Chapter2 Discrete-time signals and systems

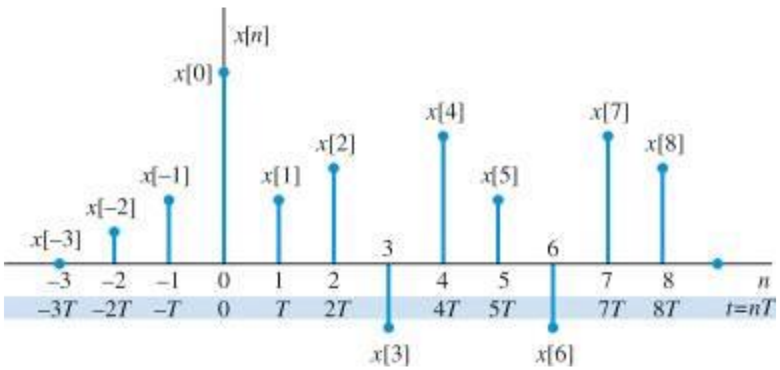
Prof. Kim, Jeongtae
Department of Electronics Engineering
Ewha Womans University

Discrete time signals and systems

- Describe discrete-time signals mathematically and generate, manipulate and plot discrete time signals using MATLAB
- Check whether a discrete-time system is linear, time-invariant, causal and stable, show that the input-output relationship of any linear time invariant system can be expressed by the convolution sum
- Determine analytically the convolution for sequences defined by simple formulas, write computer programs for the numerical computation of convolution
- Determine numerically the response of discrete-time systems described by linear constant coefficient difference equations

Discrete time signals

- A discrete-time signal $x[n]$ is a sequence of numbers defiend for every value of the integer variable n .
- We use $x[n]$ to represent either the n-th sample or entire sequence
- One may represent a signal in functional, tabular, sequence, and pictorial form
- Energy and power of a signal is defined as eq (1) and (2). Note that for a finite signal its energy is finite and power is always zero



$$E = \sum_{n=-\infty}^{\infty} |x[n]|^2 \qquad (1)$$

$$P = \lim_{L\to\infty} \{\frac{1}{2L+1} \sum_{n=-L}^{L} |x[n]|^2\} \qquad (2)$$

Elementary discrete-time signals

- Practical signals are complicated and it is not possible to represent a signal using mathematical functions
- Some simple signals useful in the representation and analysis of discrete-time signals and systems
- Unit sample sequence:

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

- Unit step sequence

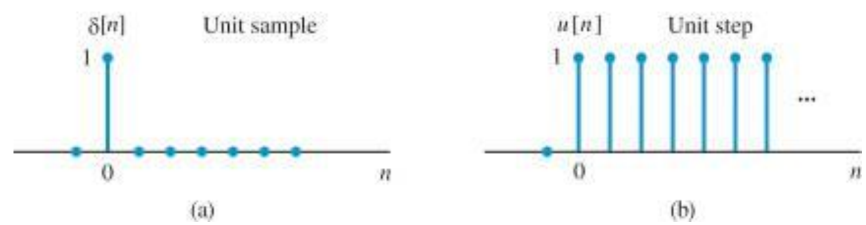
$$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

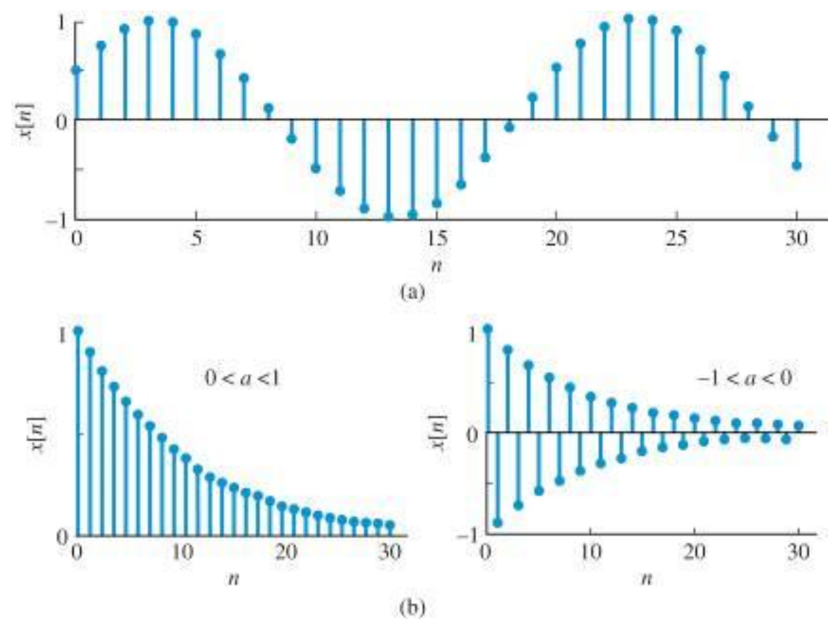
- Sinusoidal sequence

$$x[n] = A\cos(\omega_0 n + \phi)$$

- Exponential sequence

$$x[n] = Aa^n$$





Periodic sequence

- A sequence $x[n]$ is called periodic if

$$x[n] = x[n+N], \qquad \text{all} \quad n$$

- The smallest N is known as the fundamental period
- The sinusoidal sequence can be periodic, i.e.,

$$\cos(\omega_0 n + \phi) = \cos(\omega_0 n + \omega_0 N + \phi) \quad \text{if} \quad \omega_0 N = 2\pi k$$

