EE-2200 Fall-98

Lecture 10

Block Diagram Structures

30-Oct-98

Info: Web-CT, Lab, HW

- Calendar:
  - Quiz #2 on 13-Nov (Friday)
- Prob Set #4 due MONDAY
  - Solutions for #3 are posted (partial)
- Lab #5 on Filtering
  - Frequency Response

READING ASSIGNMENTS

- This Lecture:
  - Chapter 5, pp. 133-152
- Other Reading:
  - Recitation: Ch. 5, pp. 127-133, 142-146
    - CONVOLUTION
  - Next Lecture: Chapter 6, start

LECTURE OBJECTIVES

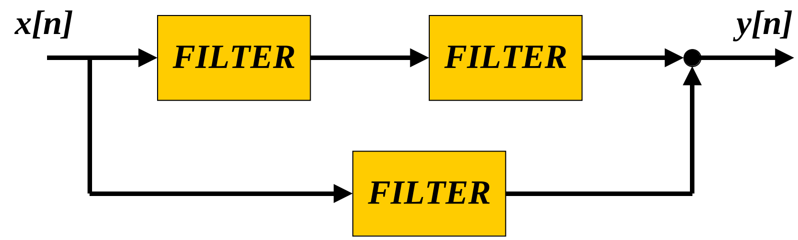
- BLOCK DIAGRAM REPRESENTATION
  - Components for Hardware
  - Connect Simple Filters Together to Build More Complicated Systems
- GENERAL PROPERTIES of FILTERS
  - LINEARITY
  - TIME-INVARIANCE
  - ==> CONVOLUTION

LTI SYSTEMS

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

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

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

MATLAB for FIR FILTER

- `yy = filter(bb,1,xx)`
  - VECTOR bb contains Filter Coefficients
  - DSP-First: `yy = firfilt(bb,xx)`
- FILTER COEFFICIENTS $\{b_k\}$

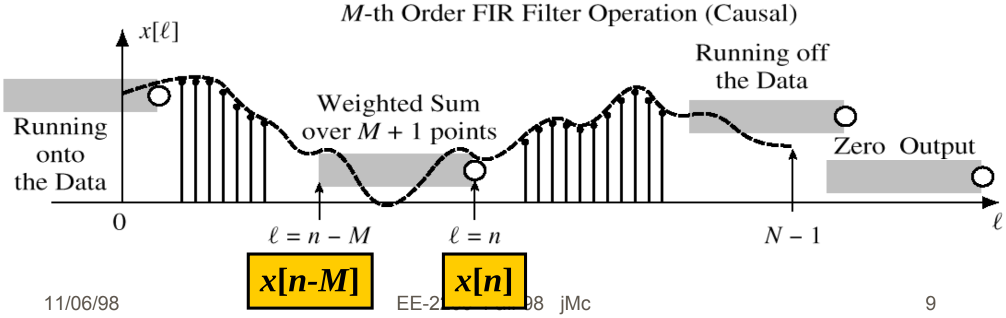
$$y[n] = \sum_{k=0}^{M} b_k \, x[n-k]$$

GENERAL FIR FILTER

- SLIDE a Length–L WINDOW over x[n]

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

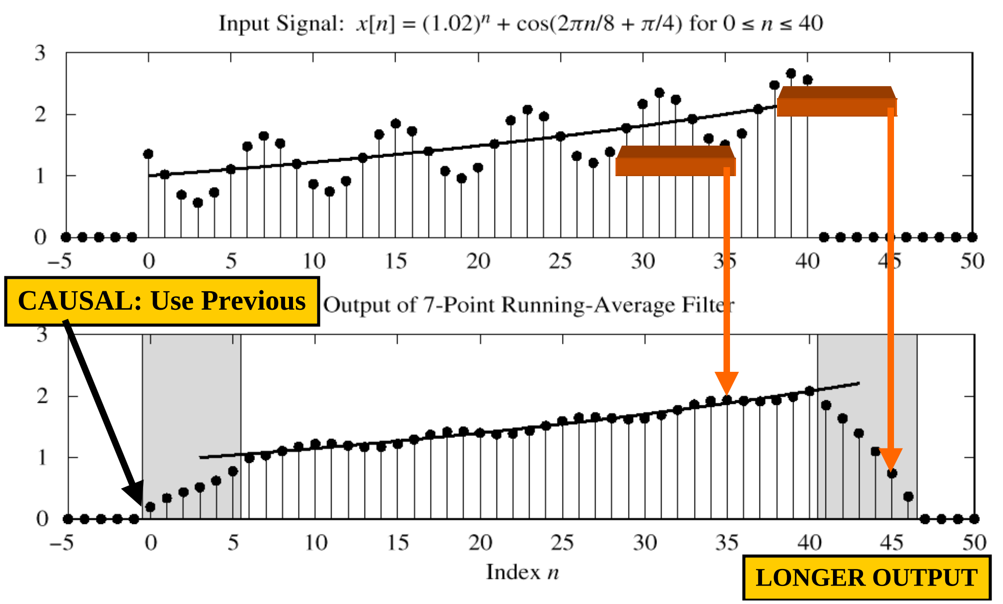
M-th Order FIR Filter Operation (Causal)



