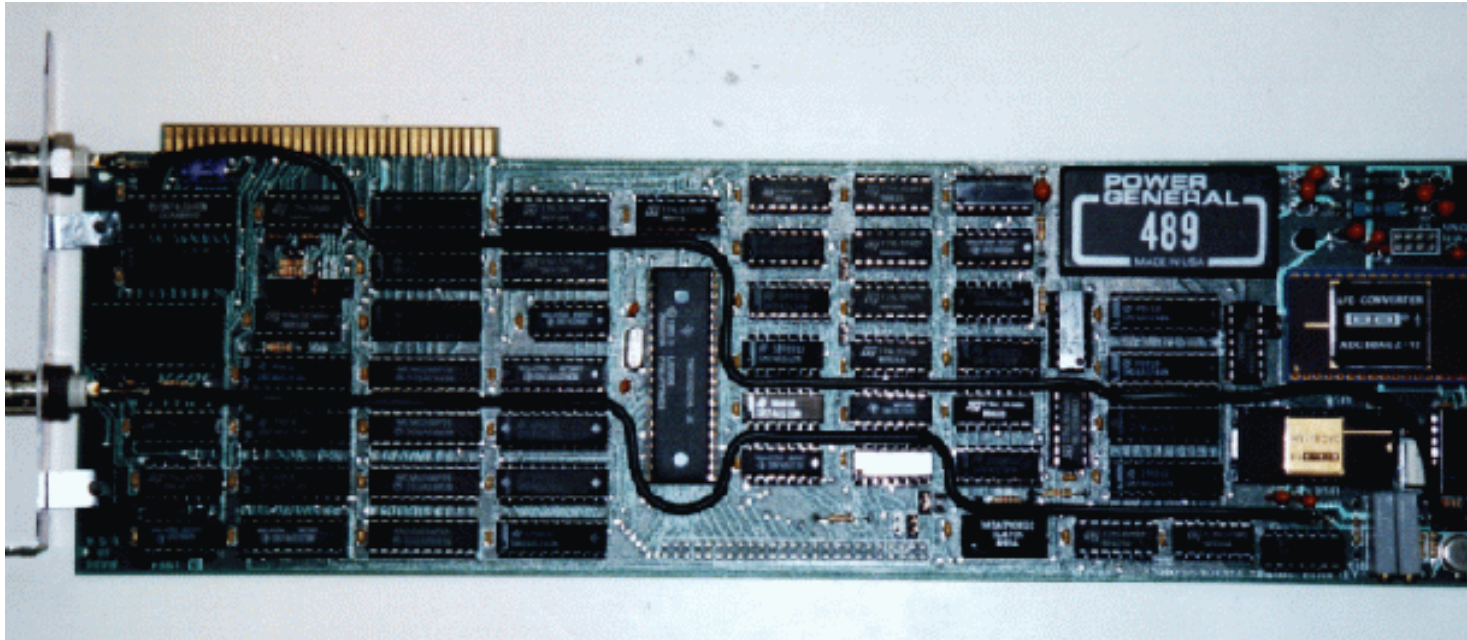
- INTRODUCE FILTERING IDEA
  - Weighted Average
  - Running Average
- FINITE IMPULSE RESPONSE FILTERS
  - **FIR** Filters
    - Show how to **compute** the output  $y[n]$  from the input signal,  $x[n]$

# Digital Filtering



- CONCENTRATE on the COMPUTER
  - PROCESSING ALGORITHMS
  - SOFTWARE (MATLAB)
  - HARDWARE: DSP chips, VLSI
- **DSP:** DIGITAL SIGNAL PROCESSING

# The TMS32010, 1983



First PC plug-in board from Atlanta Signal Processors Inc.

# Rockland Digital Filter, 1971



The image shows a photograph of the Rockland Model 4136 Programmable Digital Filter. The device is a rectangular metal box with a front panel featuring three large rotary knobs labeled 'FILTER ORDER', 'NO. ZEROES', and 'FILTER GAIN'. Below these knobs are three pairs of indicator lights labeled 'ON', 'POWER', 'OVERFLOW', and 'OVERLOAD'. The top of the device has a black label with the text 'ROCKLAND PROGRAMMABLE DIGITAL FILTER - MODEL 4136'.

**Model 4136  
PROGRAMMABLE  
DIGITAL  
FILTER**

**Variable-Order Digital Filter for Realizing All Classical Designs**

The Rockland Model 4136 Programmable Digital Filter consists of a second-order digital filter section which is multiplexed four ways to achieve eighth-order filtering. Each of the four sections has fully-programmable coefficients which are stored internally in a read-write memory.

Filter input and output words are in 16-bit parallel form at a maximum sampling rate of 80 KHz while internal computations are made with 24-bit accuracy.