- Unlike continuous case, not every sinusoidal sequence is periodic
- Note that all sinusoidal sequences having the following frequencies are indistiguishable (identical)

$$\omega_k = \omega_0 + 2k\pi, \qquad -\pi \le \omega_0 \le \pi$$

- Therefore the largest discrete frequency is $\omega_0 = \pi$, which corresponds to $N = 2$

Discrete-time systems

- A discrete-time system is a computational process or algorithm that transforms a sequence $x[n]$ into another sequence $y[n]$
- We denote a discrete-time system by

$$y[n] = \mathcal{H}\{x[n]\}$$

- Example: moving average filter (often used for reducing Gaussian noise)

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

- Example: median filter (often used for reducing spikes, salt and pepper noise)

$$y[n] = \mathrm{median}\{x[n-1], x[n-2], x[n], x[n+1], x[n+2]\}$$

Causality and stability

- Definition: A system is called causal if the present value of the output does not depend on future values of the input
- If the output of a system depends on future values of its input, the system is non-causal
- For real-time processing, causality is a necessary condition
- In off-line application, causality is not necessary

- Definition: A system is called BIBO (Bounded Input Bounded Output) stable if every bounded input signal results in a bounded output signal. That is

$$|x[n]| \leq M_x < \infty \Rightarrow |y[n]| \leq M_y < \infty$$

- Example: the moving average system is stable (why?)
- Example: An accumulator system is unstable

Linearity and time invariance

- Definition: A system is called linear if and only if the following homogeneity and additivity conditions are satisfied

$$\mathcal{H}\{a_1 x[n] + a_2 x_2[n]\} = a_1 \mathcal{H}\{x_1[n]\} + a_2 \mathcal{H}\{x_2[n]\}$$

- A system which is not linear is called nonlinear system
- Linear system cannot produce an output without being excited
- Linearity simplifies the analysis of a system

- A system is called time-invariant or fixed if and only if

$$y[n] = \mathcal{H}\{x[n]\} \Rightarrow y[n - n_0] = \mathcal{H}\{x[n - n_0]\}$$

- In other words, a time shift in the input results in a corresponding time shift in the output
- Example:

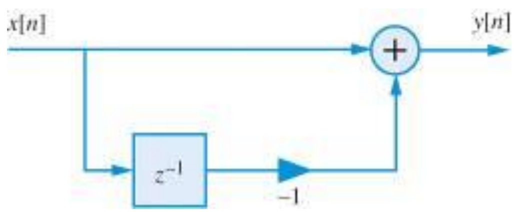
$$y[n] = x[n] \cos \omega_0 n$$

- Example: A down-sampler system

$$y[n] = \mathcal{H}\{x[n]\} = x[nM]$$

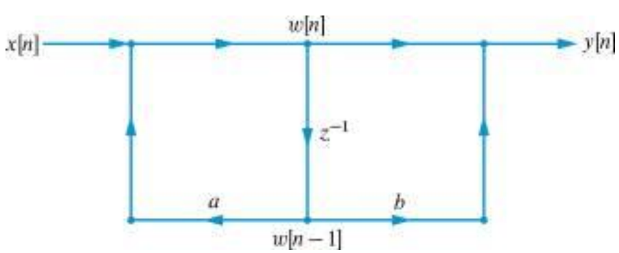
Block diagrams, signal flow graphs, and practical realizability

- Implementation of a discrete-time system can be depicted in a block diagram or signal flow graph
- Block diagram example:



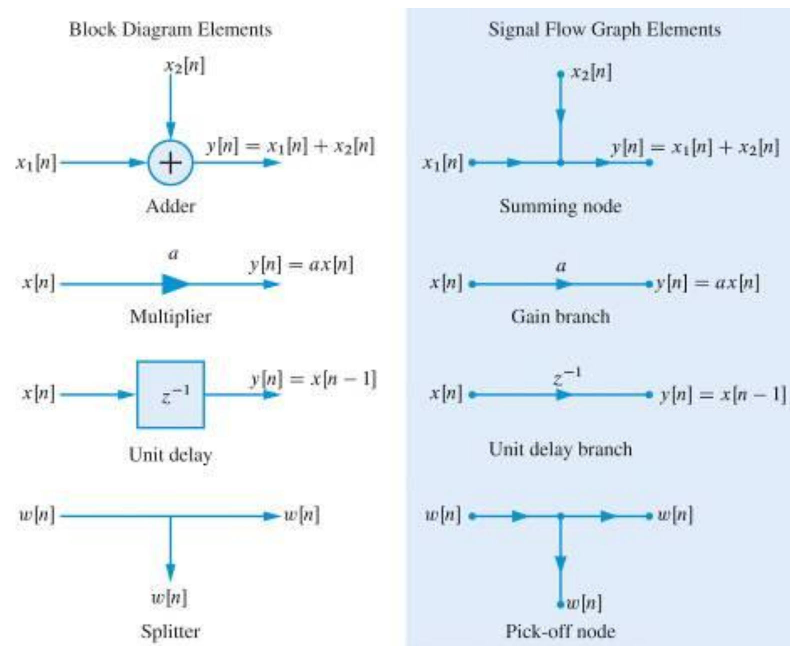
$$y[n] = x[n] - x[n-1]$$

- Signal flow graph example:



$$y[n] = x[n] + bx[n-1] + ay[n-1]$$

- Basic building blocks

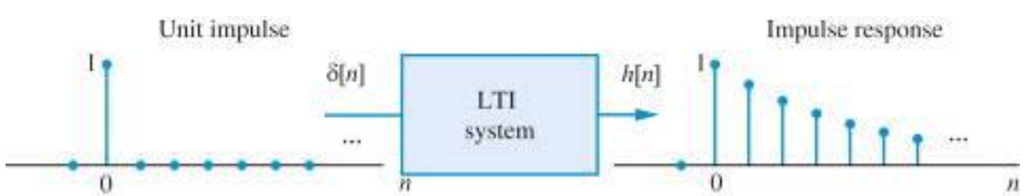


Convolution description of linear time-invariant systems

- The response of linear time invariant (LTI) system to any input can be determined from its impulse response, using convolution summation
- Thanks to the linearity, we have

$$
\begin{aligned}
x[n] &= \sum_k a_k x_k[n] = a_1 x_1[n] + a_2 x_2[n] + \dots \\
y[n] &= \sum_k a_k y_k[n] = a_1 y_1[n] + a_2 y_2[n] + \dots
\end{aligned}
$$

- We may compute the response of LTI system to each basic signal and synthesize the overall output from the individual response
- Impulse and complex exponential sequences are very important two basic sequences



Signal decomposition into impulses and convolution sum

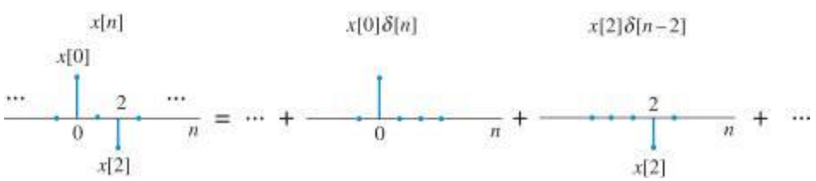
- Signal decomposition into impulses

$$x_k[n] = \begin{cases} x[k] & n = k \\ 0 & n \neq k \end{cases}$$

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

- A sequence can be represented by

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$



- Convolution sum
- Let $h_k[n]$ be the impulse response to $\delta[n-k]$
- Then the output is

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h_k[n]$$

- If we impose the additional constraint of time-invariance

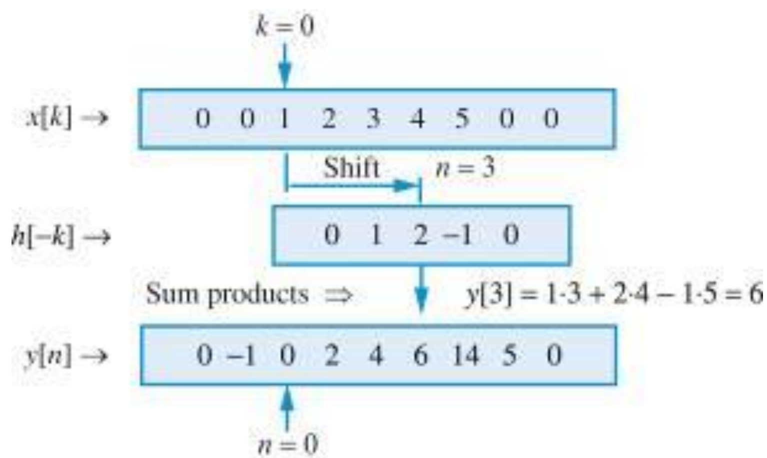
$$\delta[n] \rightarrow h[n] \Rightarrow \delta[n-k] = h_k[n] = h[n-k]$$

- Therefore, output can be computed using the following convolution sum

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k], \qquad -\infty < n < \infty$$

