Discrete-Time Signals Systems (In Time-Domain)

Zhiliang Liu
Zhiliang_Liu@uestc.edu.cn

3/6/2019

§ 2.1 Discrete-Time Signals

2.1.1 Time-Domain Representation

- **Discrete time signals {*x*[*n*]}−−represented as *sequences* with *argument* *n* being an integer in the range $-\infty \le n \le \infty$**
- **Discrete-time signal represented by {*x*[*n*]}**
- **The *nth* sample value of sequence {*x*[*n*]} is also denoted as *x*[*n*]**
- **values of *argument* *n*−−defined only for integer and undefined for non-integer values of *n***

Often the braces are ignored to denote a sequence if there is no ambiguity.

$$\{x[n]\} = x[n]$$

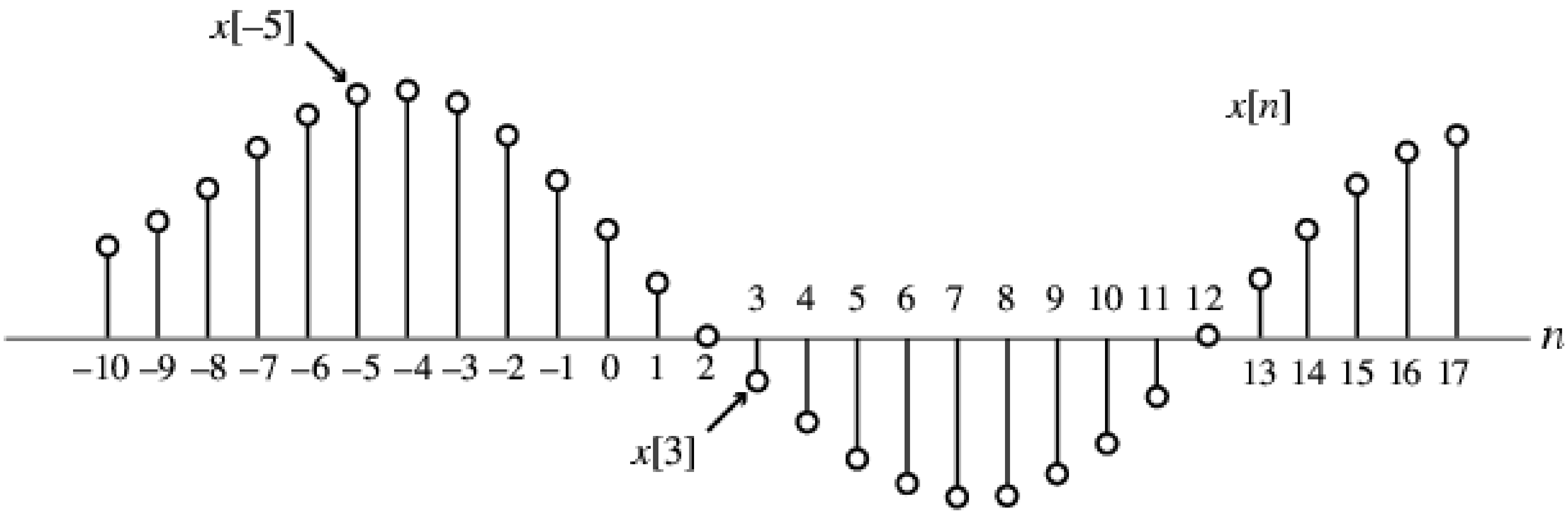
- **Discrete-time signal may also be written as a sequence of numbers inside braces:**

$$\{x[n]\}=\{\ldots,\ -0.2,\ \underset{\uparrow}{2.2},\ 1.1,\ 0.2,\ -3.7,\ 2.9,\ \ldots\}$$

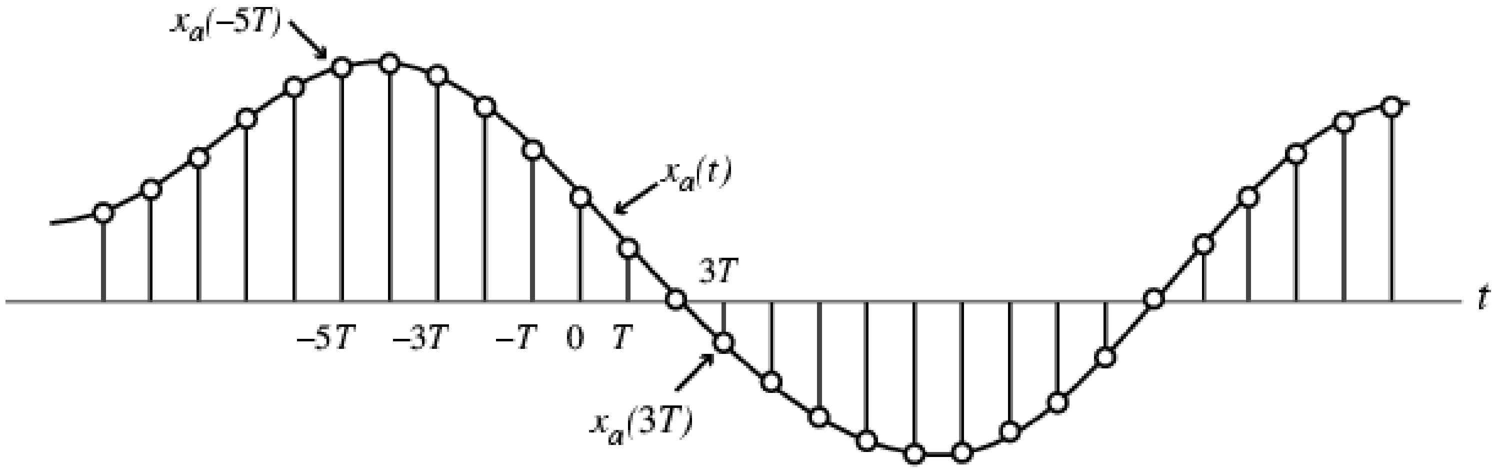
- **The arrow is placed under the sample at time index $n = 0$**
- **In the above, $x[-1]= -0.2$, $x[0]=2.2$, $x[1]=1.1$, etc.**

Vector form $x[n]=[x[0]\ \ x[1]\ \ \ldots\ x[N\text{-}1]]^T$

- ***Graphical*** **representation of a discrete-time signal with real-valued samples is as shown below:**



In some applications, a discrete-time sequence {$x[n]$} may be generated by periodically sampling a continuous-time signal $x_a(t)$ at uniform intervals of time.



- **Here, *n-th* sample is given by**

$$x[n] = x_a(t)\,\big|_{t=nT} = x_a(nT),\quad n = \ldots, -2, -1, 0, 1, \ldots$$

$\{x[n]\}$ is a real sequence, if the n-th sample $x[n]$ is real for all values of n. Otherwise, $\{x[n]\}$ is a complex sequence

- **A complex sequence $\{x[n]\}$ can be written as $\{x[n]\}=\{x_{re}[n]\}+j\{x_{im}[n]\}$ where x_{re} and x_{im} are the real and imaginary parts of $x[n]$**
- **The complex conjugate sequence of $\{x[n]\}$ is given by $\{x^*[n]\}=\{x_{re}[n]\} - j\{x_{im}[n]\}$**

- **Example** - $\{x[n]\}=\{\cos 0.25n\}$ is a real sequence $\{y[n]\}=\{e^{j0.3n}\}$ is a complex sequence

- **We can write**

$$\{y[n]\}=\{\cos 0.3n + j\sin 0.3n\}$$
$$= \{y_{\mathrm{re}}[n]\}+ j\{y_{\mathrm{im}}[n]\}$$

where $\{y_{\mathrm{re}}[n]\}=\{\cos 0.3n\}$

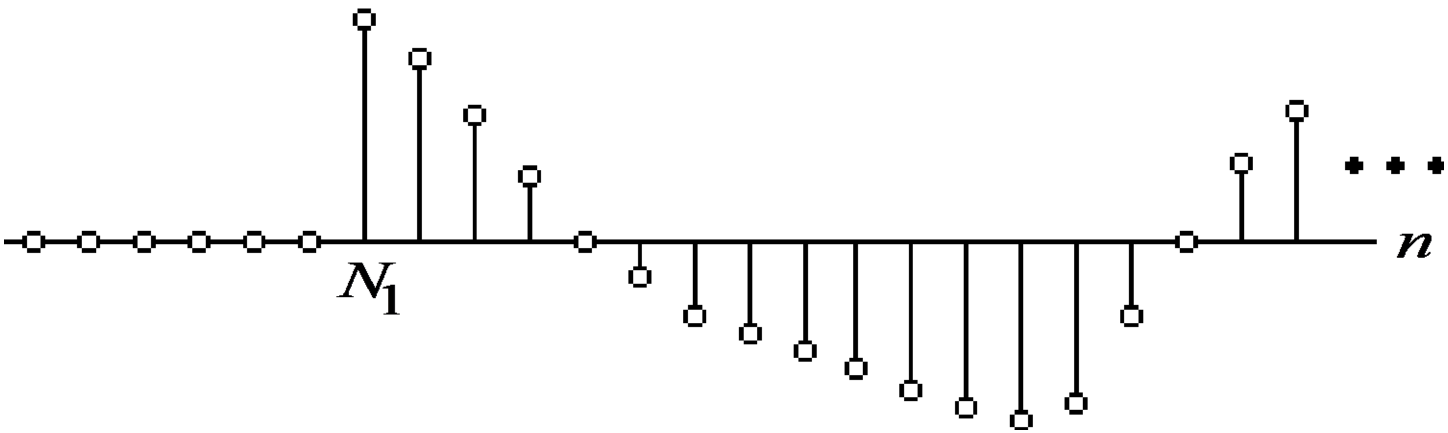
$\{y_{\mathrm{im}}[n]\}=\{\sin 0.3n\}$

- **A discrete-time signal may be a finite-length or an infinite-length sequence**
- **Finite-length (also called finite-duration or finite-extent) sequence is defined only for a finite time interval:** $N_1 \le n \le N_2$

  where $-\infty < N_1$ **and** $N_2 < \infty$ **with** $N_1 \le N_2$
- **Length or duration of the above finite-length sequence is** $N = N_2 - N_1 + 1$

Zero-padding**—append with zero-valued samples.**

- **A right-sided sequence $x[n]$ has zero-valued samples for $n < N_1$**



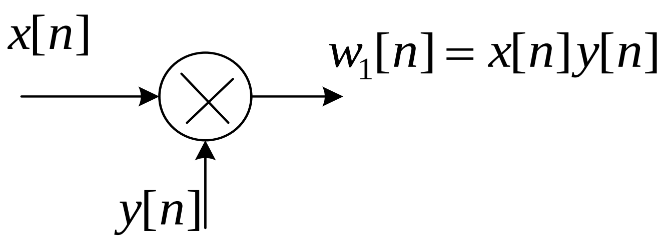
A right-sided sequence

• **If $N_1 \geq 0$, a right-sided sequence is called a causal sequence**

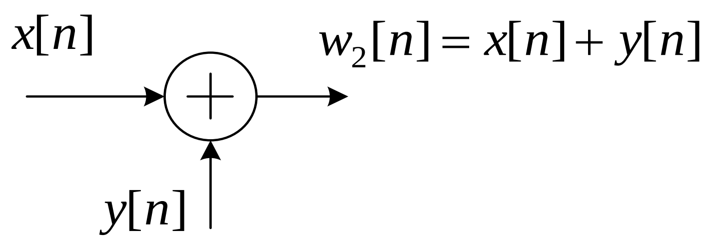
2.1.2 The Basic Operations On Sequences

1. Elementary Operations

Modulator (product or windowing)



modulator

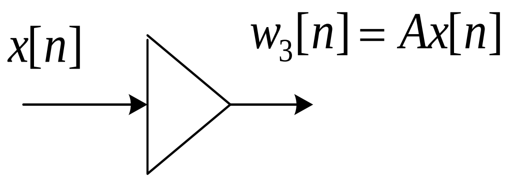


adder

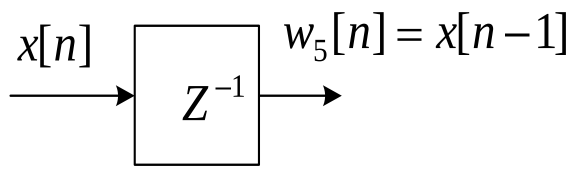
Adder (sum)