**TRANSFER FUNCTION**

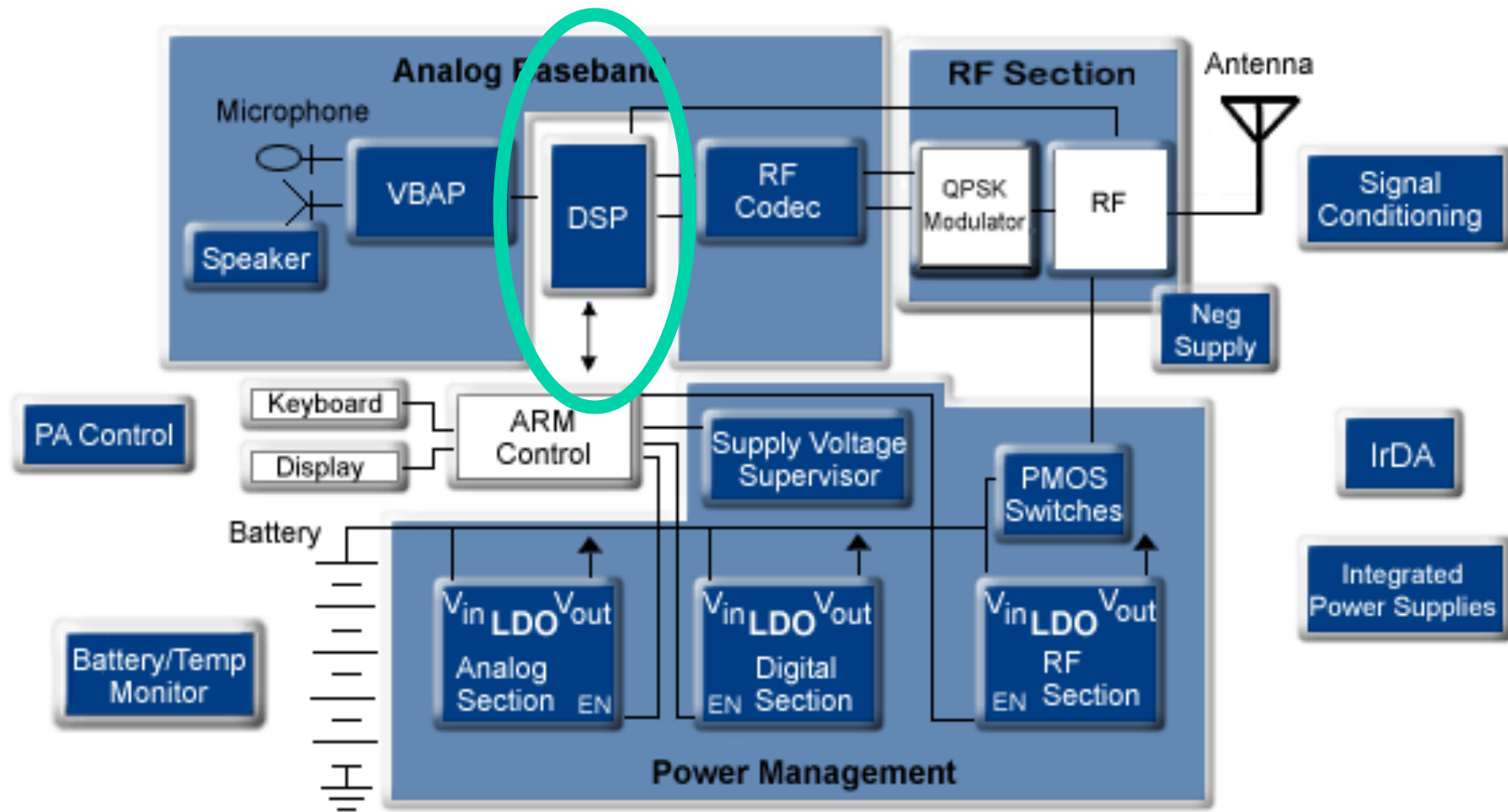
The transfer function from filter input to filter output in z-transform notation is given by

$$H_N(z) = \prod_{n=1}^N \frac{K_n(1+z^{-1}A1+z^{-2}A2)}{1-z^{-1}B1-z^{-2}B2} \quad (1)$$

where  $N=0,1,2,3,4$  is one-half the filter order se-

For the price of a small house, you could have one of these.

# Digital Cell Phone (circa. 2000)



Now it plays video



# DISCRETE-TIME SYSTEM



- OPERATE on  $x[n]$  to get  $y[n]$
- WANT a **GENERAL** CLASS of SYSTEMS
  - **ANALYZE** the SYSTEM
    - TOOLS: TIME-DOMAIN & FREQUENCY-DOMAIN
  - **SYNTHESIZE** the SYSTEM

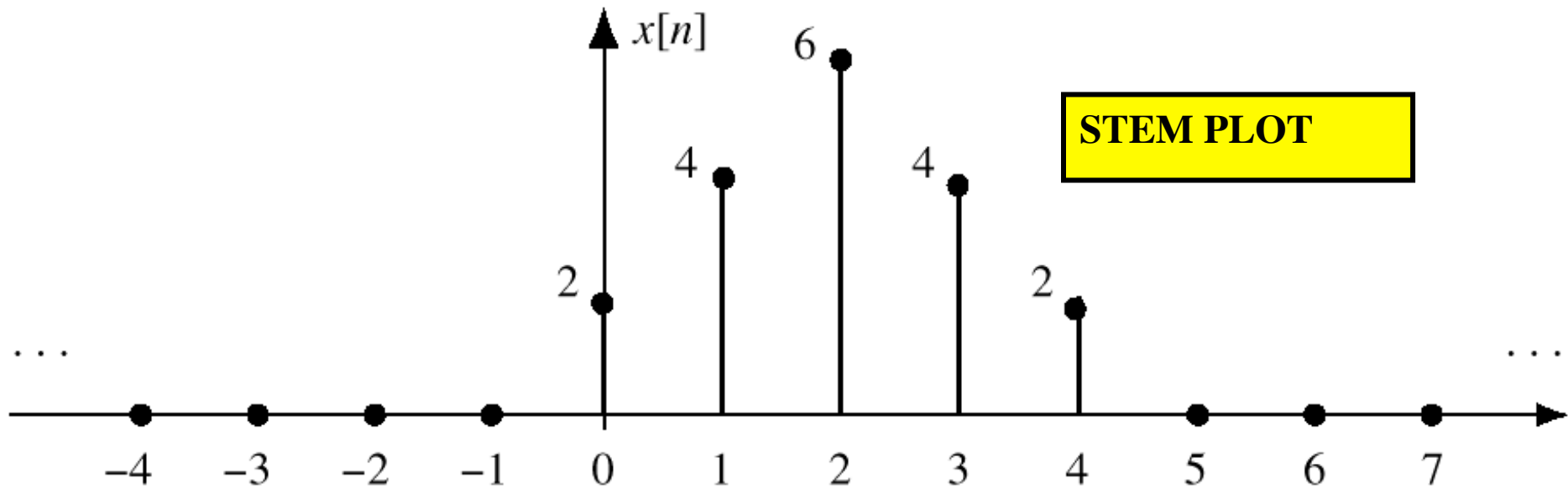
# D-T System Examples



- **EXAMPLES:**
  - POINTWISE OPERATORS
    - SQUARING:  $y[n] = (x[n])^2$
  - RUNNING AVERAGE
    - **RULE:** “the output at time  $n$  is the average of three consecutive input values”

