EE-2025 Fall-2000

Lecture 10
Linearity & Time-Invariance
29-Sept-00

Info: Web-CT, Lab, HW

- **Honor Code and Lab Reports !!!**
- Lab Quiz next week during Lab #6
  - MATLAB questions about the Labs
- Labs #5 and #6: Image Processing
  - Sampling, then Filtering: Blurring & Sharpening (De-Blur)
- Quiz #2 on 20-Oct
  - Prob Sets #3, #4, #5, #6 and #7

LAB IMAGES (TRUE)

- Getting TRUE SIZE comparisons is hard: use an image display program

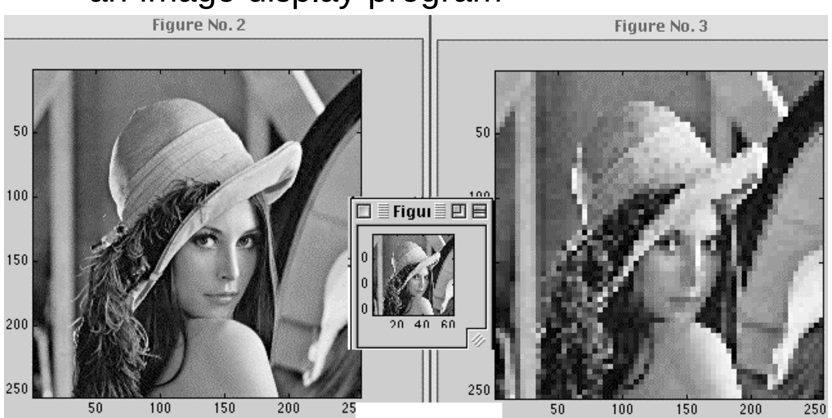


Image Display Procedure

- Example: 256 by 256 Lenna (512 is OK)
- Make MATLAB Figures in separate windows
- ALT-PRINT-SCREEN captures the active window (Win-95)
- Paste into “Paint” program
  - Under Win-95 Accessories
- Print after arranging images

LECTURE

EE-2025 Georgia Tech

Lecture 10
Linearity & Time-Invariance

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

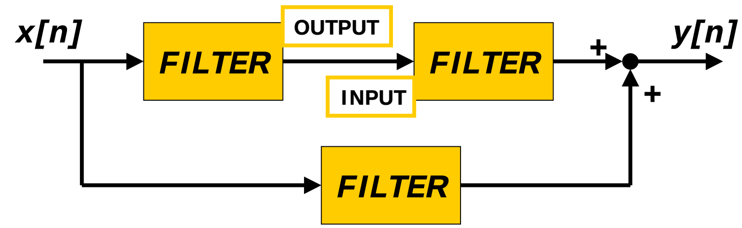
OVERVIEW

- IMPULSE RESPONSE, h[n]
  - FIR case: same as $\{b_k\}$
- CONVOLUTION
  - GENERAL: y[n] = x[n]*h[n]
- GENERAL CLASS of SYSTEMS
  - LINEAR and TIME-INVARIANT
- ALL LTI 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, $\{b_k\} = \{3, -1, 2, 1\}$

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

MATLAB for FIR FILTER

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

conv2()
for images

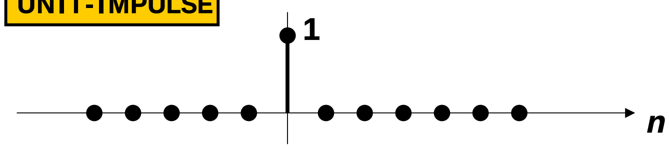
$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$

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



FIR IMPULSE RESPONSE

- Convolution = Filter Definition
  - Filter Coeffs = Impulse Response

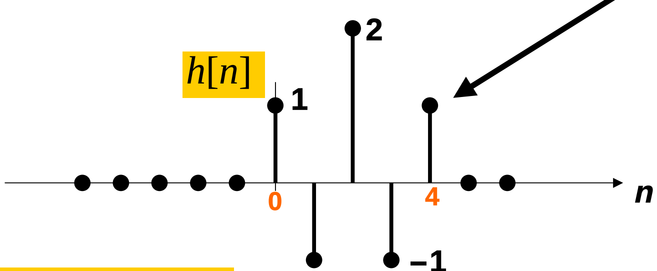
n	$n<0$	0	1	2	3	...	M	$M+1$	$n>M+1$
$x[n]=\delta[n]$	0	1	0	0	0	0	0	0	0
$y[n]=h[n]$	0	b_0	b_1	b_2	b_3	...	b_M	0	0