Multiplier

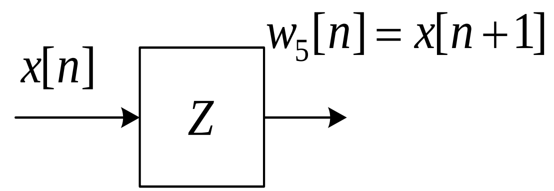
Time-shifting



multipler



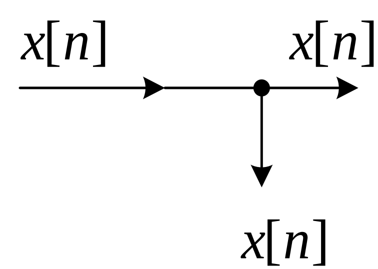
unit delay



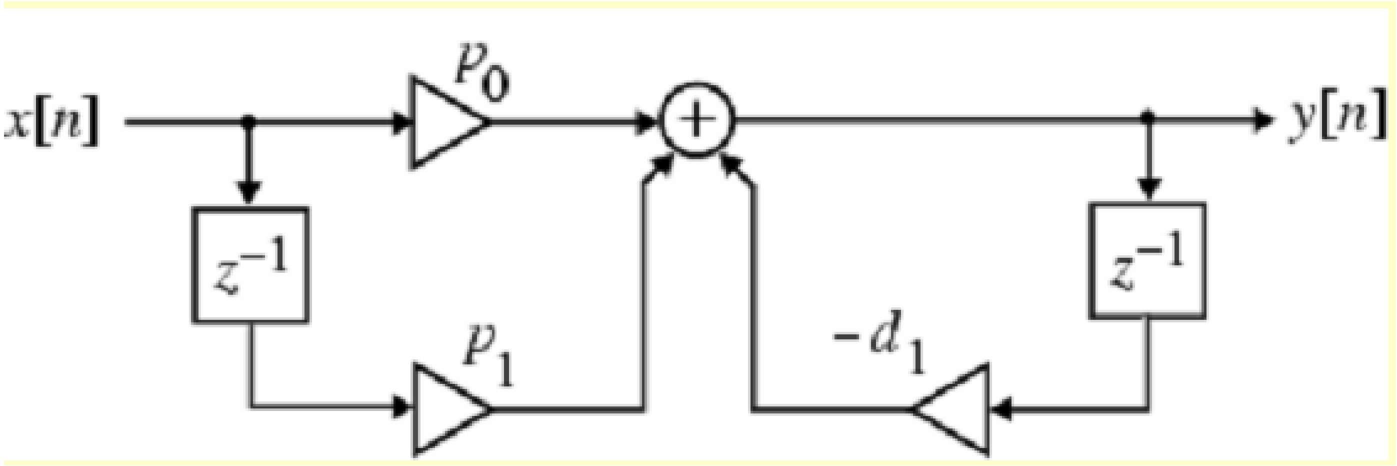
unit advance

Time-reversal (folding operation) $\boldsymbol{x[-n]}$

Pick-off node



Combination of Elementary Operations



2. Scaling in time-domain (Sampling rate alteration)

Up-sampling

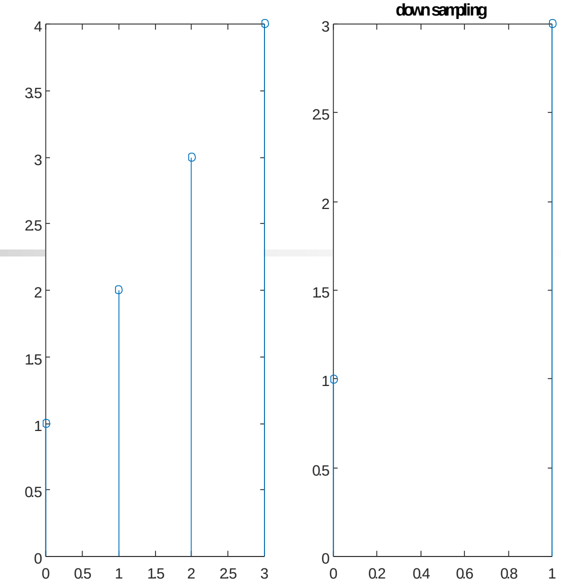
$$x_u[n] = \begin{Bmatrix} x[n/L], n = 0, \pm 1L, \pm 2L, \ldots \\ 0, otherwise \end{Bmatrix}$$

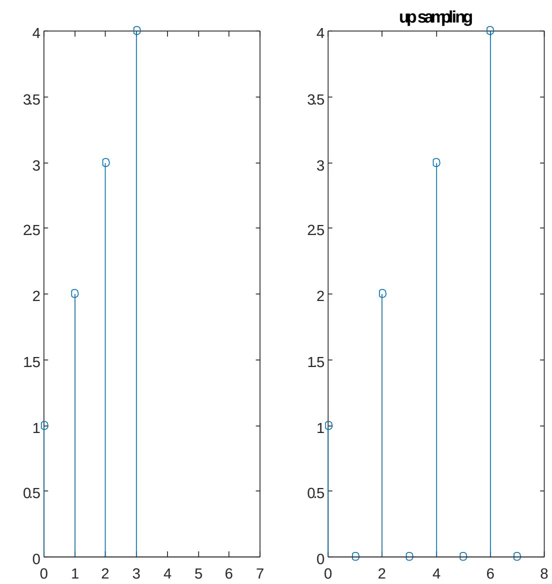
Down-sampling

$$x_d[n] = x[nM]$$

MATLAB Codes

```
close all;
x = [1 2 3 4];
y = downsample(x,2);
subplot(1,2,1);stem([1:4]-1,x);
subplot(1,2,2);xlim([0,3]);stem(
[1:2]-1,y);
title('down sampling');

x = [1 2 3 4];
y = upsample(x,2);
figure
subplot(1,2,1);stem([1:4]-
1,x);xlim([0,7]);
subplot(1,2,2);stem([1:8]-1,y);
title('up sampling');
```





2.1.3 Classification of Sequences

1. Classification based on Symmetry

Conjugate-symmetry sequence

$x^*[n]=x[-n]$ **real** $x[n]$**--even**

Conjugate-antisymmetry sequence

$x^*[n]=-x[-n]$ **real** $x[n]$**--odd**

If $\{x[n]\}$ is a real sequence, what is $\{x[n]\}$ called?

$x[n]$ is complex sequence,

Conjugate-symmetry part

$$x_{cs}[n] = \frac{1}{2}\{x[n] + x^{*}[-n]\}$$

Conjugate-antisymmetry part

$$x_{as}[n] = \frac{1}{2}\{x[n] - x^{*}[-n]\}$$

Example 2.5 A length-7 sequence(n:[-3,3]):

$\{g[n]\}=\{0,1+j4,-2+j3,\underset{\uparrow}{4-j2},-5-j6,-j2,3\}$

$$\boxed{\{g[n]\}}=\{0,\ 1+j4,\ -2+j3,\ 4-j2,\ -5-j6,\ -j2,\ 3\}$$

$\uparrow$

$$\boxed{\{g_{cs}[n]\}}=\{1.5,\ 0.5+j3,\ -3.5+j4.5,\ 4,\ -3.5-j4.5,\ 0.5-j3,\ 1.5\}$$

$\uparrow$

$$\boxed{\{g_{ca}[n]\}}=\{-1.5,\ 0.5+j,\ 1.5-j1.5,\ -j2,\ -1.5-j1.5,\ -0.5+j,\ 1.5\}$$

$\uparrow$

$$\boxed{\{g[n]\}}=\{0,\ 1+j4,\ -2+j3,\ 4-j2,\ -5-j6,\ -j2,\ 3\}$$

$$\{g^{*}[n]\}=\{0,\ 1-j4,\ -2-j3,\ 4+j2,\ -5+j6,\ j2,\ 3\}$$

$$\{g^{*}[-n]\}=\{3,\ j2,\ -5+j6,\ 4+j2,\ -2-j3,\ 1-j4,\ 0\}$$

$$\boxed{\{g_{\mathrm{cs}}[n]\}}=\{1.5,\ 0.5+j3,\ -3.5+j4.5,\ 4,\ -3.5-j4.5,\ 0.5-j3,\ 1.5\}$$

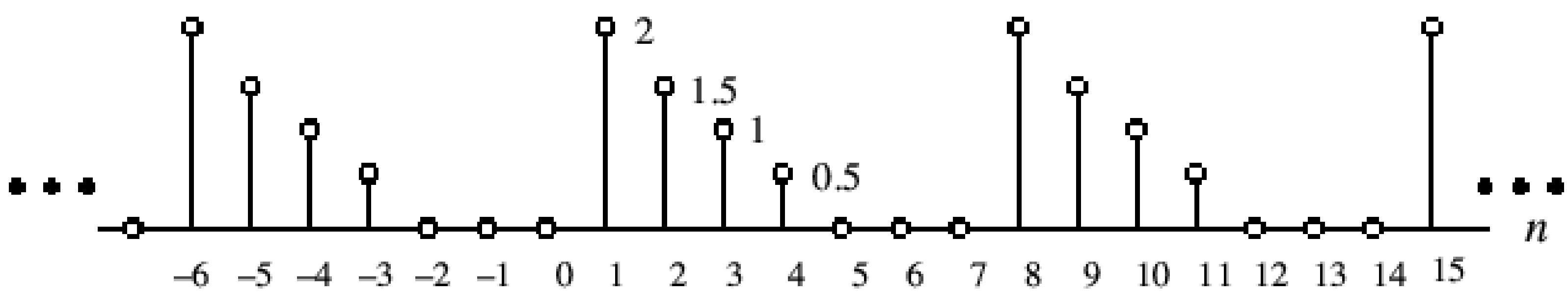
$$\boxed{\{g_{\mathrm{ca}}[n]\}}=\{-1.5,\ 0.5+j,\ 1.5-j1.5,\ -j2,\ -1.5-j1.5,\ -0.5+j,\ 1.5\}$$

2.Classification of Sequences based on periodicity

- **A sequence satisfying the periodicity condition is called an periodic sequence**

$$\tilde{x}[n] = \tilde{x}[n + kN]$$

- ***Example***



Adding two periodic sequences with different periods

$$\tilde{y}[n] = \tilde{x}_a[n] + \tilde{x}_b[n] \qquad N_a \qquad N_b$$

period $$N = \frac{N_a N_b}{GCD(N_a, N_b)}$$

GCD()---greatest common divisor

example $$f[n] = \cos\frac{\pi}{8}n + \frac{1}{2}\sin\frac{\pi}{6}n$$

Its period is 48 which is the GCD of 16 and 12.