# Discrete-Time Signal

- $x[n]$  is a LIST of NUMBERS
  - INDEXED by " $n$ "



# 3-PT Average System

- ADD 3 CONSECUTIVE NUMBERS
  - Do this for each “ $n$ ”

the following input–output equation

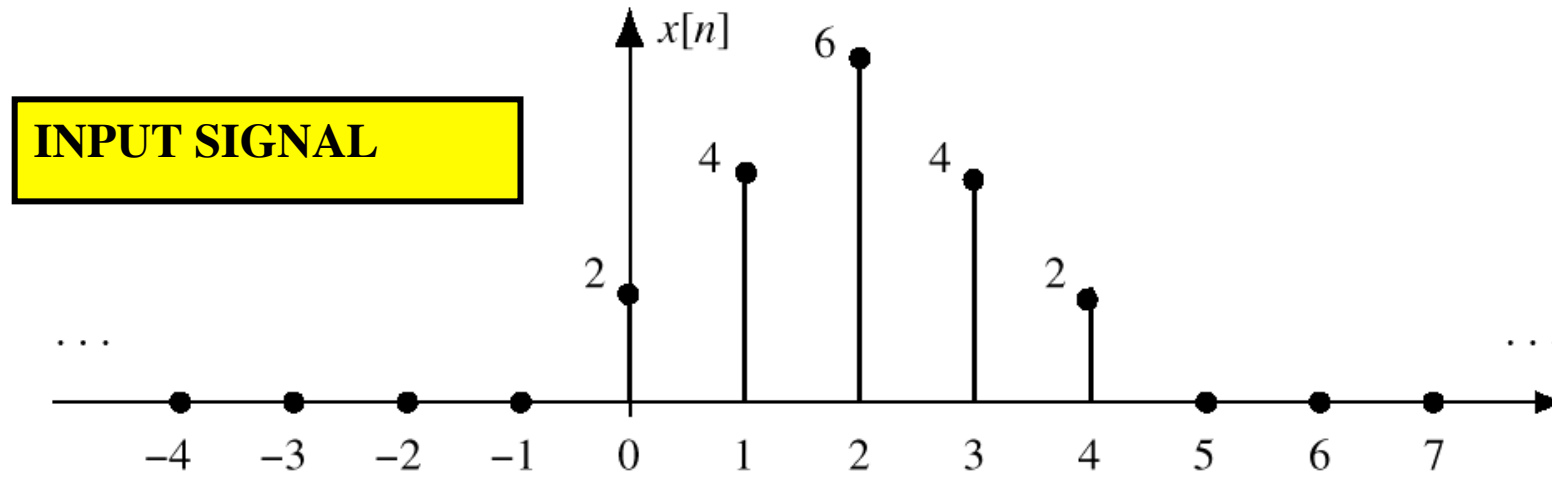
**Make a TABLE**

$$y[n] = \frac{1}{3}(x[n] + x[n + 1] + x[n + 2])$$

$n$	$n < -2$	$-2$	$-1$	$0$	$1$	$2$	$3$	$4$	$5$	$n > 5$
$x[n]$	0	0	0	2	4	6	4	2	0	0
$y[n]$	0	$\frac{2}{3}$	2	4	$\frac{14}{3}$	4	2	$\frac{2}{3}$	0	0

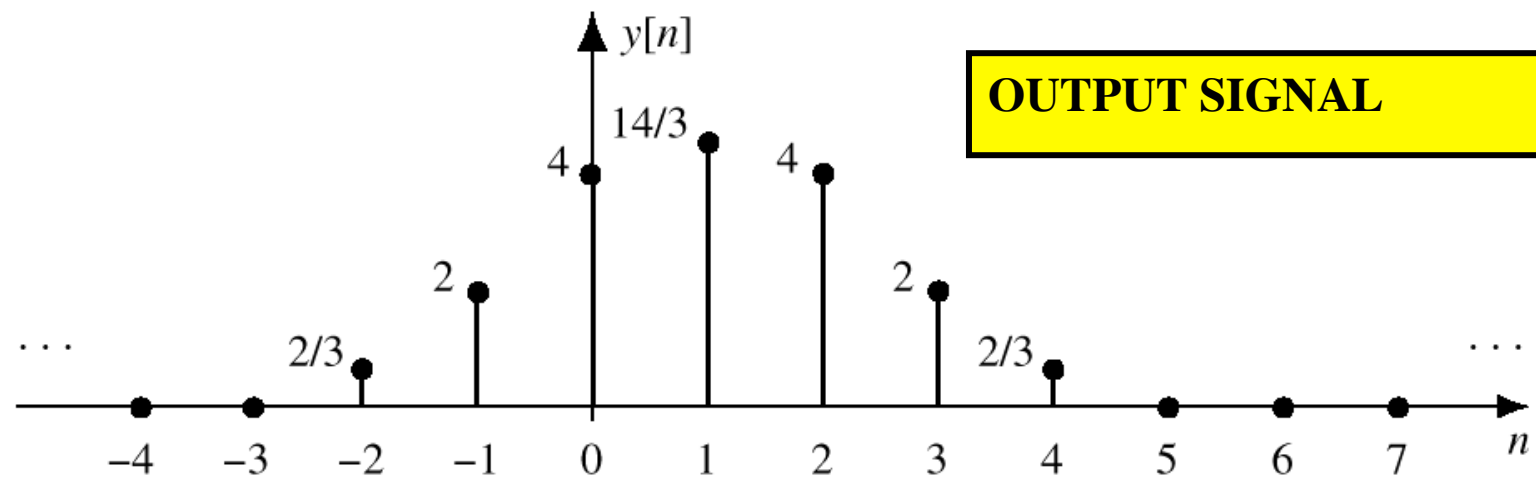
**$n=0$**   $y[0] = \frac{1}{3}(x[0] + x[1] + x[2])$