FILTERING EXAMPLE

- 7–point AVERAGER $\quad y_7[n] = \frac{1}{7}\left(\sum_{k=0}^{6} x[n-k]\right)$
  - Removes cosine
    - By making its amplitdue (A) smaller
- 3–point AVERAGER $\quad y_3[n] = \frac{1}{3}\left(\sum_{k=0}^{2} x[n-k]\right)$
  - Changes A slightly

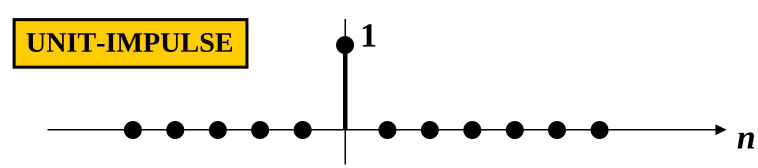
7-pt FIR EXAMPLE (AVG)



SPECIAL INPUT SIGNALS

- x[n] = SINUSOID **FREQUENCY RESPONSE**
- 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

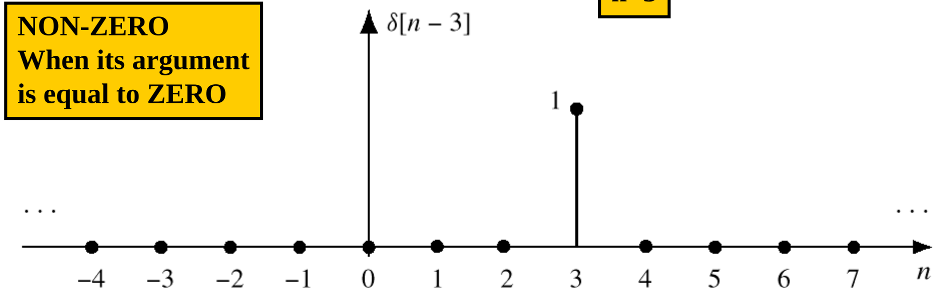
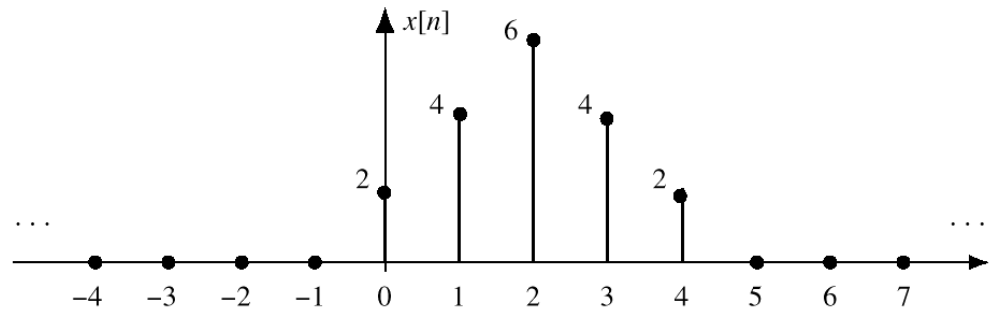


Figure 5.7 Shifted impulse sequence, $\delta[n-3]$.

MATH FORMULA for x[n]

- **Use 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 IMPULSES

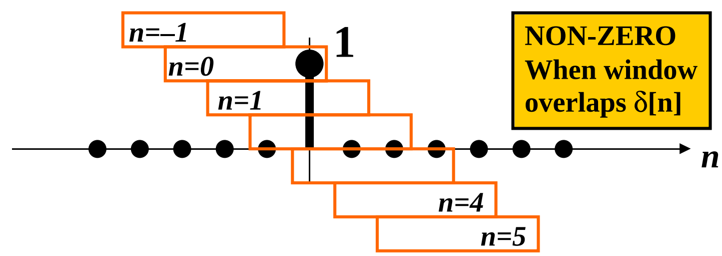
n	...	−2	−1	0	1	2	3	4	5	6	...
$2\delta[n]$	0	0	0	2	0	0	0	0	0	0	0
$4\delta[n-1]$	0	0	0	0	4	0	0	0	0	0	0
$6\delta[n-2]$	0	0	0	0	0	6	0	0	0	0	0
$4\delta[n-3]$	0	0	0	0	0	0	4	0	0	0	0
$2\delta[n-4]$	0	0	0	0	0	0	0	2	0	0	0
$x[n]$	0	0	0	2	4	6	4	2	0	0	0