3.Classification of Sequences Energy and Power Signals

- **Total energy of a sequence $x[n]$ is defined by**

$$\mathcal{E}_{\mathrm{x}} = \sum_{n=-\infty}^{\infty} \left| x[n] \right|^2$$

- **An infinite length sequence with finite sample values may or may not have finite energy**
- **A finite length sequence with finite sample values has finite energy**

- **The average power of an aperiodic sequence is defined by**

$$P_x = \lim_{K \to \infty} \frac{1}{2K+1} \sum_{n=-K}^{K} |x[n]|^2$$

•**Define the energy of a sequence *x*[*n*] over a finite interval -*K*≤ *n* ≤ *K* as**

$$\mathcal{E}_{x,K} = \sum_{n=-K}^{K} |x[n]|^2$$

- **An infinite energy signal with finite average power is called a power signal**

  Example **- A periodic sequence which has a finite average power but infinite energy**

- **A finite energy signal with zero average power is called an energy signal**

  Example **- A finite-length sequence which has finite energy but zero average power**

- ***Example*** **- Consider the causal sequence defined by**

$$x[n] = \begin{cases} 3(-1)^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

- **Note: *x*[*n*] has infinite energy**
- **Its average power is given by**

$$P_x = \lim_{K\to\infty} \frac{1}{2K+1}\left(9\sum_{n=0}^{K} 1\right) = \lim_{K\to\infty} \frac{9(K+1)}{2K+1} = 4.5$$

4. Classification of Sequences bounded, absolutely summable and squaresummable

- **A sequence {$x[n]$} is said to be bounded if**

$$|x[n]| \leq B_x < \infty$$

- ***Example*** **- The sequence {$x[n]$}={cos(0.3πn)} is a bounded sequence as**

$$|x[n]| = |\cos 0.3\pi n| \leq 1$$

- **A sequence $x[n]$ is said to be absolutely summable if**

$$\sum_{n=-\infty}^{\infty} \left| x[n] \right| < \infty$$

- ***Example*** **- The sequence**

$$y[n] = \begin{cases} 0.3^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

is an absolutely summable sequence as

$$\sum_{n=0}^{\infty} \left| 0.3^n \right| = \frac{1}{1-0.3} = 1.42857 < \infty$$

- **A sequence $x[n]$ is said to be square-summable if**

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty$$

- ***Example* - The sequence**

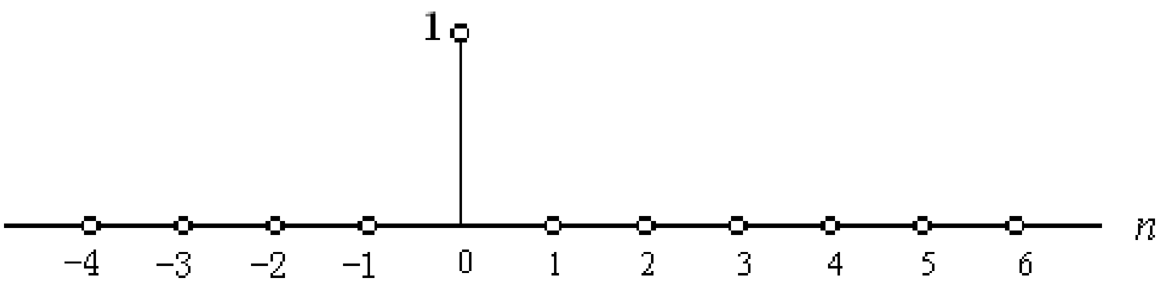
$$h[n] = \frac{\sin 0.4n}{\pi n}$$

is square-summable but not absolutely summable

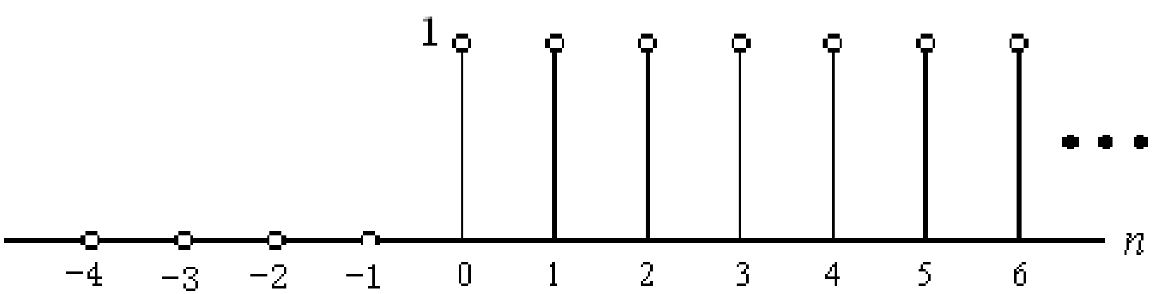
2.2 Typical Sequences and Representation

2.2.1 Basic Sequences

- **Unit sample sequence** $\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$



- **Unit step sequence** $\mu[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$

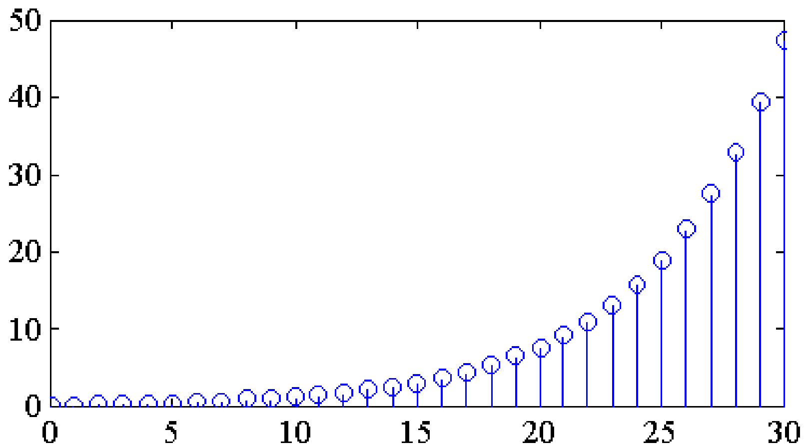


Exponential **sequence**

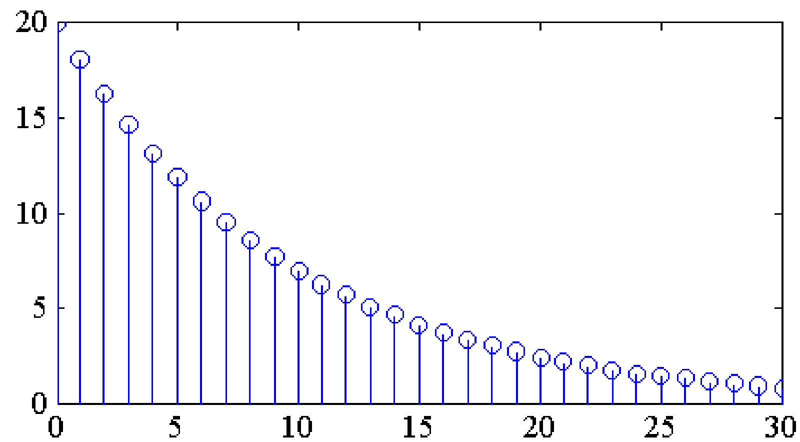
$$x[n] = A\alpha^n, -\infty < n < \infty$$

when *A* and α are real numbers.

α=1.2



α=0.9



when A and α are complex numbers.

By expressing

$$\alpha = e^{(\sigma_0 + j\omega_0)},\ A = |A|e^{j\phi}$$

we can rewrite

$$x[n] = |A|e^{\sigma_0 n}e^{j(\omega_0 n + \phi)}$$

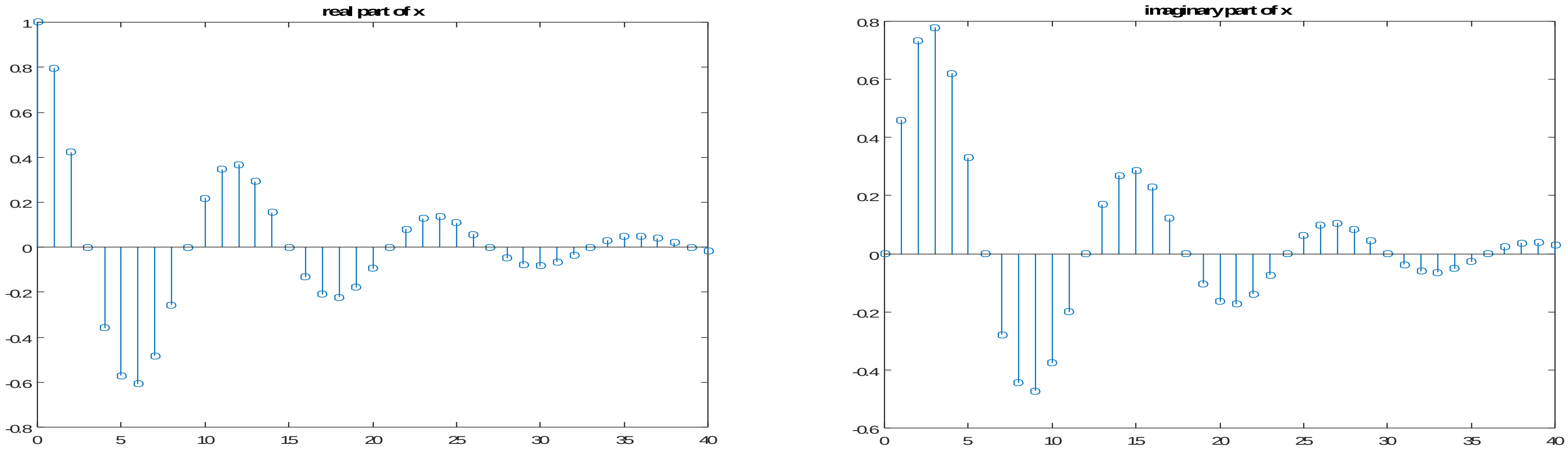
And $$x[n] = x_{re}[n] + jx_{im}[n]$$

Then, $$x_{re}[n] = |A|e^{\sigma_0 n}\cos(\omega_0 n + \phi)$$

$$x_{im}[n] = |A|e^{\sigma_0 n}\sin(\omega_0 n + \phi)$$

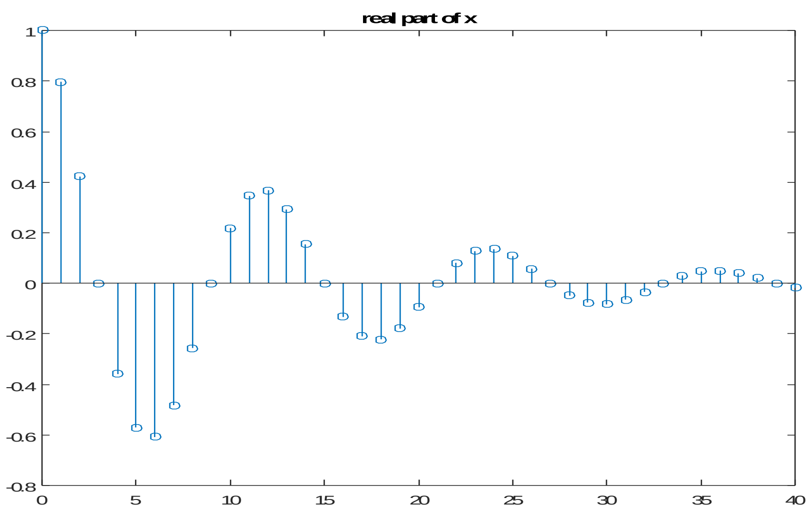
Example

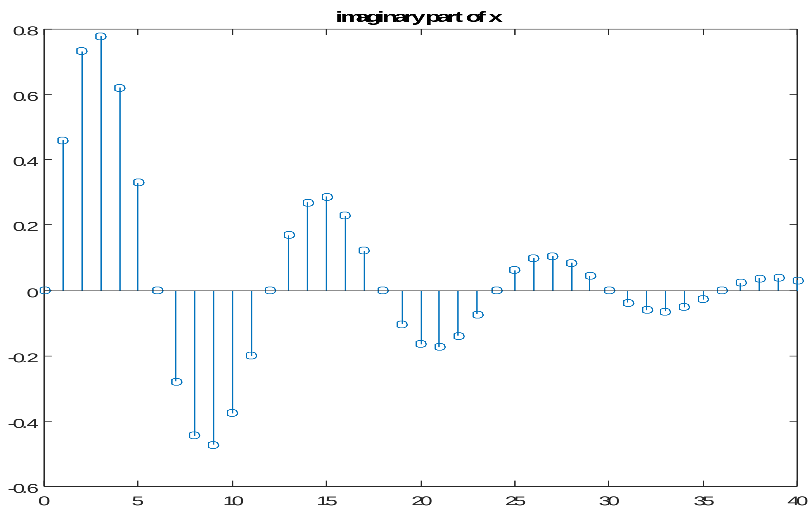
$$x[n] = \exp\left(-\frac{1}{12} + j\frac{\pi}{6}\right)n$$



Example

```
close all;
n = 0:40;
x = exp((-1/12+j*pi/6)*n);
stem(n, real(x)); title('real part of x');
figure
stem(n, imag(x)); title('imaginary part of x');
```





The Complex Exponential Sequence

$$e^{j\omega_0 n} = \cos\omega_0 n + j\sin\omega_0 n$$

Real sinusoidal sequence

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

where A is the amplitude, ω_0 is the angular frequency, and φ is the phase of $x[n]$

It is periodic if $\omega_0 = \frac{2\pi}{N} r$, where N and r must be integer number

($\omega_0 / 2\pi$ must be arational number) .