**$n=1$**   $y[1] = \frac{1}{3}(x[1] + x[2] + x[3])$



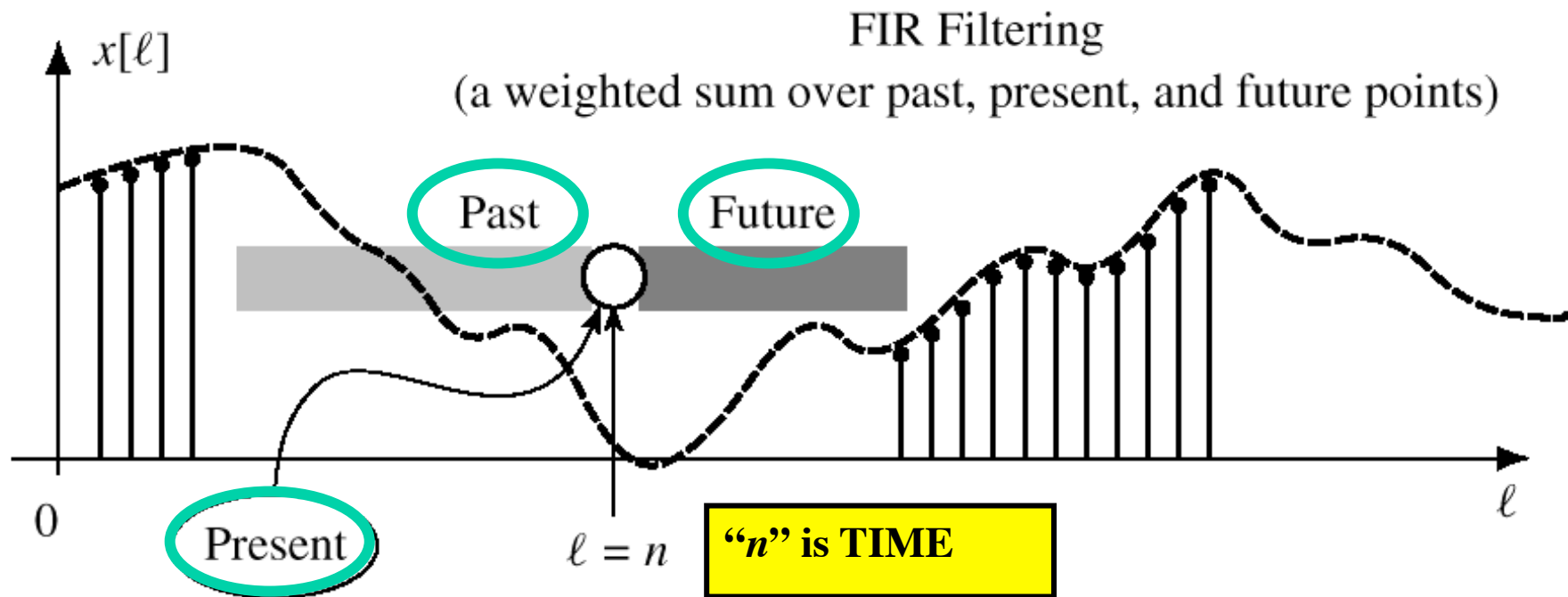
**Figure 5.2** Finite-length input signal,  $x[n]$ .

$$y[n] = \frac{1}{3} (x[n] + x[n + 1] + x[n + 2])$$



**Figure 5.3** Output of running average,  $y[n]$ .

# Past, Present, and Future



**Figure 5.4** The running-average filter calculation at time index  $n$  uses values within a sliding window (shaded). Dark shading indicates the future ( $l > n$ ); light shading, the past ( $l < n$ ).

# Another 3-pt Averager

- Uses "PAST" VALUES of  $x[n]$ 
  - IMPORTANT IF "n" represents REAL TIME
    - WHEN  $x[n]$  &  $y[n]$  ARE STREAMS

$$y[n] = \frac{1}{3}(x[n] + x[n-1] + x[n-2])$$

$n$	$n < -2$	$-2$	$-1$	$0$	$1$	$2$	$3$	$4$	$5$	$6$	$7$	$n > 7$
$x[n]$	0	0	0	2	4	6	4	2	0	0	0	0
$y[n]$	0	0	0	$\frac{2}{3}$	2	4	$\frac{14}{3}$	4	2	$\frac{2}{3}$	0	0

