EE-2200 Spring-99

Lecture 9

FIR Filtering Intro

3-May-99

Information

- Music Listening this Thursday
  - Lab Quiz next week !!!!!!!
- Problem Set #4 due Friday
  - On-Line HW due on Wed at Noon.
- MATLAB help: Wed @ 6pm, VL-456

- Quiz #2 on 24-May (Monday)

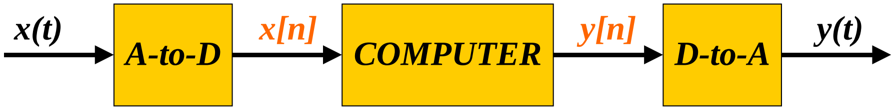
READING ASSIGNMENTS

- This Lecture:
  - Chapter 5, pp. 119-131
- Other Reading:
  - Recitation: Ch. 5, pp. 127-133, 142-146
    - CONVOLUTION
  - Next Lecture: Chapter 5, pp. 133-152

LECTURE OBJECTIVES

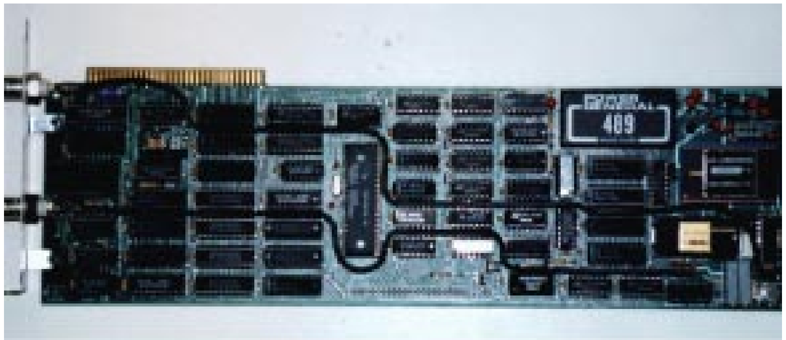
- INTRODUCE FILTERING IDEA
  - Weighted Average
  - Running Average
- FINITE IMPULSE RESPONSE FILTERS
  - FIR Filters
  - Show how to compute the output y[n] from the input signal, x[n]

DIGITAL FILTERING



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

The TMS32010, 1983

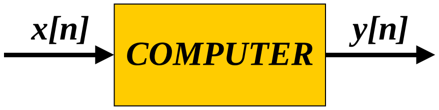


First PC plug-in board from Atlanta Signal Processors Inc.

Rockland Digital Filter, 1971

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

DISCRETE-TIME SYSTEM



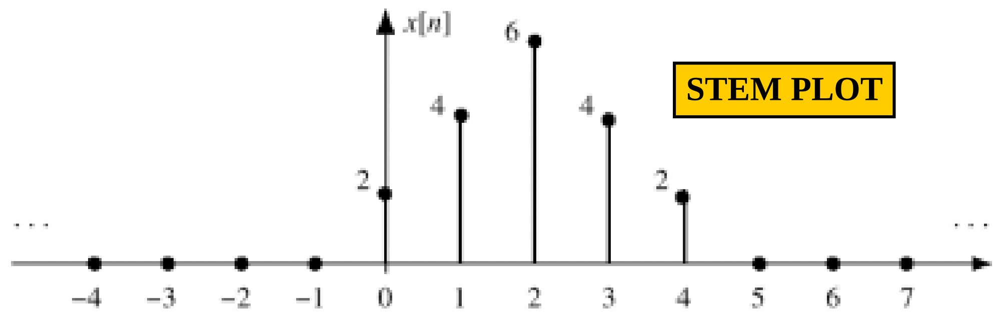
- OPERATE on x[n] to get y[n]
- WANT GENERAL CLASS of SYSTEMS
  - ANALYZE the SYSTEM
    - TOOLS: TIME-DOMAIN & FREQUENCY-DOMAIN
  - SYNTHESIZE the SYSTEM

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**

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

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

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

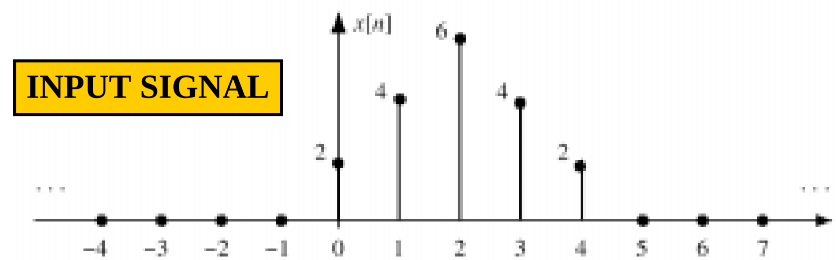


Figure 5.2 Finite-length input signal, $x[n]$.