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

MATH FORMULA for h[n]

- Use SHIFTED IMPULSES to write h[n]

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



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

LTI: Convolution Sum

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

FINITE LIMITS

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

Same as $\mathbf{b_k}$

FINITE LIMITS

CONVOLUTION Example

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

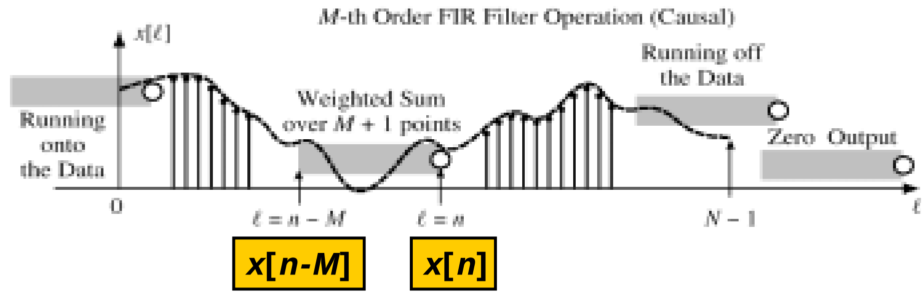
$$x[n] = u[n]$$

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
	0	1	1	1	1	1	1	1	1
	0	0	-1	-1	-1	-1	-1	-1	-1
	0	0	0	2	2	2	2	2	2
	0	0	0	0	-1	-1	-1	-1	-1
	0	0	0	0	0	1	1	1	1
$y[n]$	0	1	0	2	1	2	2	2	...

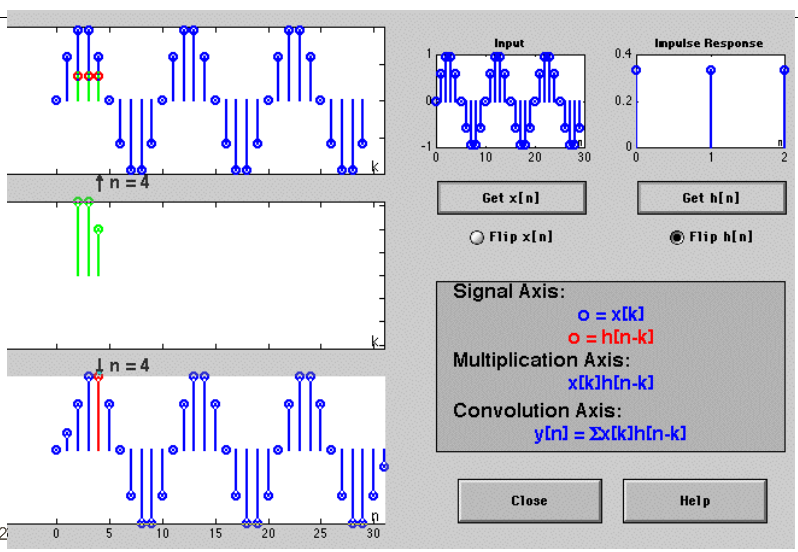
GENERAL FIR FILTER

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

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



CONVDEMO: MATLAB GUI



HARDWARE STRUCTURES

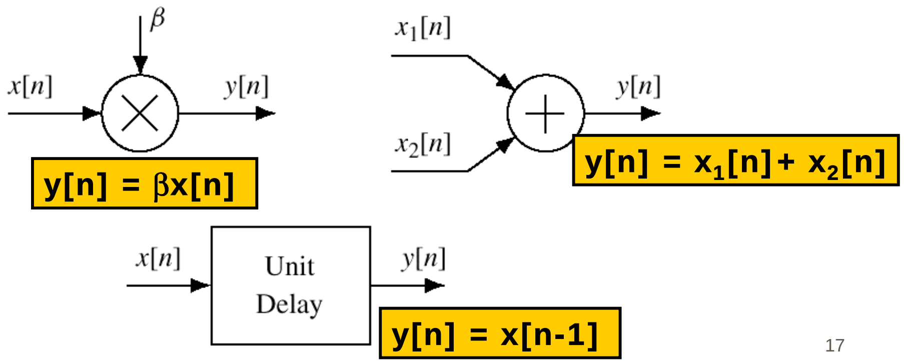
$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$