# General FIR Filter

- FILTER COEFFICIENTS  $\{b_k\}$

- DEFINE THE FILTER

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

- For example,

$$b_k = \{3, -1, 2, 1\}$$

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

$$= 3x[n] - x[n-1] + 2x[n-2] + x[n-3]$$



# General FIR Filter

- FILTER COEFFICIENTS  $\{b_k\}$

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

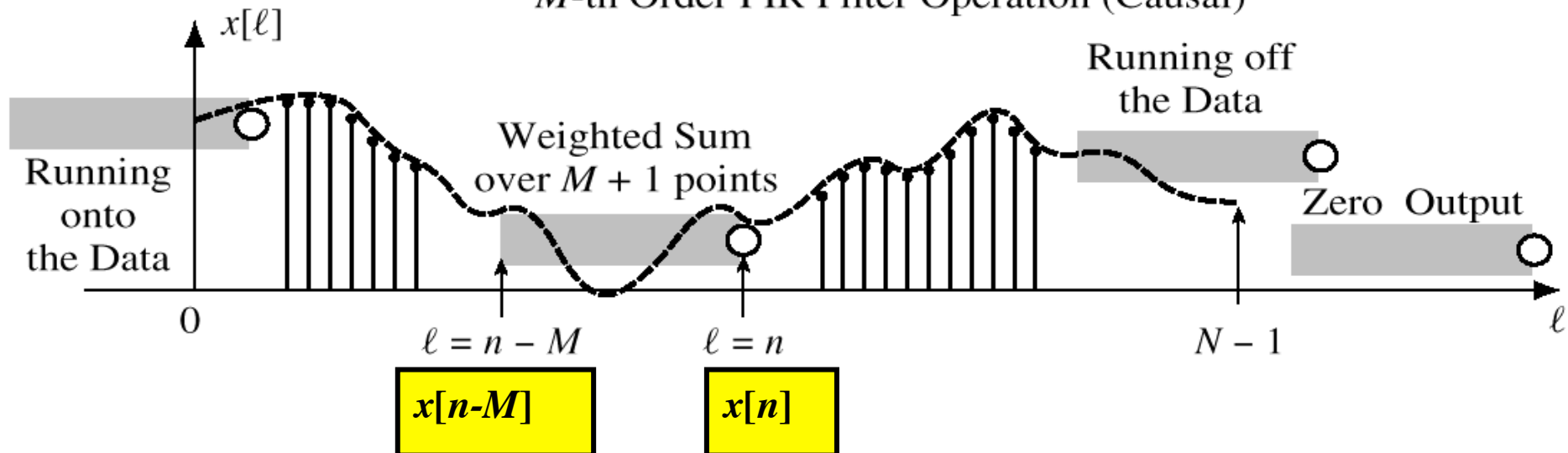
- FILTER **ORDER** is  $M$
- FILTER **LENGTH** is  $L = M + 1$ 
  - NUMBER of FILTER COEFFS is  $L$

# General FIR Filter

- SLIDE a WINDOW across  $x[n]$

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

*M*-th Order FIR Filter Operation (Causal)



# Filtered Stock Signal



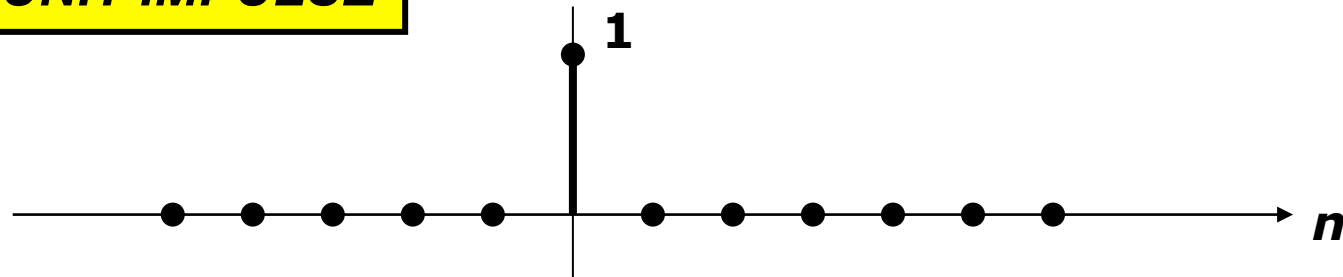
# Special Input Signals

- $x[n] = \text{SINUSOID}$
- $x[n]$  has only one NON-ZERO VALUE

**FREQUENCY RESPONSE (LATER)**

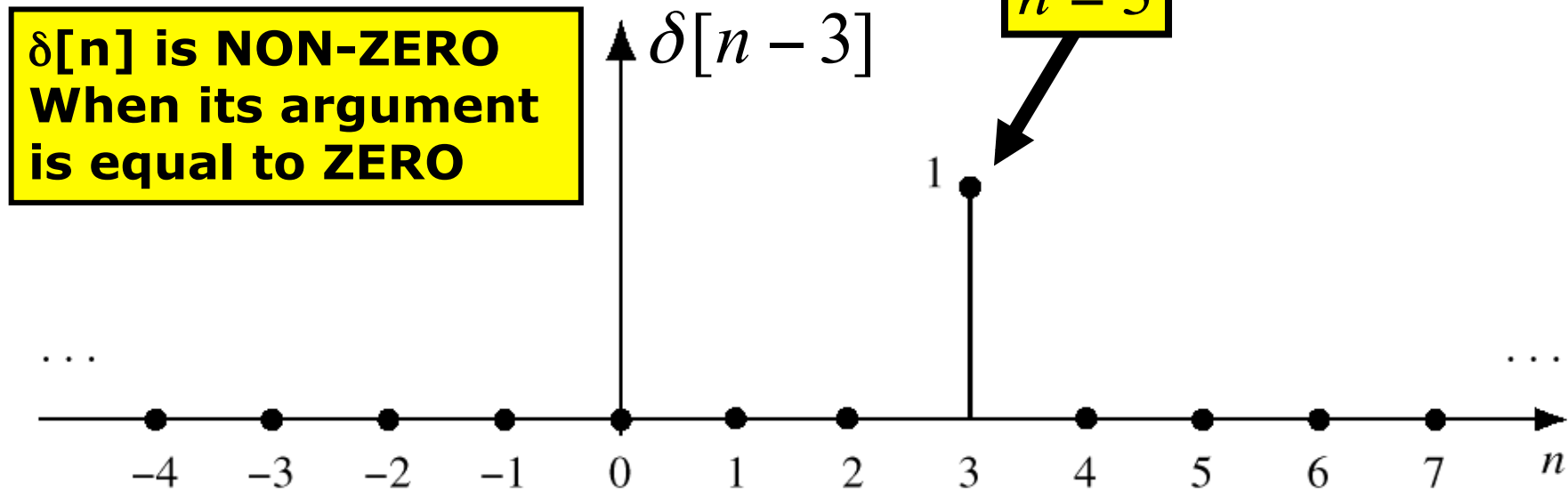
$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

**UNIT-IMPULSE**



# UNIT IMPULSE SIGNAL $\delta[n]$

$n$	...	-2	-1	0	1	2	3	4	5	6	...
$\delta[n]$	0	0	0	1	0	0	0	0	0	0	0
$\delta[n - 3]$	0	0	0	0	0	0	1	0	0	0	0