Understanding the convolution sum

- The samples of the sequence $x[k]$ are in natural order whereas the sample of the sequence $h[k]$ are in reverse order
- To determine the value of $y[n]$ for $n = n_0$, the flipped impulse sequence is shifted so that the sample $h[0]$ is aligned to the sample $x[n_0]$



$$x[n] = \{1, 2, 3, 4, 5\}, h[n] = \{-1, 2, 1\}$$

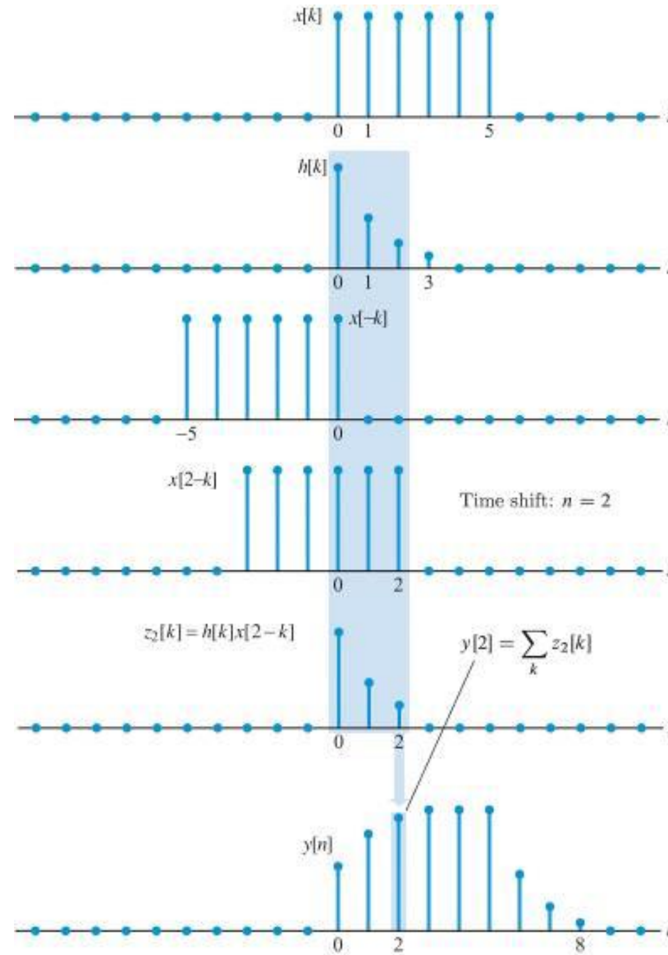
Understanding the convolution sum

- Change the index of sequences from n to k
- Flip the sequence $h[k]$ about $k = 0$
- Shift the flipped sequence by n samples
- Multiply the sequence $x[k]$ and $h[n-k]$

to obtain the sequence $z_n[k] = x[k]h[n-k]$

- Sum all the samples of $z_n[k]$ to determine the output sample at the given value of the shift
- Repeat the above steps for all desired values of n
- Note that interchanging input and the impulse response does not change the output of the convolution sum

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



Convolution as a superposition of scaled and shifted replicas

- The output of a LTI system can be viewed the sum of scaled and shifted impulse responses
- The scale factor is the value of input sequence

$$\delta[n] \xrightarrow{\mathcal{H}} h[n] \qquad \text{(Impulseresponse)}$$

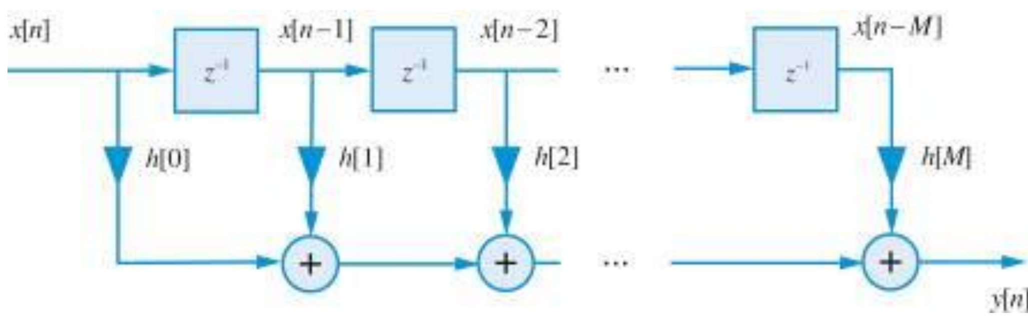
$$\delta[n-k] \xrightarrow{\mathcal{H}} h[n-k] \qquad \text{(Time-Invariance)}$$

$$x[k]\delta[n-k] \xrightarrow{\mathcal{H}} x[k]h[n-k] \qquad \text{(Homogeneity)}$$

$$\underbrace{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]}_{x[n]} \xrightarrow{\mathcal{H}} \underbrace{\sum_{k=-\infty}^{\infty} x[k]h[n-k]}_{y[n]} \qquad \text{(Additivity)}$$

FIR versus IIR system

- The duration of the impulse response leads to two different types of linear time-invariant system
- If the impulse response has a finite number of nonzero samples (finite support), we have a FIR (Finite Impulse Response) system
- Otherwise, we have a IIR (Infinite Impulse Response) system
- A class of IIR systems can be realized using a finite amount of memory and arithmetic operations