Why?

How to determine the *fundamental period*?

To verify the above fact, consider

$$x_1[n] = \cos(\omega_0 n + \varphi)$$

$$x_2[n] = \cos[\omega_0(n+N) + \varphi]$$

Now

$x_2[n] = \cos(\omega_0(n+N) + \varphi)$

$= \cos(\omega_0 n+\varphi)\cos\omega_0 N - \sin(\omega_0 n+\varphi)\sin\omega_0 N$

which will be equal to $\cos(\omega_0 n+\varphi)=x_1[n]$ only if $\sin\omega_0 N= 0$ and $\cos\omega_0 N = 1$.

These two conditions are met if and only if $\omega_0 N= 2\pi m$ or $2\pi/\omega_0 = N/m$.

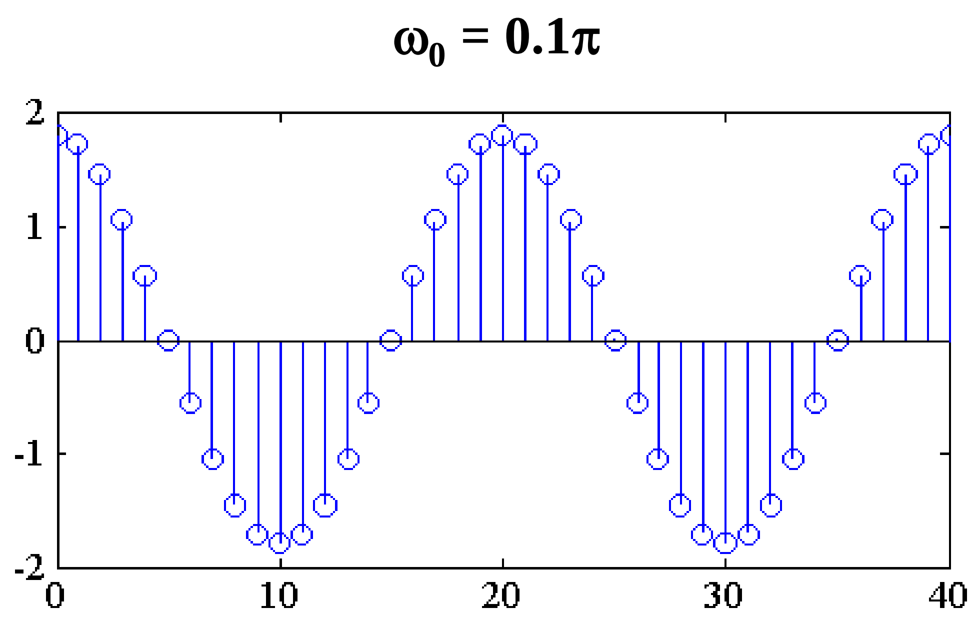
where N and m are positive integers.

Smallest value of N satisfying $\omega_0 N=2\pi m$ is the *fundamental period* of the sequence.

- If $2\pi/\omega_0$ is not an rational number, the sequence is aperiodic
- *Example* - $x[n]=\sin(3n+\varphi)$ is an aperiodic sequence.

$y[n]=\cos(0.1\pi n)$ is a periodic sequence.

Hence $N= 2\pi r/\omega 0$ $= 20$ for $r = 1$



Because $e^{j\omega_0 n} = e^{j(\omega_0 + 2k\pi)n}$ **(*k* is integer),the discrete sinusoidal signal has the highest angular frequency** $\pm\pi$ **. For example,**

$$g_1[n] = \cos(0.6\pi n)$$

$$g_2[n] = \cos(1.4\pi n)$$

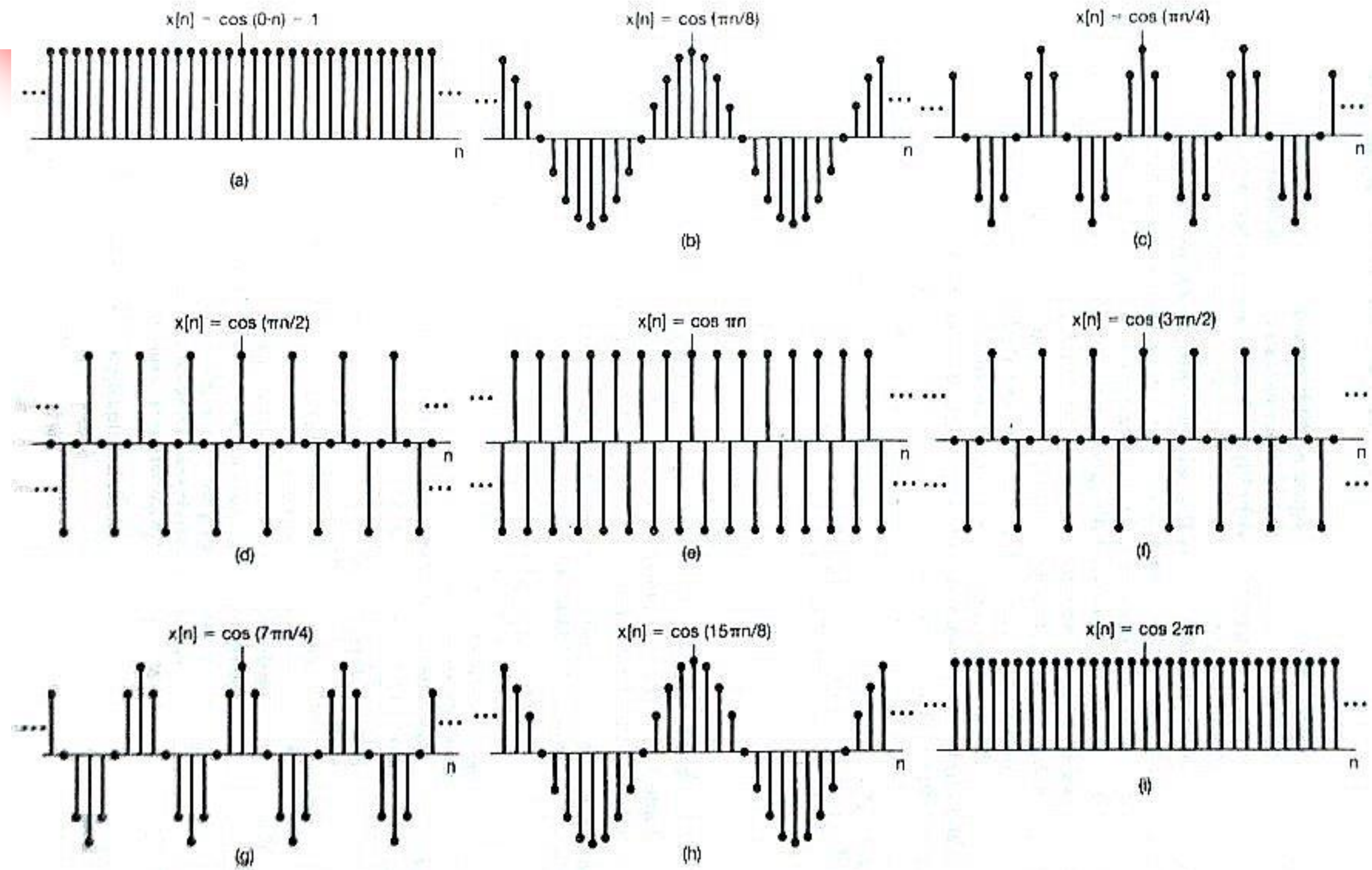
$$g_3[n] = \cos(2.6\pi n)$$

Observation

$$g_2[n] = \cos(1.4\pi n) = \cos\big((2\pi - 0.6\pi)n\big) = \cos(0.6\pi n)$$

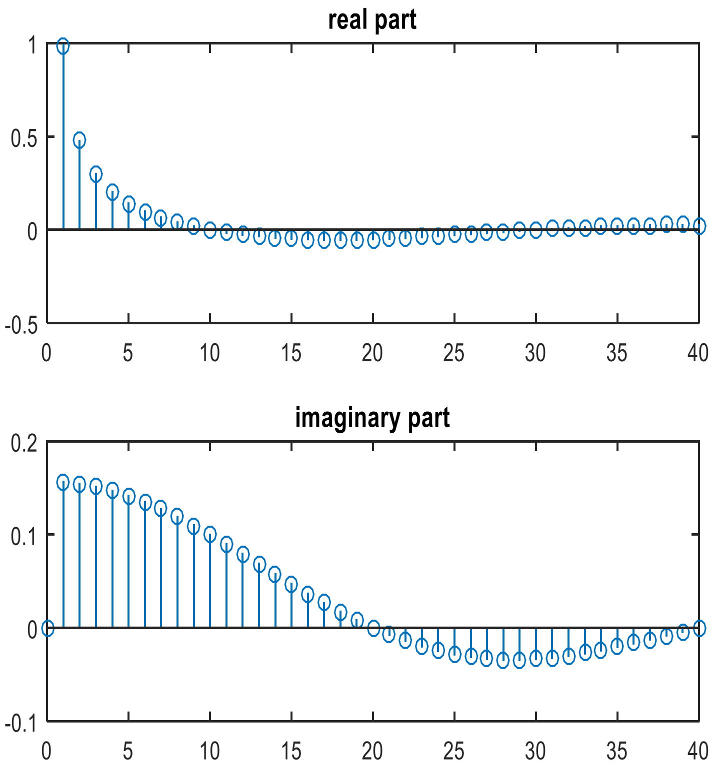
$$g_3[n] = \cos(2.6\pi n) = \cos\big((2\pi + 0.6\pi)n\big) = \cos(0.6\pi n)$$

Periodicity Properties



2.2.2 Sequence Generation Using MATLAB

Read and Exercise!