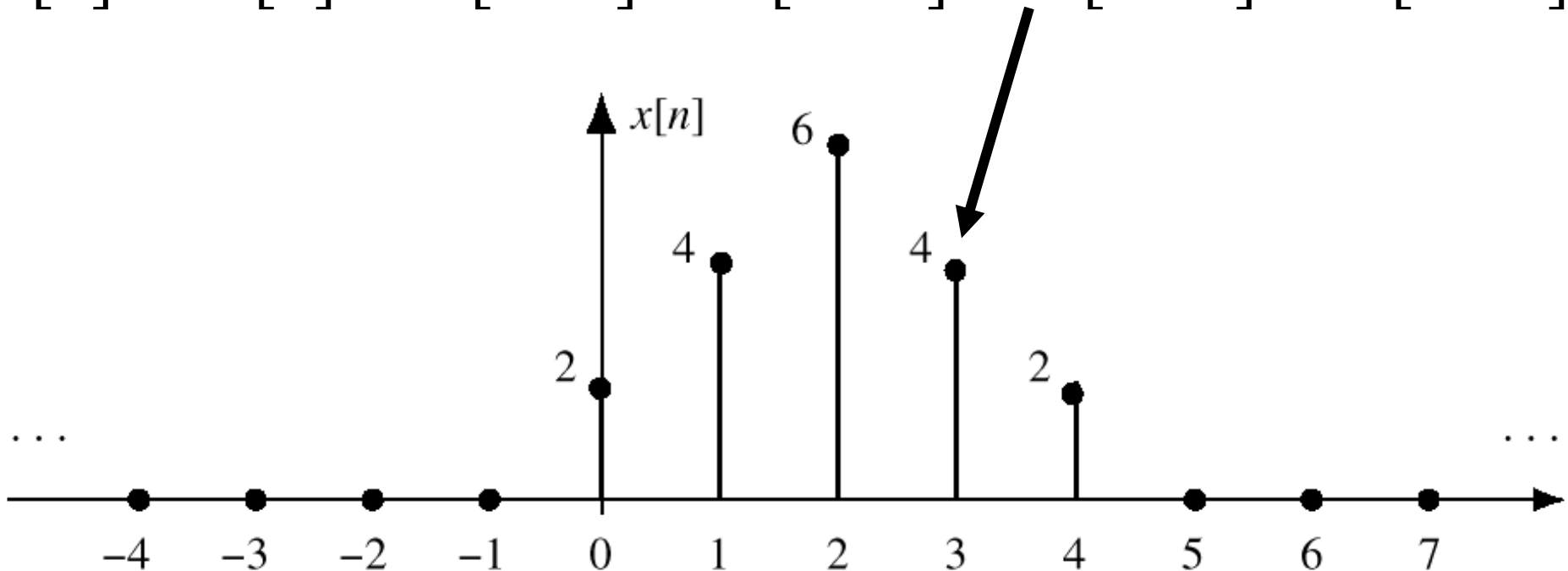


**Figure 5.7** Shifted impulse sequence,  $\delta[n - 3]$ .

# MATH FORMULA for $x[n]$

- Use **SHIFTED** IMPULSES to write  $x[n]$

$$x[n] = 2\delta[n] + 4\delta[n - 1] + 6\delta[n - 2] + 4\delta[n - 3] + 2\delta[n - 4]$$



# Sum of Shifted Impulses

$n$	...	-2	-1	0	1	2	3	4	5	6	...
$2\delta[n]$	0	0	0	2	0	0	0	0	0	0	0
$4\delta[n - 1]$	0	0	0	0	4	0	0	0	0	0	0
$6\delta[n - 2]$	0	0	0	0	0	6	0	0	0	0	0
$4\delta[n - 3]$	0	0	0	0	0	0	4	0	0	0	0
$2\delta[n - 4]$	0	0	0	0	0	0	0	2	0	0	0
$x[n]$	0	0	0	2	4	6	4	2	0	0	0

$$x[n] = \sum_k x[k]\delta[n - k] \quad \leftarrow \text{This formula ALWAYS works}$$

$$= \dots + x[-1]\delta[n + 1] + x[0]\delta[n] + x[1]\delta[n - 1] + \dots \quad (5.3.6)$$

# 4-pt Averager

- CAUSAL SYSTEM: USE PAST VALUES

$$y[n] = \frac{1}{4} (x[n] + x[n-1] + x[n-2] + x[n-3])$$

- INPUT = UNIT IMPULSE SIGNAL =  $\delta[n]$

$$x[n] = \delta[n]$$

$$y[n] = \frac{1}{4} \delta[n] + \frac{1}{4} \delta[n-1] + \frac{1}{4} \delta[n-2] + \frac{1}{4} \delta[n-3]$$

- OUTPUT is called “**IMPULSE RESPONSE**”

$$h[n] = \{ \dots, 0, 0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, \dots \}$$