Properties of linear time-invariant system

- Commutative

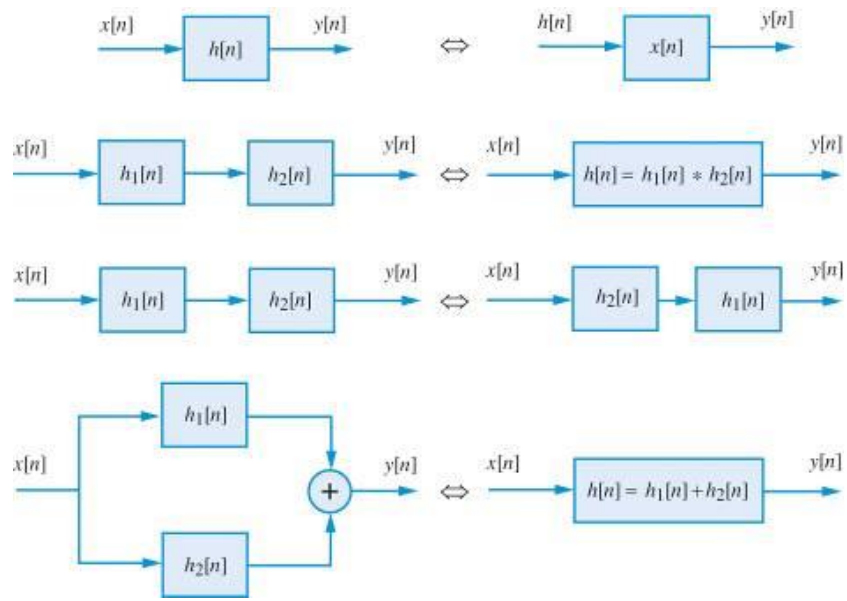
$$h[n] * x[n] = x[n] * h[n]$$

- Cascade interconnection

$$\begin{aligned} v[n] &= \sum_{k=-\infty}^{\infty} x[k] h_1[n-k] \\ y[n] &= \sum_{m=-\infty}^{\infty} h_2[m] v[n-m] \\ y[n] &= \sum_{k=-\infty} x[k] \sum_{m=-\infty}^{\infty} h_2[m] h_1[(n-k)-m]. \end{aligned}$$

- Parallel interconnection

$$y[n] = h_1[n] * x[n] + h_2[n] x[n] = (h_1[n] + h_2[n]) * x[n]$$



Causality and Stability

- Since a linear time-invariant system is completely characterized by its impulse response, causality and stability can be checked using impulse response
- A linear time-invariant system is causal if

$$h[n] = 0, \text{for } n < 0$$

- A linear time-invariant system is BIBO stable if and only if the impulse response is absolutely summable, i.e.,

$$\sum_{n=-\infty}^{\infty} |h[n]| < \infty$$

- Proof
- FIR systems are always stable

Response to simple test sequences

- To understand the behavior of LTI system, we study its effect on some simple test signals. Then we can use the principle of superposition to understand its effect on more complicated signals
- The impulse response is very useful
- Consider the response to the sequence

$$\begin{aligned} x[n] &= a^n, \quad -\infty < n < \infty \\ y[n] &= \sum_{k=-\infty}^{\infty} h[k]a^{n-k} = \left(\sum_{k=-\infty}^{\infty} h[k]a^{-k} \right) a^n \end{aligned}$$

- An important case occurs for $a = e^{j\omega}$

$$y[n] = \left(\sum_{k=-\infty}^{\infty} h[k]e^{-j\omega k} \right) e^{j\omega n} = H(e^{j\omega})e^{j\omega n}$$

Analytic evaluation of convolution

- To compute the output, we need to sum all nonzero values of the produce sequence

$$h[k]x[n-k], -\infty < k < \infty$$

- For finite input sequence and impulse response sequence, one can compute the result of convolution graphically
- Partial overlap (left)