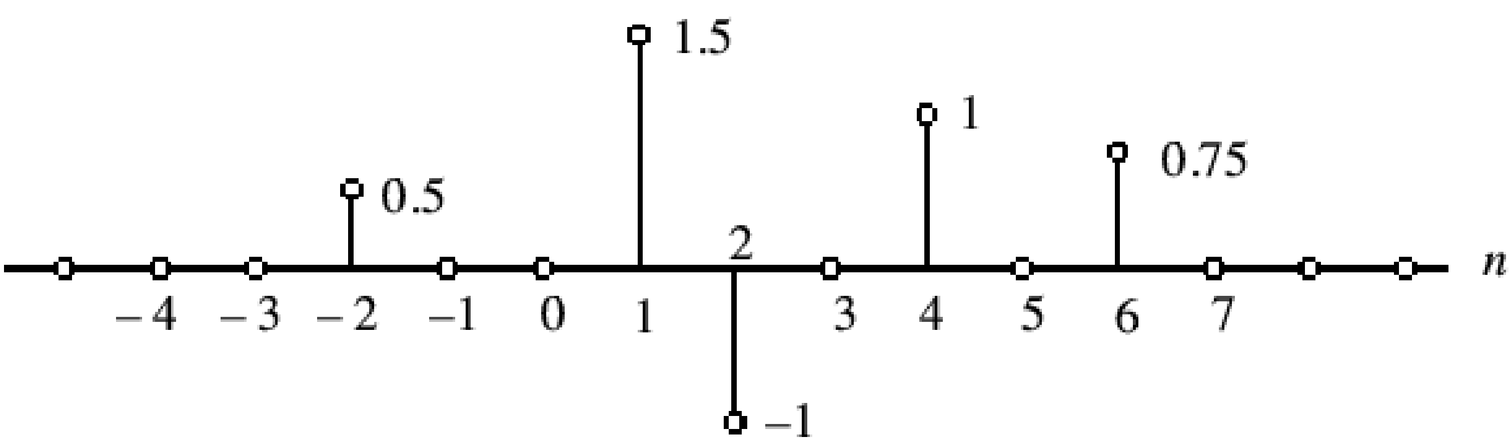



```
N = 40;
n = 0:N;
K = 1./n;
c = pi/20*i;
x = K.*exp(c*n);
subplot(2,1,1);
stem(n,real(x));title('real part');
subplot(2,1,2);
stem(n,imag(x));title('imaginary part');
```

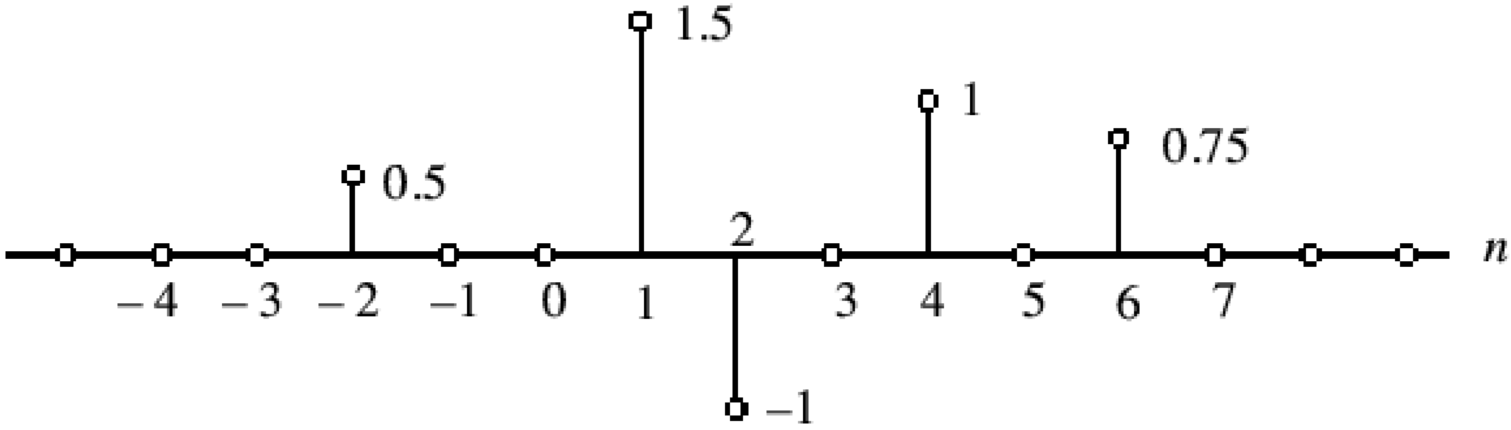
2.2.3 Representation of an Arbitrary Sequence

- **An arbitrary sequence can be represented in the time-domain as a weighted sum of some basic sequence and its delayed (advanced) versions**

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



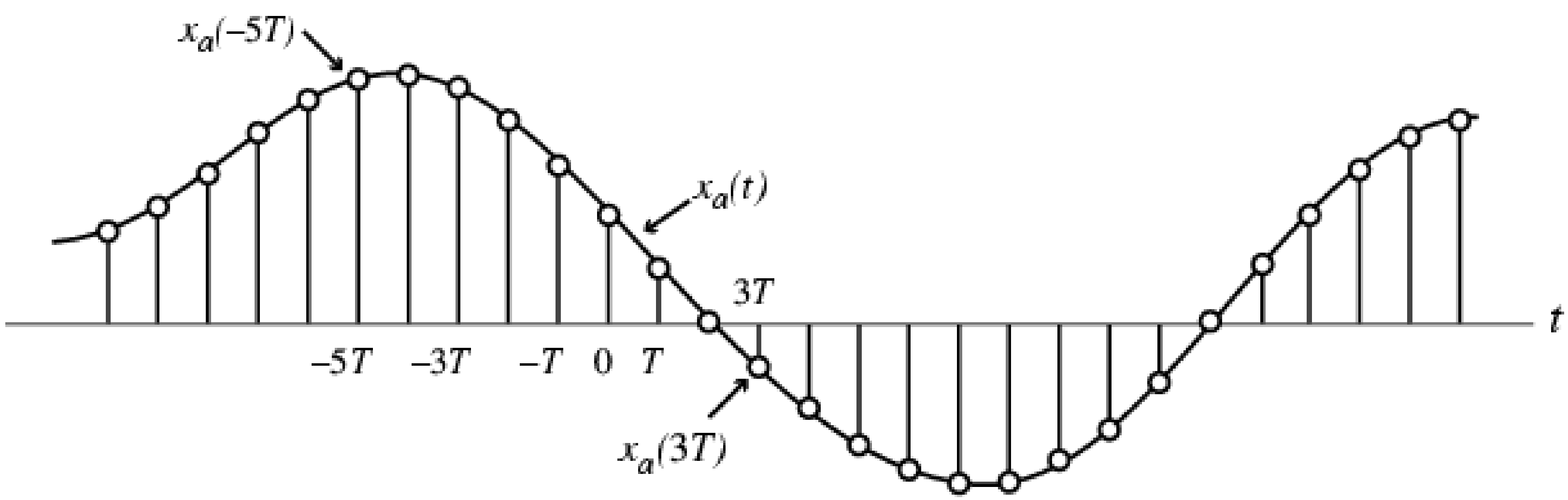
Example



$$x[n] = 0.5\delta[n+2] + 1.5\delta[n-1] - \delta[n-2] + \delta[n-4] + 0.75\delta[n-6]$$

2.3 The Sampling Process

- Often, a discrete-time sequence $x[n]$ is developed by uniformly sampling a continuous-time signal $x_a(t)$ as indicated below



- The relation between the two signals is

$$x[n] = x_a(t)|_{t=nT} = x_a(nT), \quad n = \ldots, -2, -1, 0, 1, 2, \ldots$$

- **Time variable t of $x_a(t)$ is related to the time variable n of $x[n]$ only at discrete-time instants t_n given by**

$$t_n = nT = n/F_T = 2\pi n/\Omega_T$$

with $F_T = 1/T$ denoting the sampling frequency and $\Omega_T = 2\pi F_T$ denoting the sampling angular frequency.

- **Consider the continuous-time signal**

$$x(t) = A\cos(2\pi f_o t + \phi) = A\cos(\Omega_o t + \phi)$$

The corresponding discrete-time signal is

$$x[n] = A\cos(\Omega_o nT + \phi) = A\cos(\frac{2\pi\Omega_o}{\Omega_T} n + \phi)$$

$$= A\cos(\omega_o n + \phi)$$

where

$$\omega_o = 2\pi\,\Omega_o / \,\Omega_T = \,\Omega_o T$$

is the normalized digital angular frequency of *x*[*n*]

- **If the unit of sampling period T is in seconds**
  - **The unit of normalized digital angular frequency ω_0 is radians/sample.**
  - **The unit of analog frequency f_0 is hertz (Hz) (cycles/second)**
  - **The unit of analog angular frequency Ω_0 is radians/second**
  - **The unit of sampling frequency f_T is samples/second**
  - **So, the unit of normalized digital angular frequency ω_0 is radians/sample**

- **The three continuous-time signals**

$$g_1(t) = \cos(6\pi t)$$

$$g_2(t) = \cos(14\pi t)$$

$$g_3(t) = \cos(26\pi t)$$

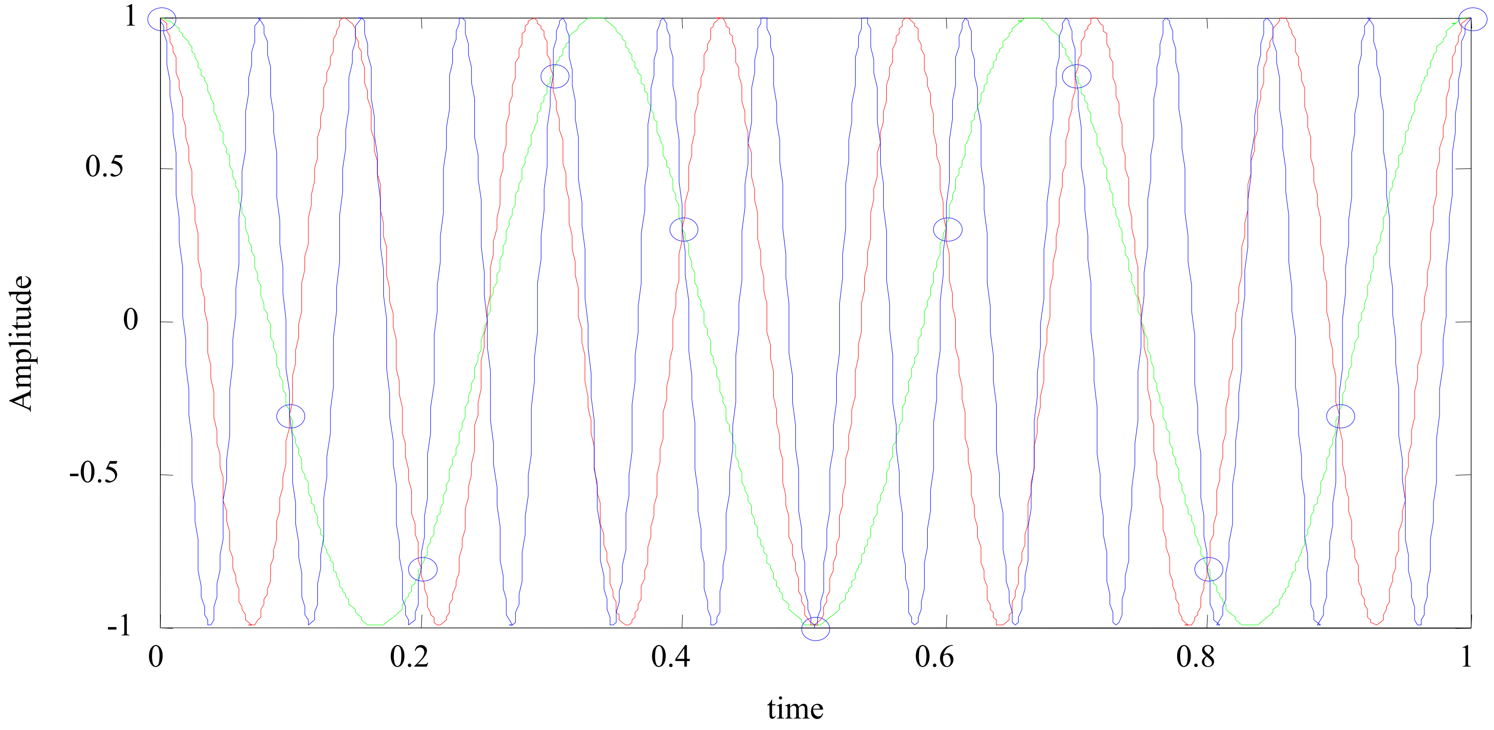
of frequencies 3 Hz, 7 Hz, and 13 Hz, are sampled at a sampling rate of 10 Hz, i.e. with *T* = 0.1 sec. generating the three sequences

$$g_1[n] = \cos(0.6\pi n)$$

$$g_2[n] = \cos(1.4\pi n)$$

$$g_3[n] = \cos(2.6\pi n)$$

- **Plots of these sequences (shown with circles) and their parent time functions are shown below:**