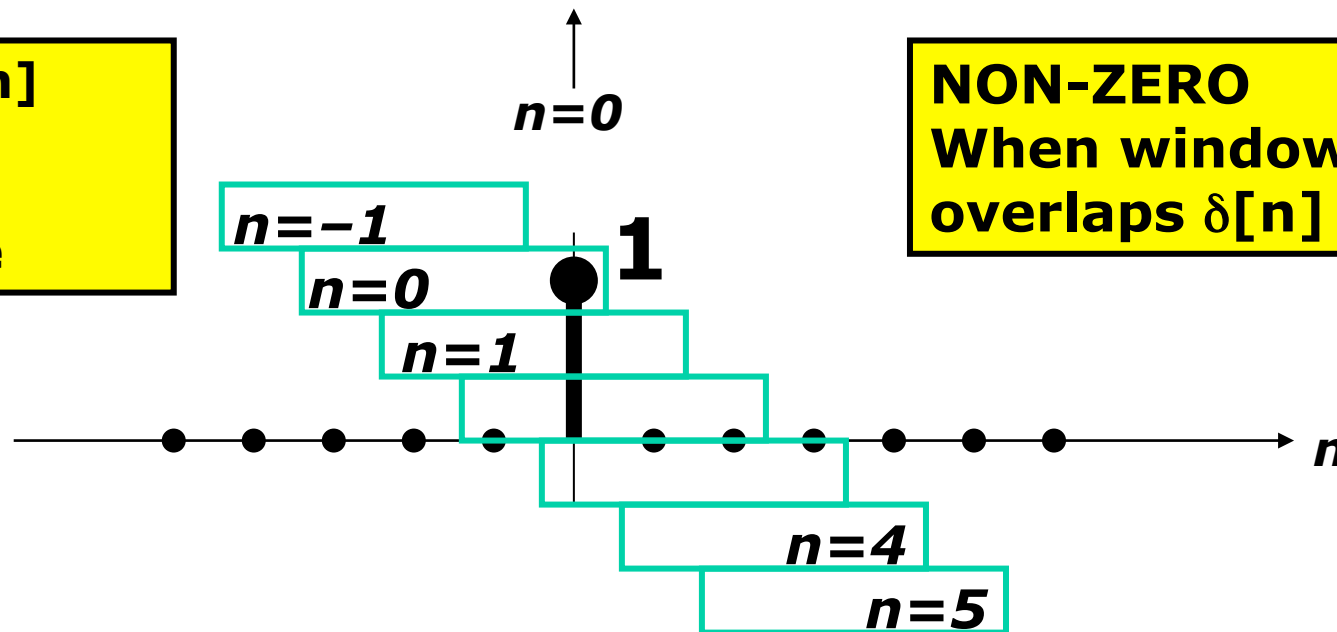
# 4-pt Avg Impulse Response

$$y[n] = \frac{1}{4} (x[n] + x[n-1] + x[n-2] + x[n-3])$$

$\delta[n]$  "READS OUT" the FILTER COEFFICIENTS

$$h[n] = \{\dots, 0, 0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, \dots\}$$

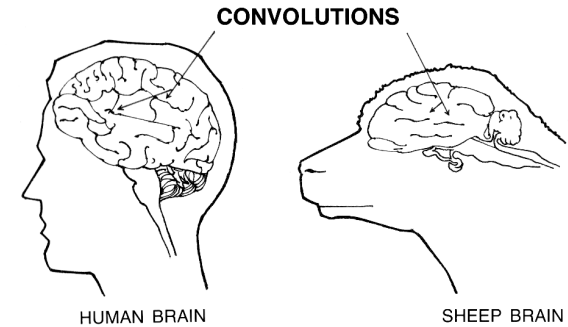
"h" in  $h[n]$   
denotes  
Impulse  
Response