$$x[n] = \sum_k x[k]\delta[n-k]$$ ← **This formula ALWAYS works**

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

4-pt AVERAGER

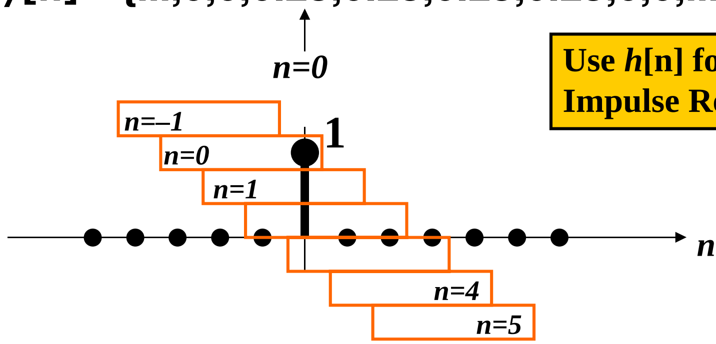
- **CAUSAL SYSTEM: USE PAST VALUES**
- **INPUT = UNIT IMPULSE SIGNAL = $\delta[n]$**
- **OUTPUT is called "IMPULSE RESPONSE"**



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

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,...}



Use *h*[n] for Impulse Response

FIR IMPULSE RESPONSE

- Convolution = Filter Definition
  - Filter Coeffs = Impulse Response

n	$n<0$	0	1	2	3	$\dots$	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	$\dots$	b_M	0	0

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

HARDWARE STRUCTURES

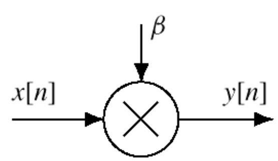
$$y[n] = \sum_{k=0}^{M} b_k\,x[n-k]$$

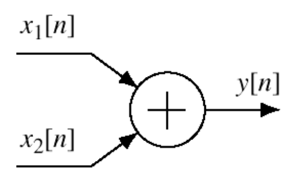
- INTERNAL STRUCTURE of "FILTER"
  - WHAT COMPONENTS ARE NEEDED?
  - HOW DO WE "HOOK" THEM TOGETHER?
- SIGNAL FLOW GRAPH NOTATION

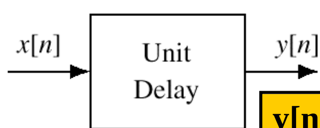
HARDWARE ATOMS

- Add, Multiply & Store

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







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

FIR STRUCTURE

- **Direct Form**

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

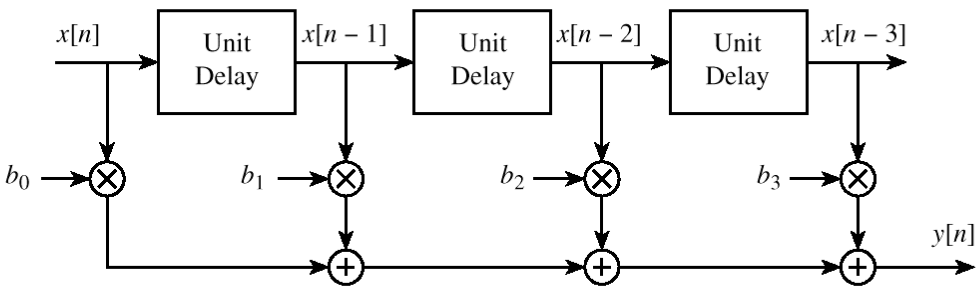
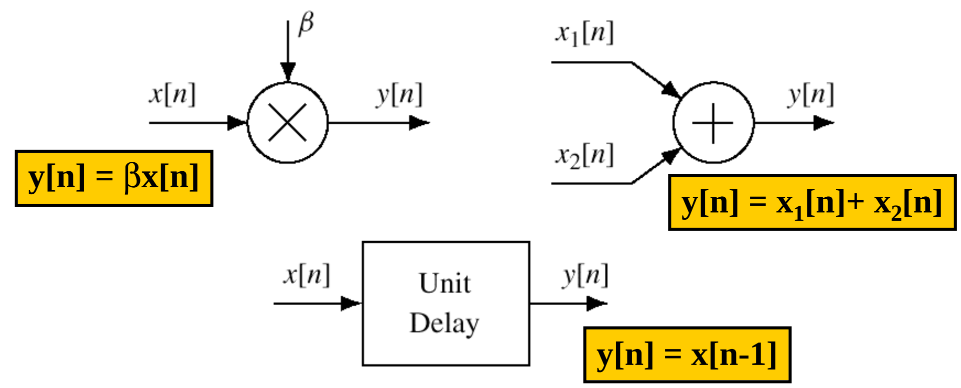