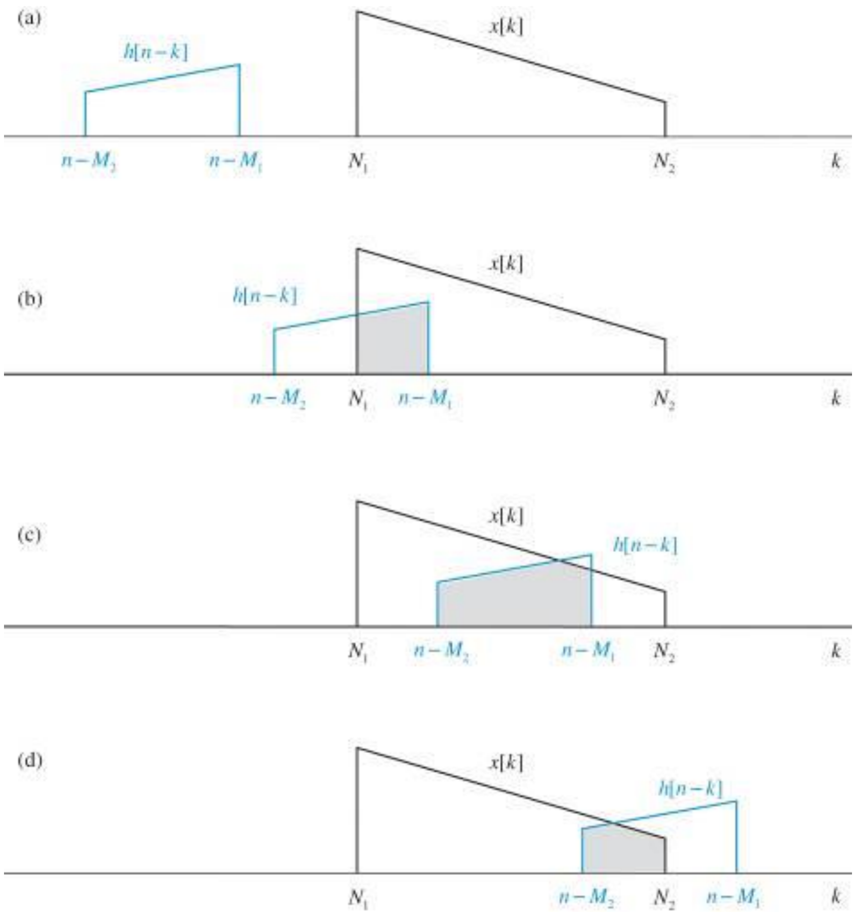
$$y[n] = \sum_{k=N_1}^{n-M_1} x[k]h[n-k], \text{for} N_1 + M_1 \le n \le N_1 + M_2$$

- Full overlap

$$y[n] = \sum_{k=n-M_2}^{n-M_1} x[k]h[n-k], \text{for} N_1 + M_1 < n < M_1 + N_2$$

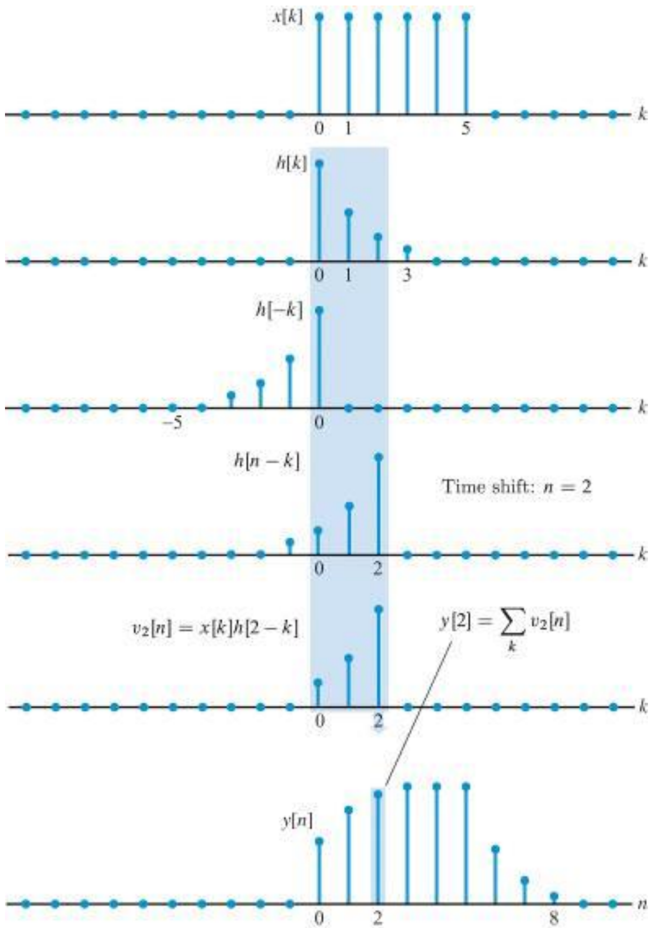
- Partial overlap (right)