**NON-ZERO**  
When window  
overlaps  $\delta[n]$

# FIR Impulse Response

- Convolution = Filter Definition
  - Filter Coeffs = Impulse Response



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

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

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

**CONVOLUTION**

# Filtering Example

- 7-point AVERAGER

- Removes cosine

- By making its amplitude (A) smaller

$$y_7[n] = \sum_{k=0}^6 \left(\frac{1}{7}\right)x[n-k]$$

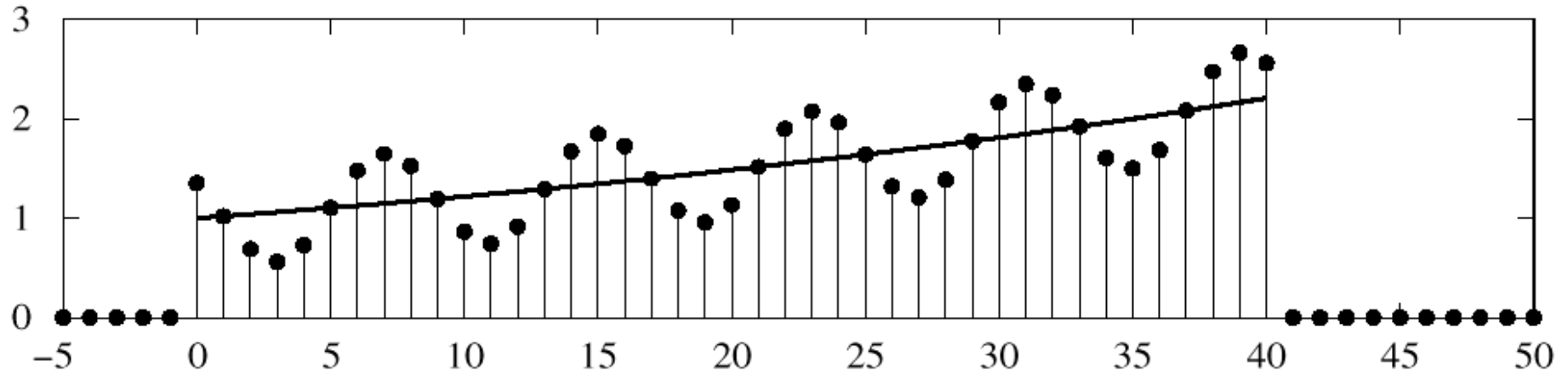
- 3-point AVERAGER

- Changes A slightly

$$y_3[n] = \sum_{k=0}^2 \left(\frac{1}{3}\right)x[n-k]$$

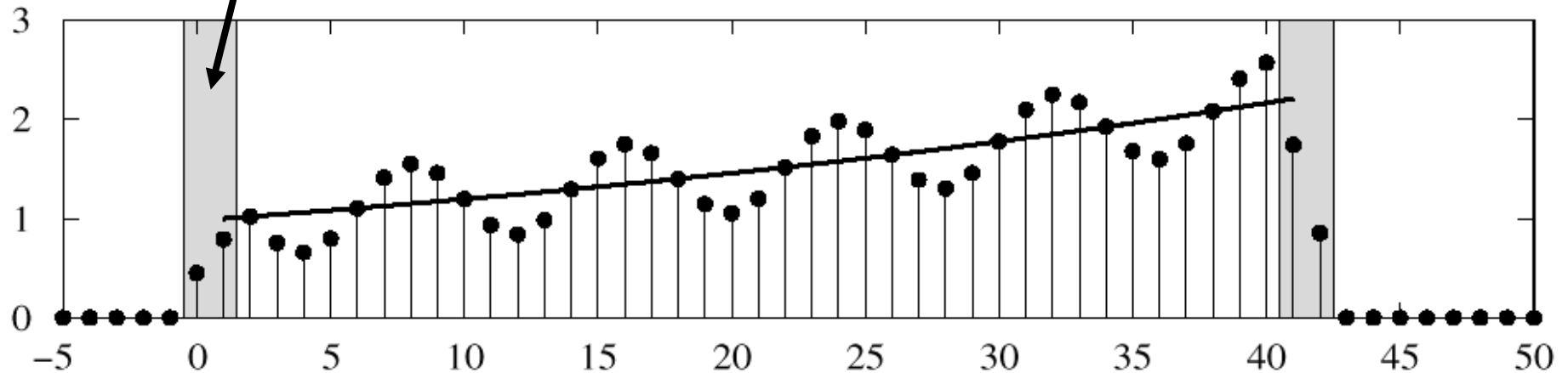
# 3-pt AVG EXAMPLE

Input :  $x[n] = (1.02)^n + \cos(2\pi n / 8 + \pi / 4)$  for  $0 \leq n \leq 40$



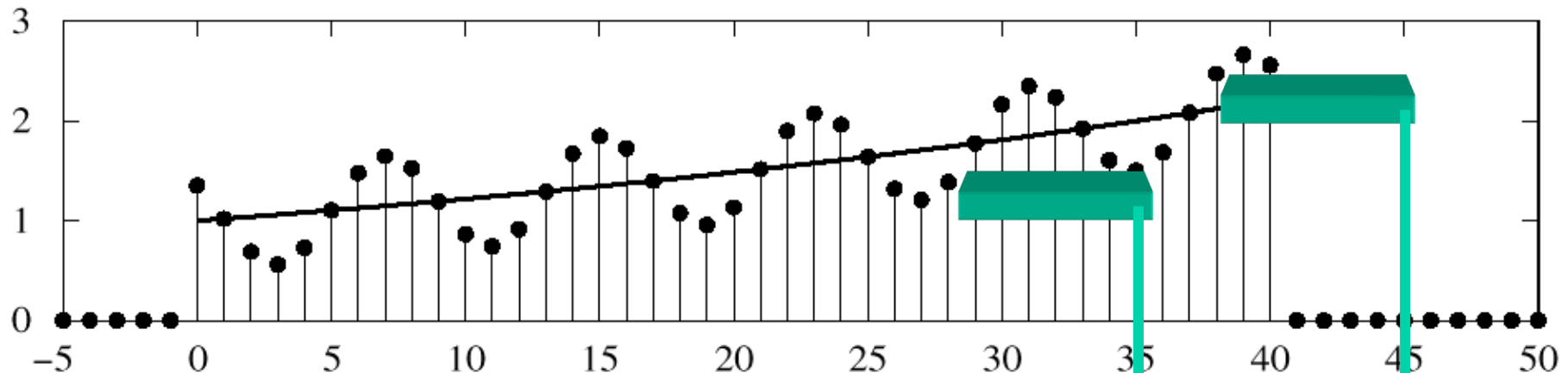
**USE PAST VALUES**

Output of 3-Point Running-Average Filter



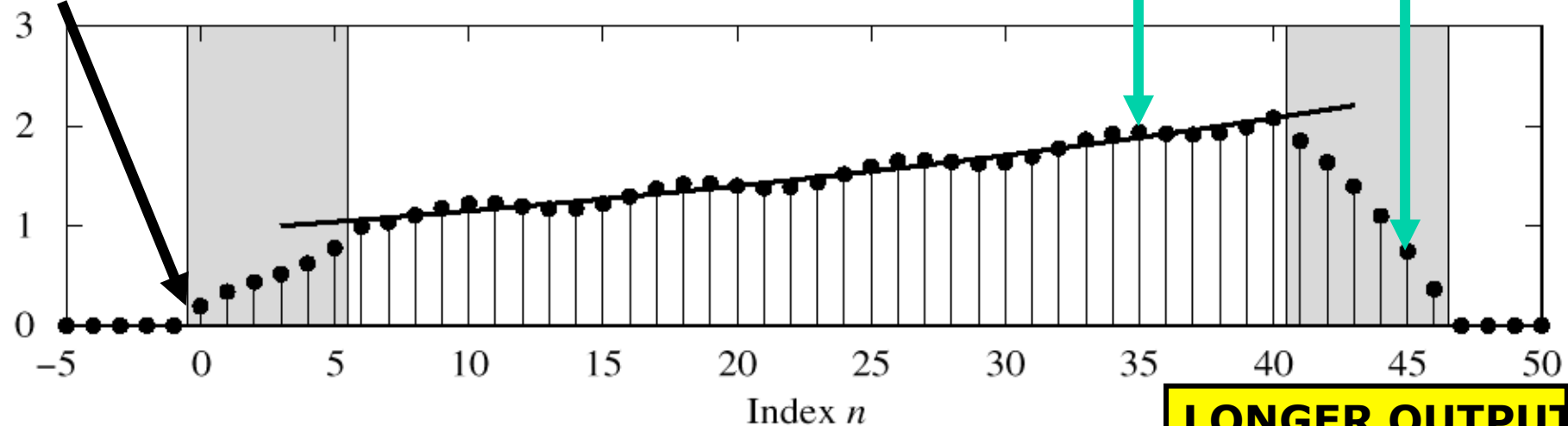
# 7-pt FIR EXAMPLE (AVG)

Input :  $x[n] = (1.02)^n + \cos(2\pi n/8 + \pi/4)$  for  $0 \leq n \leq 40$



**CAUSAL: Use Previous**

Output of 7-Point Running-Average Filter



**LONGER OUTPUT**



*That's all Folks!*

- Next week

<>

Section 5-4  
Section 5-5  
Section 5-6  
Section 5-7