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

HARDWARE ATOMS

- Add, Multiply & Store

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



FIR STRUCTURE

- Direct Form

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

SIGNAL FLOW GRAPH

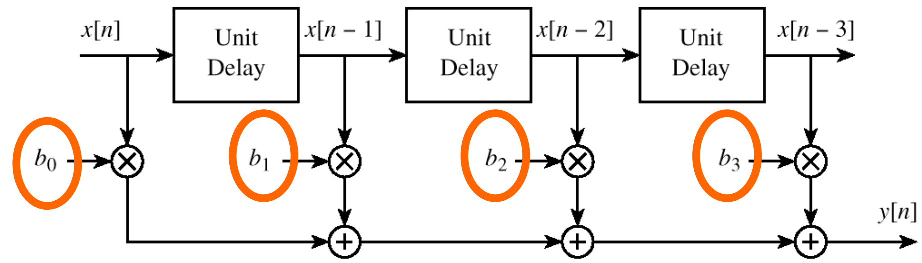
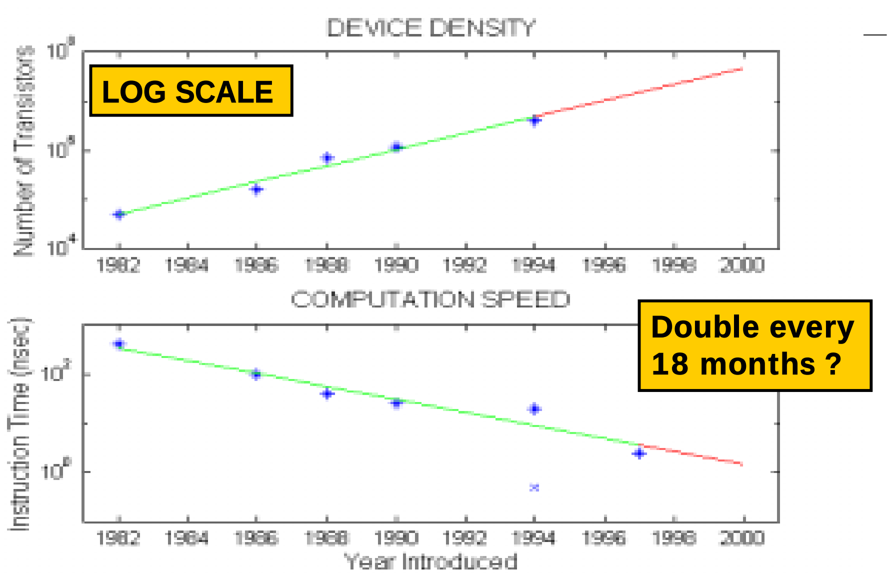


Figure 5.13 Block-diagram structure for the *M*th order FIR filter.

Moore's Law for TI DSPs



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

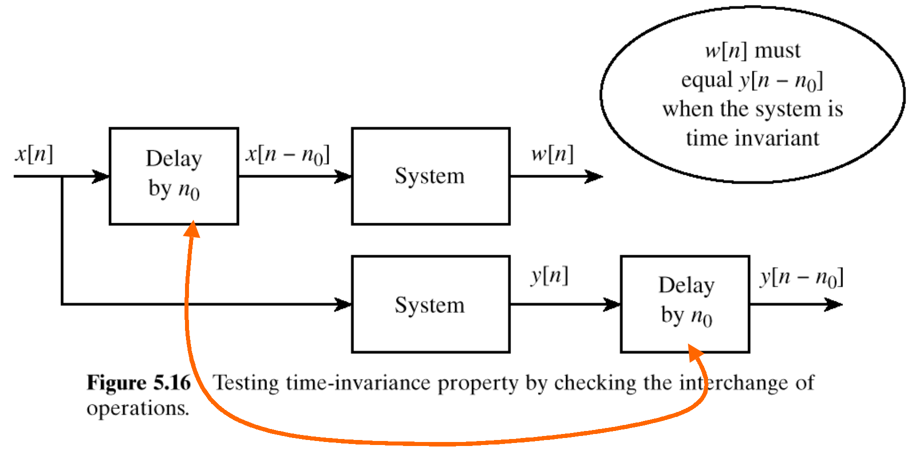


Figure 5.16 Testing time-invariance property by checking the interchange of operations.