Figure 5.13 Block-diagram structure for the Mth order FIR filter.

SIGNAL FLOW GRAPH

- **WRITE EQUATIONS @ EACH ELEMENT**



FLOW GRAPH --> FIR

- **Starting from the Block Diagram, write the FIR Filtering Equation:**

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

- **PROCEDURE:**
  - **NAME** the signals at the **INPUT** to the delay elements
  - Write an equation at the **OUTPUT** of each adder

ALTERNATE STRUCTURE

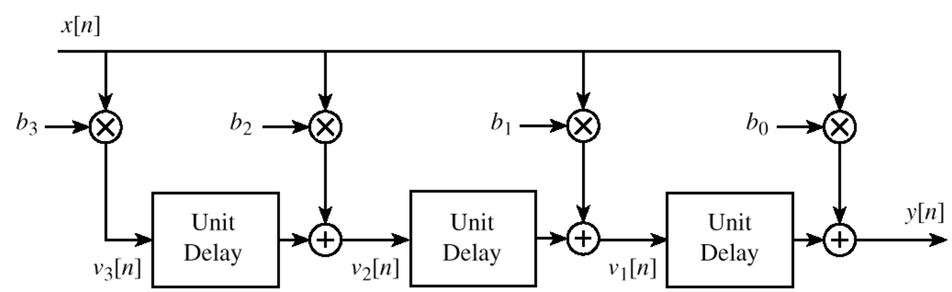


Figure 5.14 Transposed form block diagram structure for the Mth order FIR filter.

$$y[n] = b_0 x[n] + v_1[n-1]$$
$$v_1[n] = b_1 x[n] + v_2[n-1]$$
$$v_2[n] = b_2 x[n] + v_3[n-1]$$
$$v_3[n] = b_3 x[n]$$

DIFFERENT COMPUTATION

TDF DERIVATION (M=3)

$$y[n] = b_0 x[n] + v_1[n-1]$$
$$v_1[n] = b_1 x[n] + v_2[n-1]$$
$$v_2[n] = b_2 x[n] + v_3[n-1]$$
$$v_3[n] = b_3 x[n]$$

$$y[n] = b_0 x[n] + b_1 x[n-1] + v_2[n-2]$$
$$v_2[n] = b_2 x[n] + b_3 x[n-1]$$
$$\Longrightarrow \quad y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] + b_3 x[n-3]$$

SYSTEM PROPERTIES

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

TIME-INVARIANCE

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

TESTING Time-Invariance

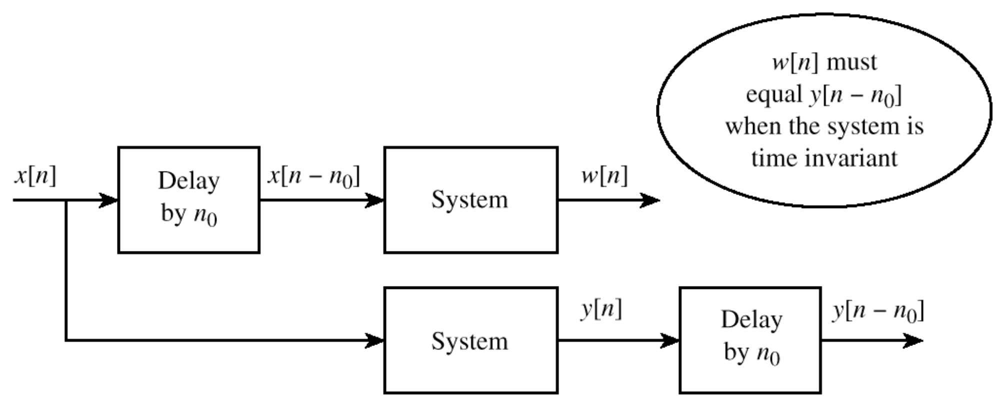


Figure 5.16 Testing time-invariance property by checking the interchange of operations.

LINEAR SYSTEM

- 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