$$y[n] = \sum_{k=n-M_2}^{N_2} x[k]h[n-k], \text{for} M_1 + N_1 < n < M_2 + N_2$$



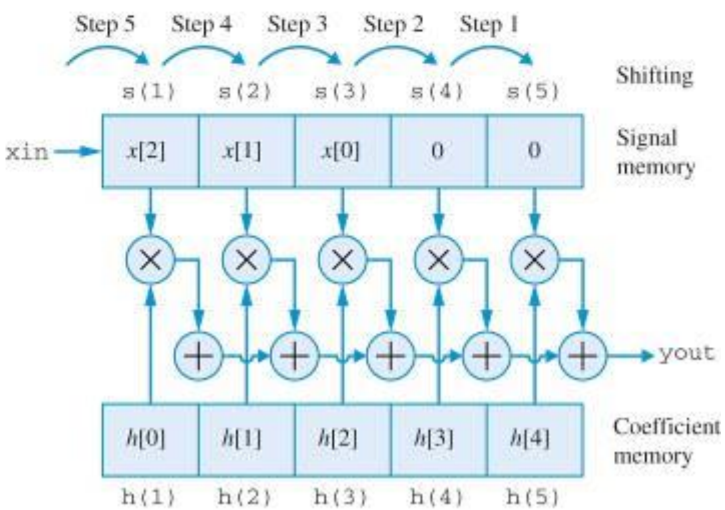
Graphical convolution example

- Convolution of two finite sequences



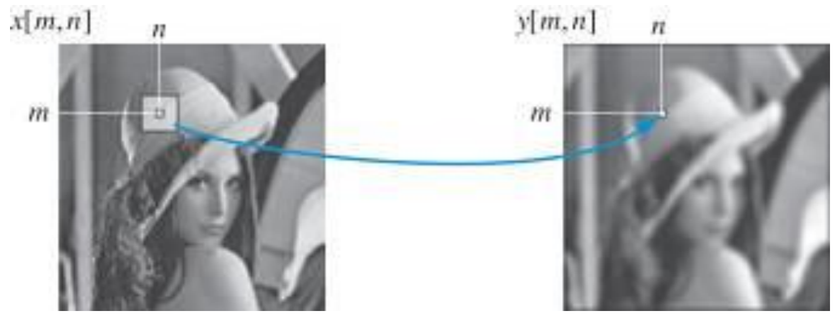
Real-time implementation of FIR filters

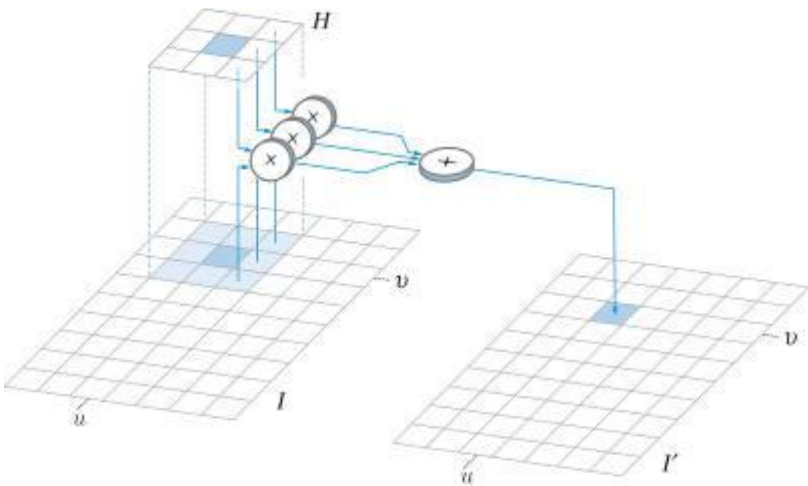
- In most real-time applications, we wish to compute the output sample immediately after the arrival of the input sample
- This approach, which proceeds on a sample-by-sample basis upon the input sequence, is known as stream processing



FIR spatial filters

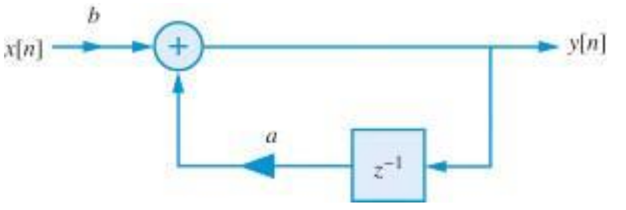
- A digital image can be represented by a 2D discrete space signal
  $x[m,n] \in \{(0, M-1) \times (0, N-1)\}$
- FIR low pass filtering
  $$y[m,n] = \sum_{k=-K}^{K} \sum_{l=-L}^{L} h[k,l]x[m-k,n-l]$$
- If all the value of the impulse response is the same, the system is moving average low pass filter
- Filter implementation
  $$y[m,n] = \sum_{k=-K}^{K} \sum_{l=-L}^{L} x[k,l]h[m-k,n-l]$$
- Filter array is rotated by 180 degree.
- The rotated array is moved over image so that the origin of the impulse response coincides with the current pixel
- All filter coefficients are multiplied with image pixels and the results are added





Linear constant-coefficient difference equations

- LTI system man be described by a linear constant coefficient difference equation
- Following two systems are identical



$$
\begin{aligned}
y[n] &= ay[n-1] + bx[n] \\
h[n] &= ba^n u[n]
\end{aligned}
$$

- Zero input response

$$y_{zi}[n] = a^{n+1} y[-1], \qquad n \geq 0$$

- Zero-state response of a causal system when input is applied at 0.

$$y_{zs}[n] = \sum_{k=0}^{n} h[k]x[n-k]$$

$$y[n] = y_{zi}[n] + y_{zs}[n]$$

Steady state and transient step responses

- Consider unit step response of a system

$$y[n] = \sum_{k=0}^{n} ba^k + a^{n+1}y[-1] = b\frac{1-a^{n+1}}{1-a} + a^{n+1}y[-1]$$