- **Note that each sequence has exactly the same sample value for any given *n*.** **Why?**

- **This fact can also be verified by observing that**

$$g_2[n] = \cos(1.4\pi n) = \cos\big((2\pi - 0.6\pi)n\big) = \cos(0.6\pi n)$$

$$g_3[n] = \cos(2.6\pi n) = \cos\big((2\pi + 0.6\pi)n\big) = \cos(0.6\pi n)$$

- **As a result, all three sequences are identical and it is difficult to associate a unique continuous-time function with each of these sequences.**

- **The above phenomenon of a continuous-time signal of higher frequency acquiring the identity of a sinusoidal sequence of lower frequency after sampling is called aliasing.**

Recall $\omega_0 = 2\pi\Omega_0/\Omega_T$

Thus if $\Omega_T > 2\Omega_0$, then ω_0 of the discrete-time signal obtained by sampling the parent continuous-time sinusoidal signal will be in the range $-\pi < \omega < \pi$

Conclusion: No aliasing

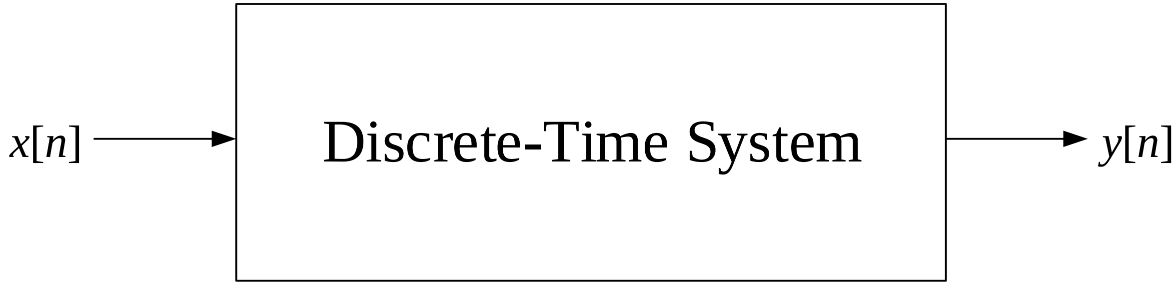
- **On the other hand, if $\Omega_T < 2\Omega_0$, the normalized digital angular frequency will foldover into a lower digital frequency**

  $\omega_0 = (2\pi\Omega_0/\Omega_T)_{2\pi}$ in the range $-\pi<\omega<\pi$ because of aliasing. (e.g., $\cos(1.4\pi n)$)

- **Hence, to prevent aliasing, the sampling frequency Ω_T should be greater than 2 times the frequency Ω_0 of the sinusoidal signal being sampled. (Sampling Theorem)**

2.4 Discrete-Time Systems

- **Linear System**
- **Shift-Invariant System**
- **Causal System**
- **Stable System**
- **Passive and Lossless Systems**



§ 2.4.1 Linear Discrete-Time Systems

- **Definition - If $y_1[n]$ is the output due to an input $x_1[n]$, and $y_2[n]$ is the output due to an input $x_2[n]$ then for an input**

$$x[n] = ax_1[n] + bx_2[n]$$

the output is given by

$$y[n] = ay_1[n] + by_2[n]$$

- **Above property must hold for any arbitrary constants a and b and for all possible inputs $x_1[n]$ and $x_2[n]$.**
- **Hence, the above system is linear.**

§ 2.2.2 Shift-Invariant System

- **For a shift-invariant system, if $y_1[n]$ is the response to an input $x_1[n]$, then the response to an input**

$$x[n] = x_1[n - n_0]$$

is simply

$$y[n] = y_1[n - n_0]$$

where n_0 is any positive or negative integer.

- **The above relation must hold for any arbitrary input and its corresponding output**
- **The above property is called time-invariance property, or shift-invariant property**

Linear Time-Invariant system

- Linear Time-Invariant (LTI) System —
  A system satisfying both the linearity and the time-invariance property
- LTI systems are mathematically easy to analyze and characterize, and consequently, easy to design
- Highly useful signal processing algorithms have been developed utilizing this class of systems over the last several decades

Generally, The math model of a LTI system is a linear difference equation with constant coefficients, as following form:

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

Note: Up-sampling

$$y[n] = \begin{cases} x[n/L], n = 0, \pm 1L, \pm 2L, \ldots \\ 0, otherwise \end{cases}$$

is a time-varying system, so does Down-sampling.

Passive and Lossless Systems

- A discrete-time system is defined to be **passive** if, for every finite-energy input $x[n]$, the output $y[n]$ has, at most, the same energy, i.e.

$$\sum_{n=-\infty}^{\infty} |y[n]|^2 \le \sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty$$

- For a **lossless system**, the above inequality is satisfied with an equal sign for **every input**

2.5 Analysis of LTI Systems In Time-Domain (2.5-2.8)

2.5.1 Time-Domain Characterization of LTI Discrete-Time System

- **Input-Output Relationship -**
  A consequence of the linear, time-invariance property is that an LTI discrete-time system is completely characterized by its impulse response
- **Knowing the impulse response one can compute the output of the system for any arbitrary input**

Impulse and Step Responses

- **The response of a discrete-time system to a unit sample sequence $\{\delta[n]\}$ is called the unit impulse response or simply, the impulse response, and is denoted by $\{h[n]\}$**
- **The response of a discrete-time system to a unit step sequence $\{\mu[n]\}$ is called the unit step response or simply, the step response, and is denoted by $\{s[n]\}$**

Some Examples

- ***Example*** - The impulse response of the system

$$y[n]=a_1x[n]+a_2x[n-1]+a_3x[n-2]+a_4x[n-3]$$

is obtained by setting $x[n] = \delta[n]$ resulting in

$$h[n] = a_1\delta[n]+a_2\delta[n-1] +a_3\delta[n-2] +a_4\delta[n-3]$$

- The impulse response is thus a finite-length sequence of length 4 given by

$$\{h[n]\} = \{\underset{\uparrow}{a_1}, a_2, a_3, a_4\}$$

- ***Example*** **- The impulse response of the discrete-time accumulator**

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

is obtained by setting $x[n] = \delta[n]$ **resulting in**

$$h[n] = \sum_{\ell=-\infty}^{n} \delta[\ell] = \mu[n]$$

- ***Example*** **- The impulse response $\{h[n]\}$ of the factor-of-2 interpolator**

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

is obtained by setting $x_u[n] = \delta[n]$ and is given by

$$h[n] = \delta[n] + \frac{1}{2}(\delta[n-1] + \delta[n+1])$$

- **The impulse response is thus a finite-length sequence of length 3:**

$$\{h[n]\} = \{0.5, \quad \underset{\uparrow}{1} \quad 0.5\}$$

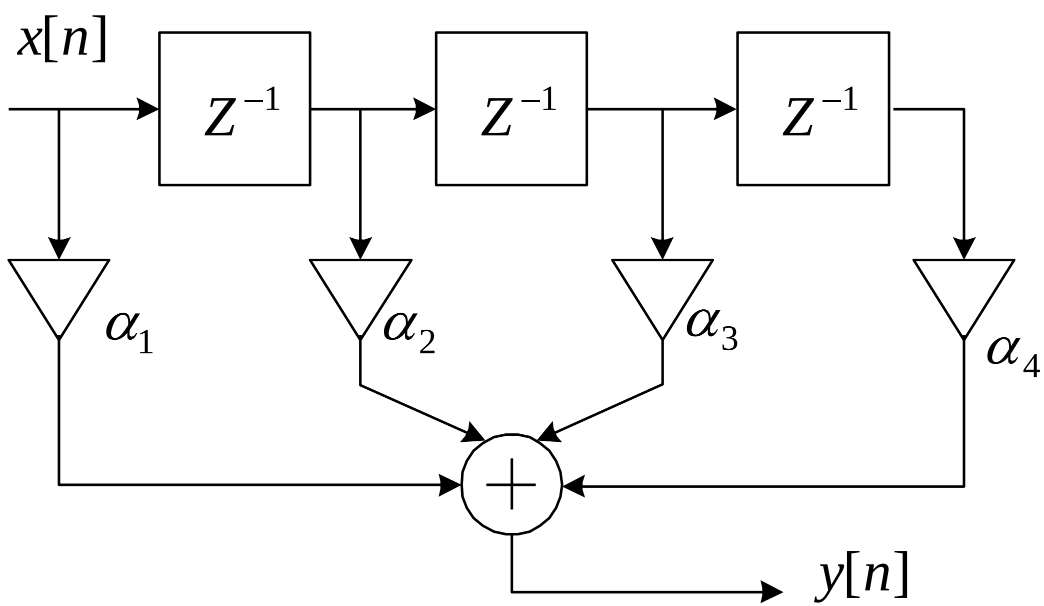
Classification of LTI Systems based on $h[n]$

1. Input-Output Relationship;
   Impulse Response;
2. FIR And IIR System
   (recursive and non-recursive)

Example

non-recursive

(FIR)