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

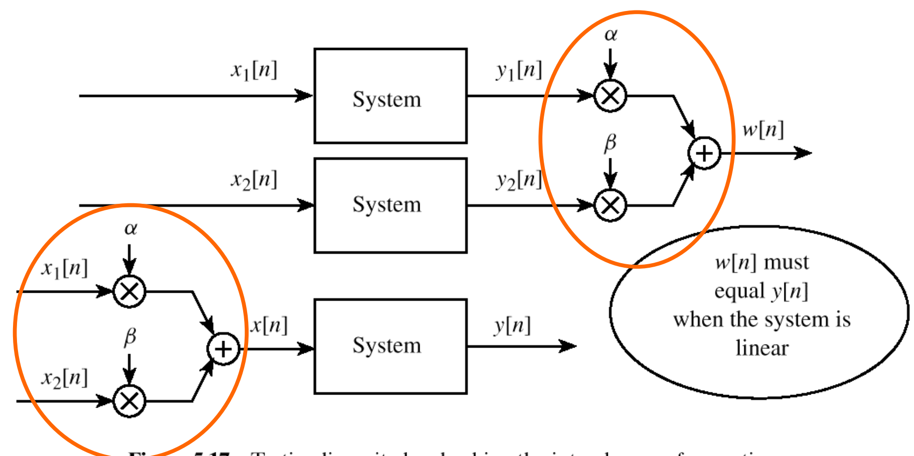


Figure 5.17 Testing linearity by checking the interchange of operations.

LTI SYSTEMS

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

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, another way to compute the output:

$$y[n] = (\delta[n] - \delta[n-1]) * x[n]$$

CASCADE SYSTEMS

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

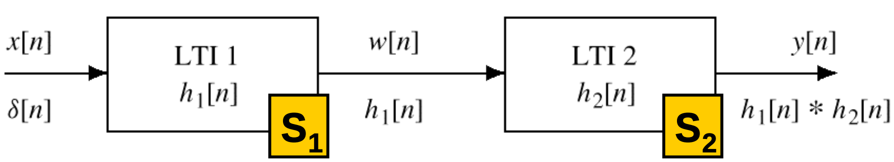


Figure 5.19 A Cascade of Two LTI Systems.

CASCADE EQUIVALENT

- Find “overall” h[n] for a cascade ?

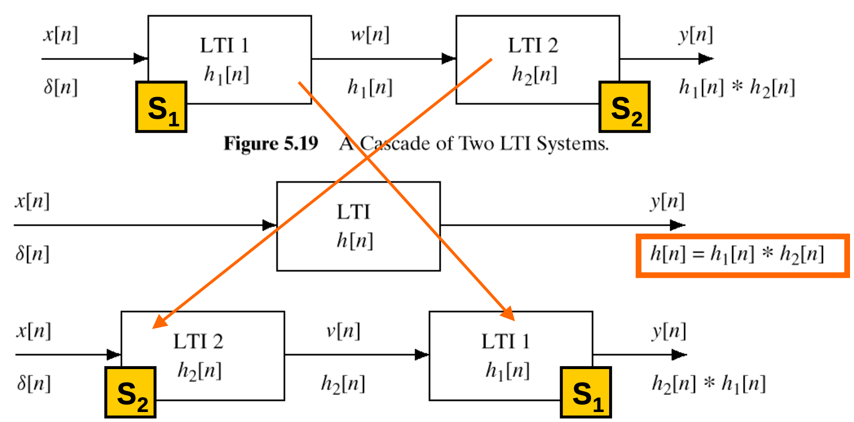


Figure 5.19 A Cascade of Two LTI Systems.

Figure 5.20 Switching the order of cascaded LTI systems.