- Steady state response

$$y_{ss} = \lim_{n\to\infty} y[n] = b\frac{1}{1-a}$$

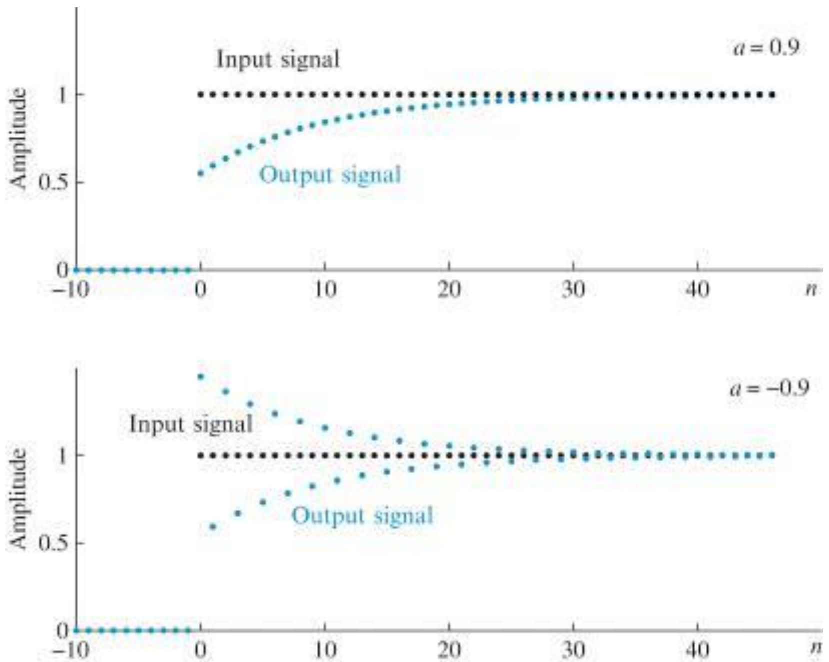
- Transient response

$$y_{tr}[n] = y[n] - y_{ss}[n] = b\frac{-a^{n+1}}{1-a} + a^{n+1}y[-1]$$

- In general

$$\begin{aligned} y_{zi}[n] &\neq y_{tr}[n] \\ y_{ss}[n] &\neq y_{zs}[n] \end{aligned}$$

- If a system is stable $\quad y_{ss}[n] = \lim_{n\to\infty} y_{zs}[n]$

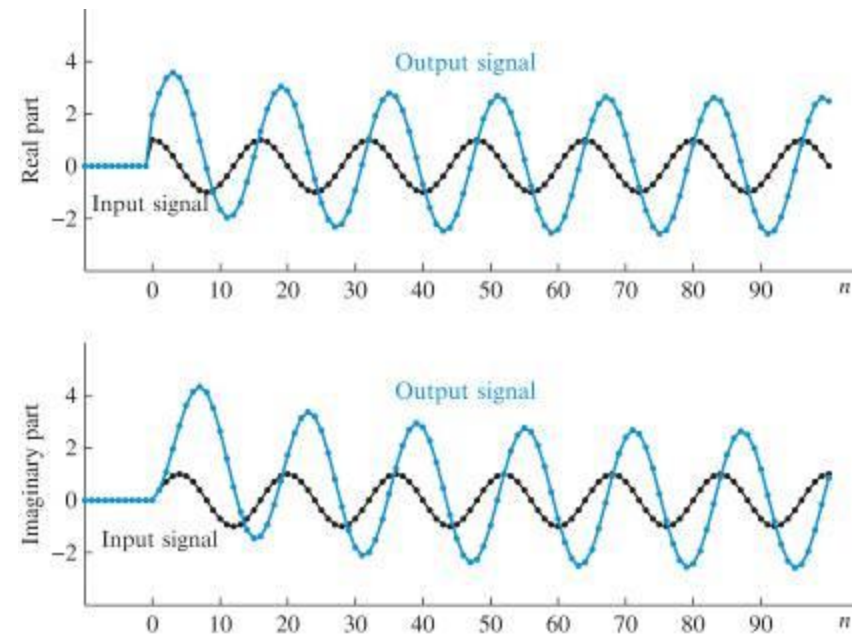


Response to a suddenly applied complex exponential sequences

- If we set $x[n] = e^{j\omega_0 n}$, output is computed as follows:

$$y[n] = a^{n+1}y[-1] + \frac{1 - a^{n+1}e^{-j\omega_0(n+1)}}{1 - ae^{-j\omega_0}}e^{j\omega_0 n}$$

- This can be splited into three terms
- The steady state response tracks the input
- The transient response reveals properties of the system but eventually dies out if the system is stable



$$y[n] = \underbrace{\overbrace{a^{n+1}y[n]}^{y_{zi}[n]} + \overbrace{\frac{-a^{n+1}e^{-j\omega_0(n+1)}}{1 - ae^{-j\omega_0}}e^{j\omega_0 n}}^{y_{zs}[n]}}_{y_{tr}[n]} + \underbrace{\overbrace{\frac{1}{1 - ae^{-j\omega_0}}e^{j\omega_0 n}}^{y_{zs}[n]}}_{y_{ss}[n]}$$