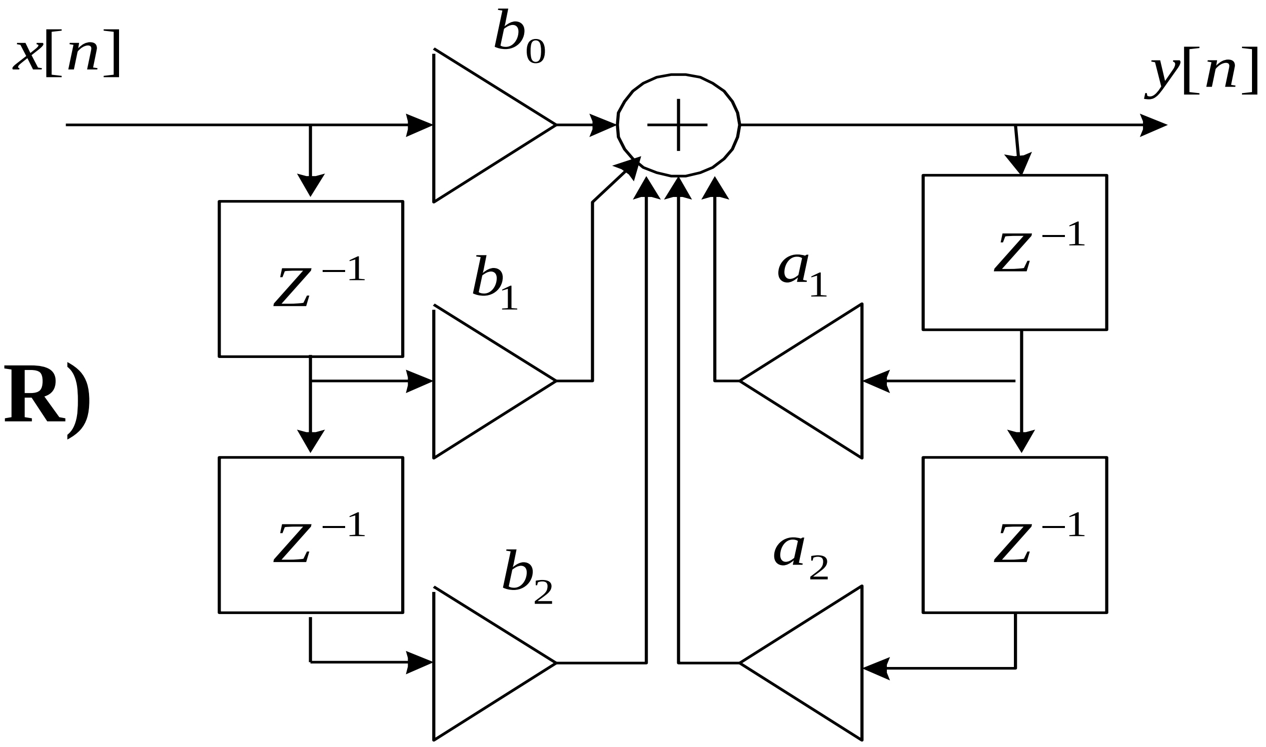


$$y[n] = \alpha_1 x[n] + \alpha_2 x[n-1] + \alpha_3 x[n-2] + \alpha_4 x[n-3]$$

Example

recursive

(IIR)



$$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] + a_1 y[n-1] + a_2 y[n-2]$$

- ***Example*** **- The familiar numerical integration formulas that are used to numerically solve integrals of the form**

$$y(t) = \int_0^t x(\tau)d\tau$$

can be shown to be characterized by linear constant coefficient difference equations, and hence, are examples of IIR systems

- **If we divide the interval of integration into n equal parts of length T, then the previous integral can be rewritten as**

$$y(nT) = y\left[(n-1)T\right] + \int_{(n-1)T}^{nT} x(\tau)\,d\tau$$

where we have set $t = nT$ and used the notation

$$y(nT) = \int_{0}^{nT} x(\tau)\,d\tau$$

- **Using the trapezoidal method we can write**

$$\int_{(n-1)T}^{nT} x(\tau)d\tau = \frac{T}{2}\{x((n-1)T) + x(nT)\}$$

Hence, a numerical representation of the definite integral is given by

$$y(nT) = y((n-1)T) + \frac{T}{2}\{x((n-1)T) + x(nT)\}$$

- **Let $y[n] = y(nT)$ and $x[n] = x(nT)$ with $y(0)=0$**
- **Then**

$$y(nT) = y((n-1)T) + \frac{T}{2}\{x((n-1)T) + x(nT)\}$$

reduces to

$$y[n] = y[n-1] + \frac{T}{2}\{x[n] + x[n-1]\}$$

which is recognized as the difference equation representation of a first-order IIR discrete-time system.

Use the basic operations to represent the system!

2.5.2 Convolution Sum

- **Let $h[n]$ denote the impulse response of a LTI discrete-time system**
- **Compute its output $y[n]$ for the input:**

$$x[n] = 0.5\delta[n+2] + 1.5\delta[n-1] - \delta[n-2] + 0.75\delta[n-5]$$

- **As the system is linear, we can compute its outputs for each member of the input separately and add the individual outputs to determine $y[n]$**

- **Likewise, as the system is linear**

Input **output**

$$0.5\delta[n+2] \rightarrow 0.5h[n+2]$$

$$1.5\delta[n-1] \rightarrow 1.5h[n-1]$$

$$-\delta[n-2] \rightarrow -h[n-2]$$

$$0.75\delta[n-5] \rightarrow 0.75h[n-5]$$

Hence because of the linearity property we get

$$y[n] = 0.5h[n+2] + 1.5h[n-1] - h[n-2] + 0.75h[n-5]$$

- **Now, any arbitrary input sequence $x[n]$ can be expressed as a linear combination of delayed and advanced unit sample sequences in the form**

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

- **The response of the LTI system to an input $x[k]\delta[n-k]$ will be $x[k]h[n-k]$**

- **Hence, the response $y[n]$ to an input**

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \qquad \textbf{(2.73a)}$$

will be

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

which can be alternately written as

$$y[n] = \sum_{k=-\infty}^{\infty} x[n-k]h[k] \qquad \textbf{(2.73b)}$$

Convolution Sum

- **The summation**

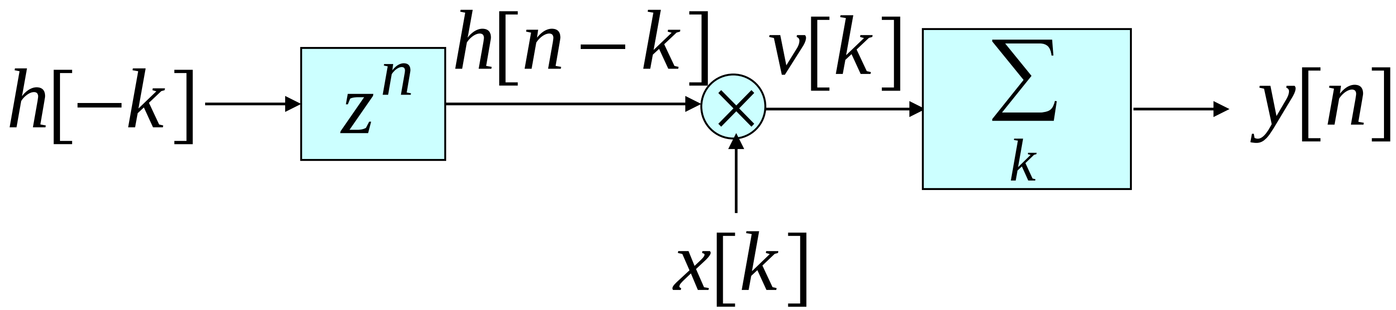
$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = \sum_{k=-\infty}^{\infty} x[n-k]h[n]$$

is called the convolution sum of the sequences $x[n]$ and $h[n]$ and represented compactly as

$$y[n] = x[n] \circledast h[n]$$

Convolution Sum

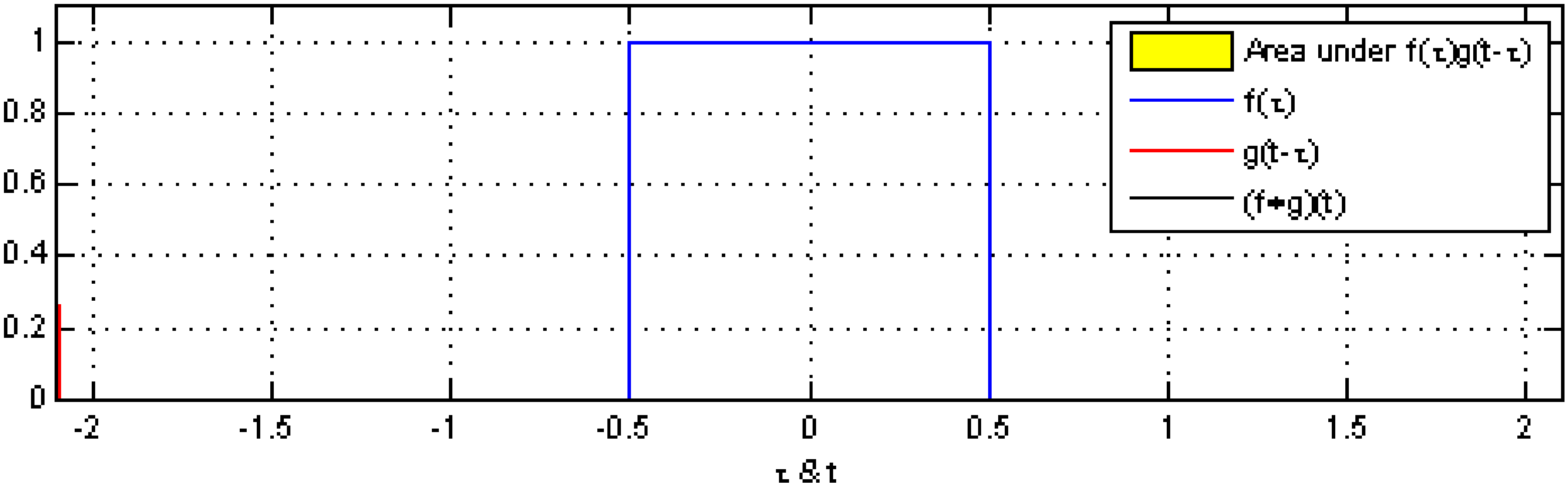
- **Schematic Representation**



- **The computation of an output sample using the convolution sum is simply a sum of products.**

- **Involves fairly simple operations such as additions, multiplications, and delays.**

Convolution Sum



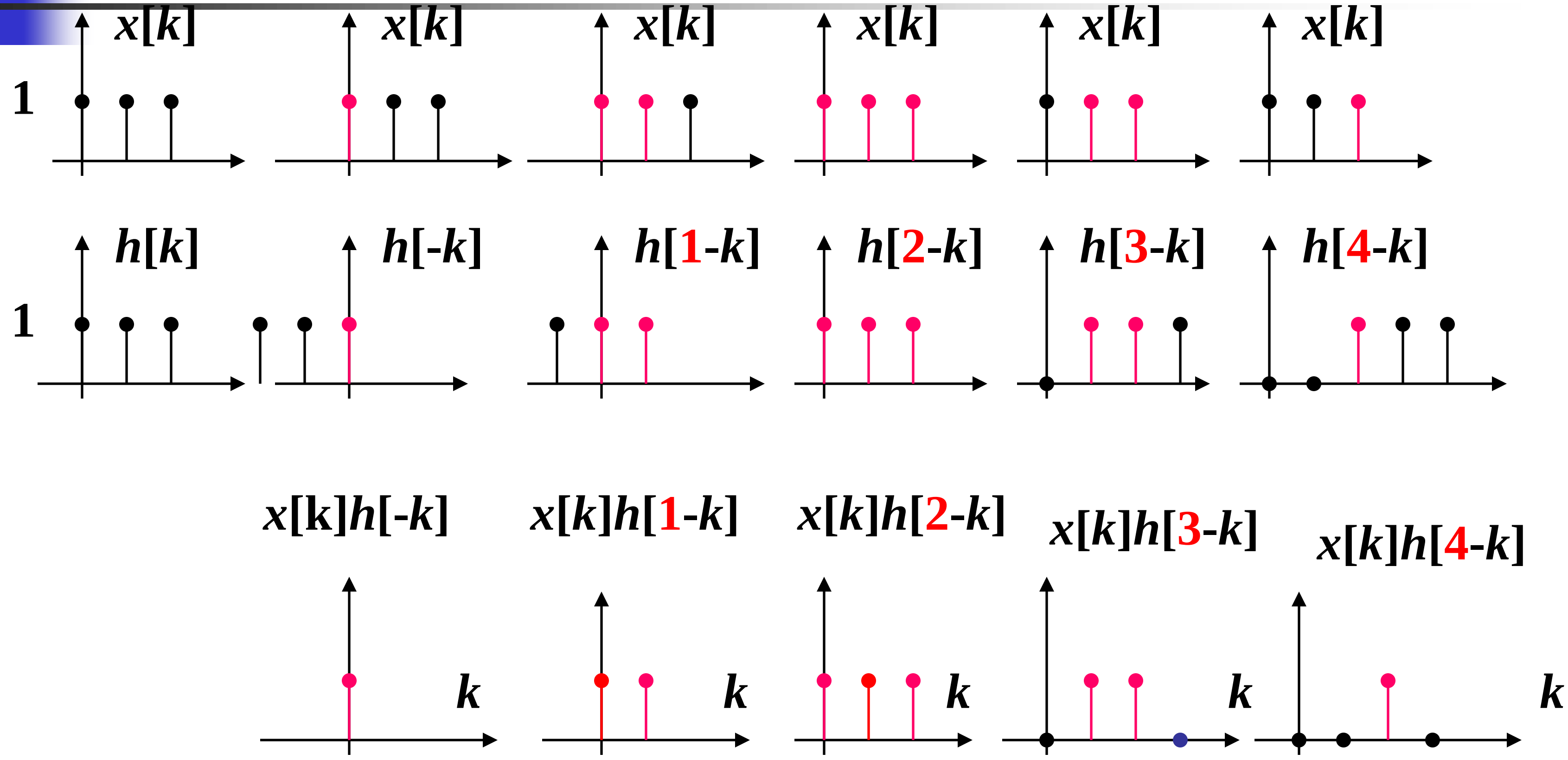
Method I

- ***Example*: Develop the sequence $y[n]$ generated by the convolution of the sequences $x[n]$ and $h[n]$:**