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

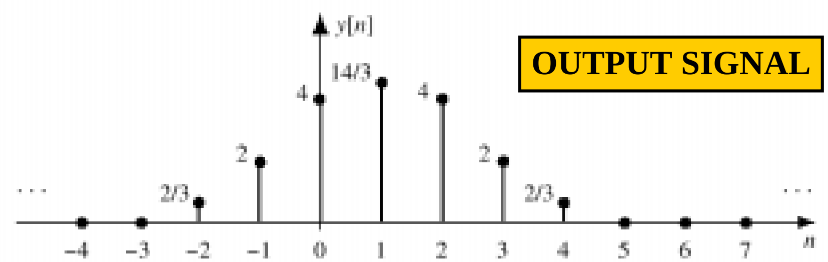


Figure 5.3 Output of running average, $y[n]$.

PAST, PRESENT, FUTURE

Sec. 5.2 The Running Average Filter **123**

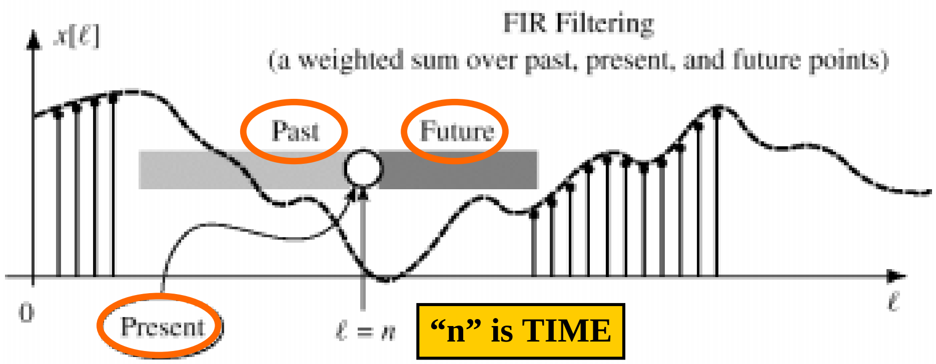


Figure 5.4 The running-average filter calculation at time index n uses values within a sliding window (shaded). Dark shading indicates the future ($\ell > n$); light shading, the past ($\ell < n$).

ANOTHER 3-pt AVERAGER

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

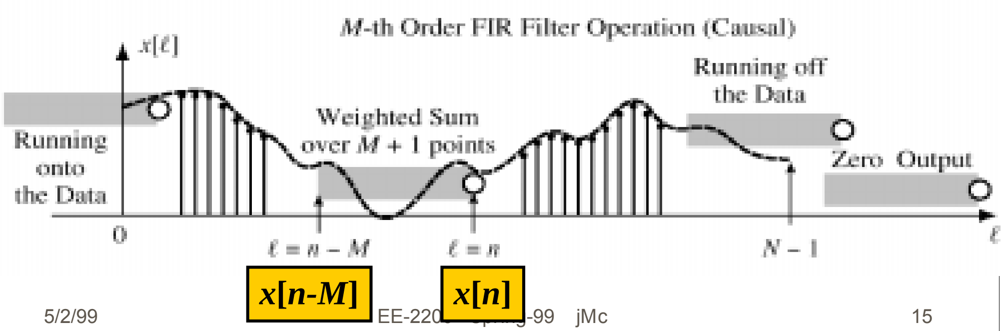
$$y[n] = \tfrac{1}{3}(x[n] + x[n-1] + x[n-2])$$

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

- SLIDE a WINDOW across x[n]

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



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

FILTERING EXAMPLE

- 7–point AVERAGER

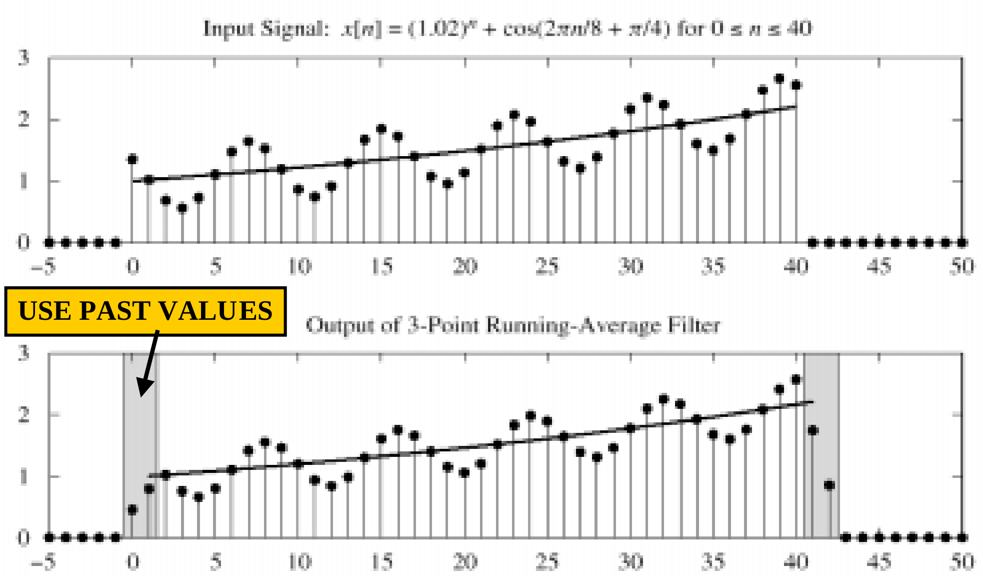
$$y_7[n] = \frac{1}{7}\left(\sum_{k=0}^{6} x[n-k]\right)$$

  - Removes cosine
    - By making its amplitdue (A) smaller
- 3–point AVERAGER

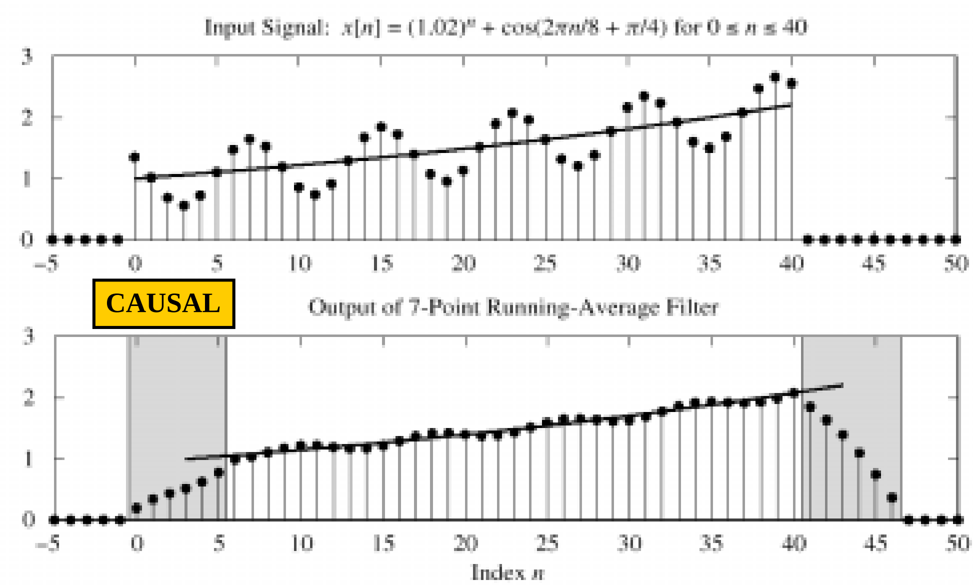
$$y_3[n] = \frac{1}{3}\left(\sum_{k=0}^{2} x[n-k]\right)$$

  - Changes A slightly

3-pt AVG EXAMPLE



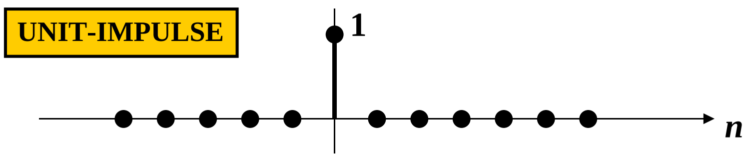
7-pt AVG EXAMPLE



SPECIAL INPUT SIGNALS

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

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



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

n=3

NON-ZERO When its argument is equal to ZERO

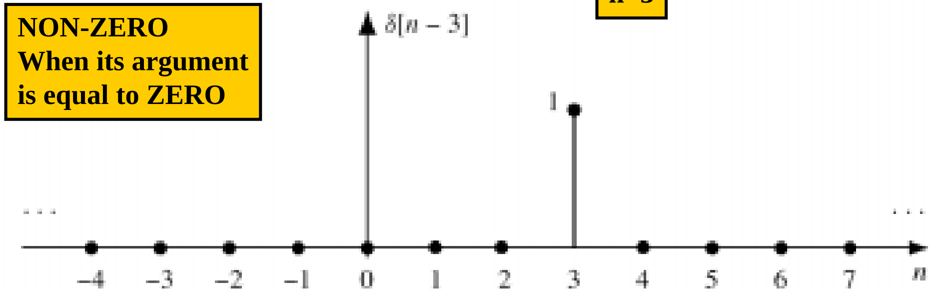
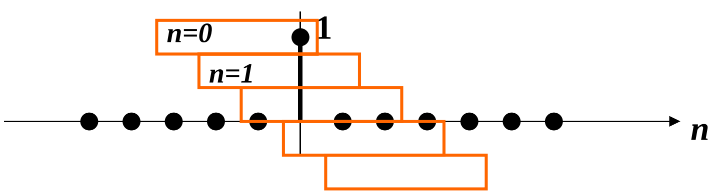


Figure 5.7 Shifted impulse sequence, $\delta[n-3]$.

4-pt AVERAGER

- CAUSAL SYSTEM: USE PAST VALUES
- INPUT = UNIT IMPULSE SIGNAL
- OUTPUT is called "IMPULSE RESPONSE"



4-pt Avg Impulse Response

- y[n] = 0.25(x[n]+x[n−1]+x[n−2]+x[n−3])
- "READS OUT" the FILTER COEFFICIENTS
- y[n] = {...,0,0,0.25,0.25,0.25,0.25,0,0,...}

↑ n=0

Use *h*[n] for Impulse Response