$$\boldsymbol{x[n] = h[n] = \delta[n]+\delta[n\text{-}1]+\delta[n\text{-}2]}$$

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

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



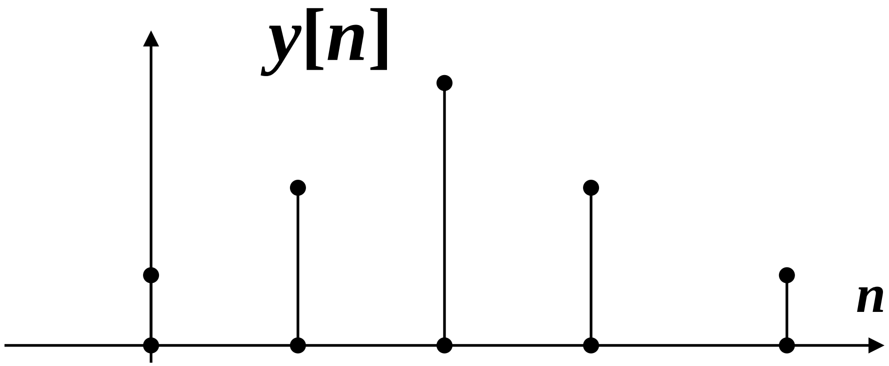
$\sum x[k]h[-k]$ $=1$

$\sum x[k]h[1-k]$ $=2$

$\sum x[k]h[2-k]$ $=3$

$\sum x[k]h[3-k]$ $=2$

$\sum x[k]h[4-k]$ $=1$



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

- **In general, if the lengths of the two sequences being convolved are M and N, then the sequence generated by the convolution is of length $M+N-1$**

Method II: In tabular form,

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

$x[n]$:	3	-2	4		
$h[n]$:	4	2	-1		
	4×3	$4\times(-2)$	4×4		
		2×3	$2\times(-2)$	2×4	
			-1×3	$-1\times(-2)$	-1×4

$\{y[n]\} = \{12 \quad \underset{\uparrow}{-2} \quad 9 \quad 10 \quad -4\}$

n	0	1	2	3	4
$x[n]$	x_1	x_2	x_3		
$h[n]$	h_1	h_2	h_3		
	h_1x_1	h_1x_2	h_1x_3		
		h_2x_1	h_2x_2	h_2x_3	
			h_3x_1	h_3x_2	h_3x_3
	h_1x_1	$h_1x_2 + h_2x_1$	$h_1x_3 + h_2x_2 + h_3x_1$	$h_2x_3 + h_3x_2$	h_3x_3

Method III:

Example- **$x[n]=(1/2)^{n-2}\mu[n-2]$**

$h[n]=\mu[n+2]$

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

$$=\sum_{k=-\infty}^{\infty}(\frac{1}{2})^{k-2}\mu[k-2]\mu[(n-k)+2]$$

$$=\sum_{k=2}^{n+2}(\frac{1}{2})^{k-2}=\frac{(\frac{1}{2})^{2-2}-(\frac{1}{2})^{n+2-2+1}}{1-\frac{1}{2}}$$

$$=[2-(\frac{1}{2})^{n}]\mu[n]$$

Method IV:

Convolution Sum Using MATLAB

- **The M-file `conv` implements the convolution sum of two finite-length sequences**
- If `a=[-2  0  1  -1  3]`

  `b=[1  2  0  -1]`

  then `conv(a,b)` yields

  `[-2  -4  1  3  1  5  1  -3]`

Example The Impulse response of system:

$$y[n]+0.7y[n-1]-0.45y[n-2]-0.6y[n-3]=$$
$$0.8x[n]-0.44x[n-1]+0.36x[n-2]+0.02x[n-3]$$

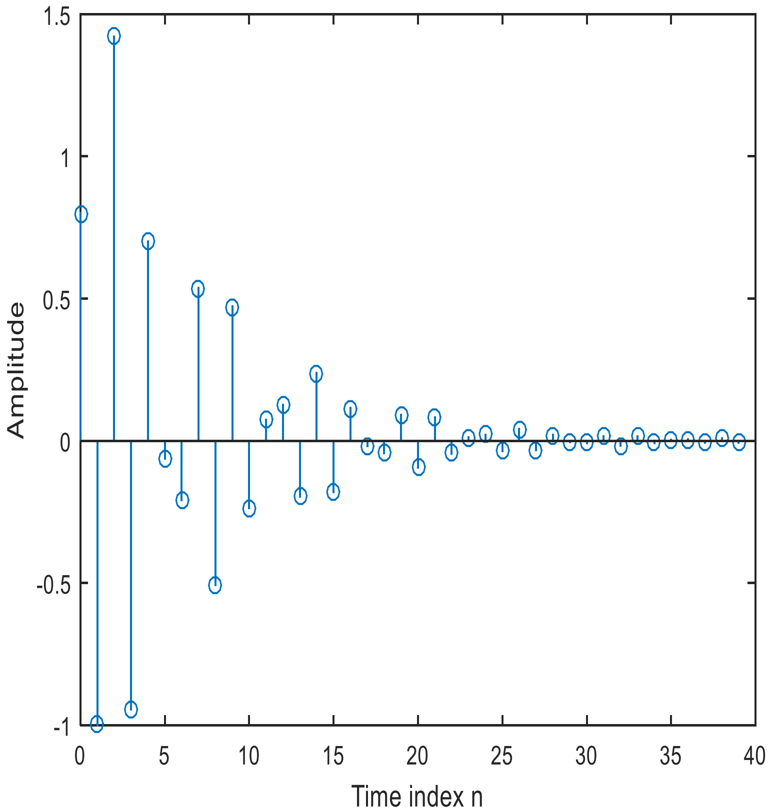
Is wanted. Compute the output when input is

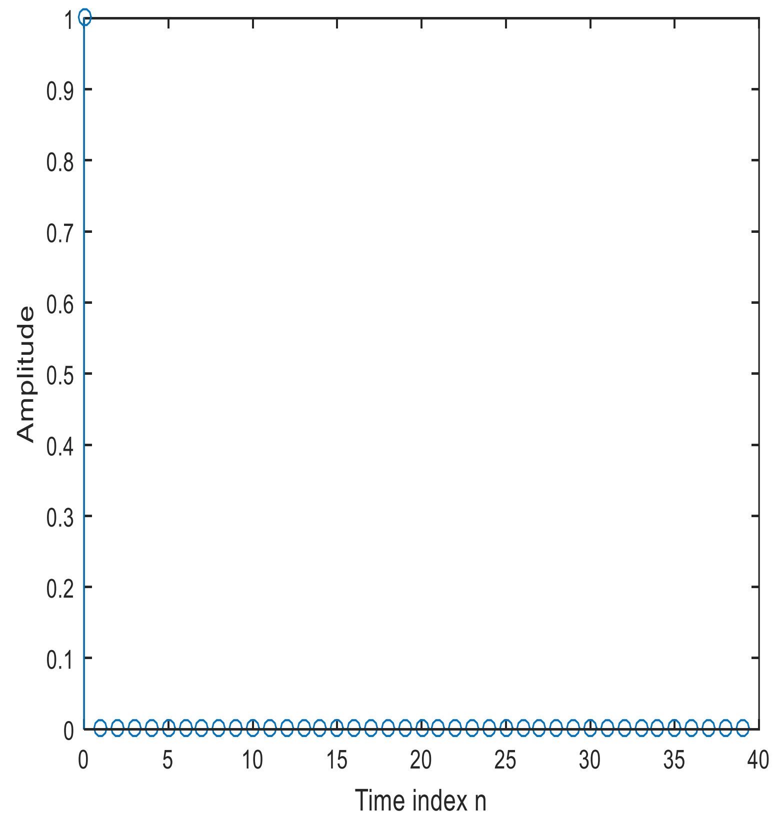
$$\boldsymbol{x[n] = \{\underset{\uparrow}{1}\ 2\ 0.5\ 0.4\}}$$

The MATLAB program of changing input

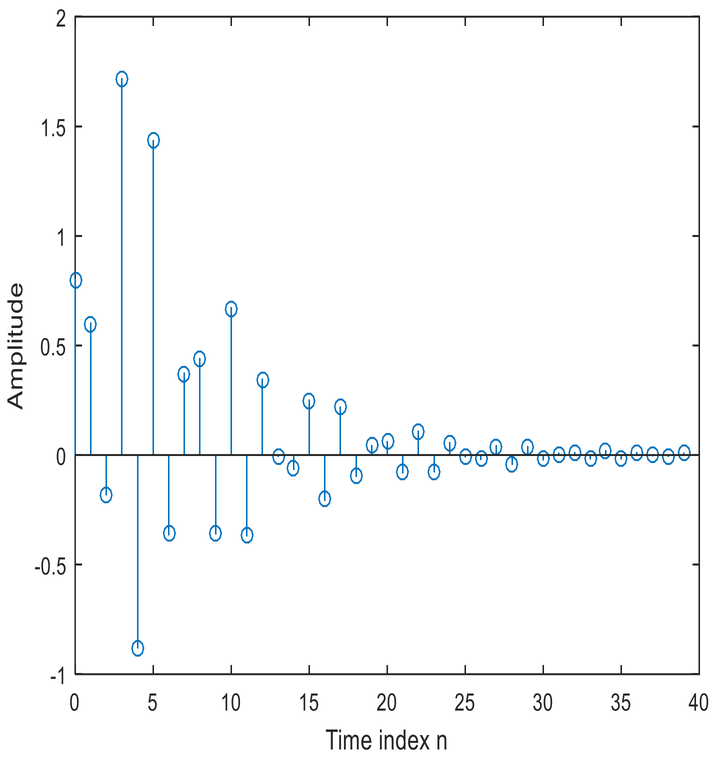
```
% Example 2.37 change input
% Illustration of Impulse Response Computation
% N = input('Desired impulse response length = ');
% p = input('Type in the vector p =');
% d = input('Type in the vector d =');

N = 40;
p = [0.8 -0.44 0.36 0.02];
d = [1 0.7 -0.45 -0.6];

x = [1 zeros(1,N-1)]; % x = [1 2 0.5 0.4 zeros(1,N-4)];

%unit impulse
y = filter(p,d,x);
figure(1);
k = 0:1:N-1;
stem(k,y)
xlabel('Time index n'); ylabel('Amplitude')
figure(2);
k = 0:1:N-1;
stem(k,x)
xlabel('Time index n'); ylabel('Amplitude')
```

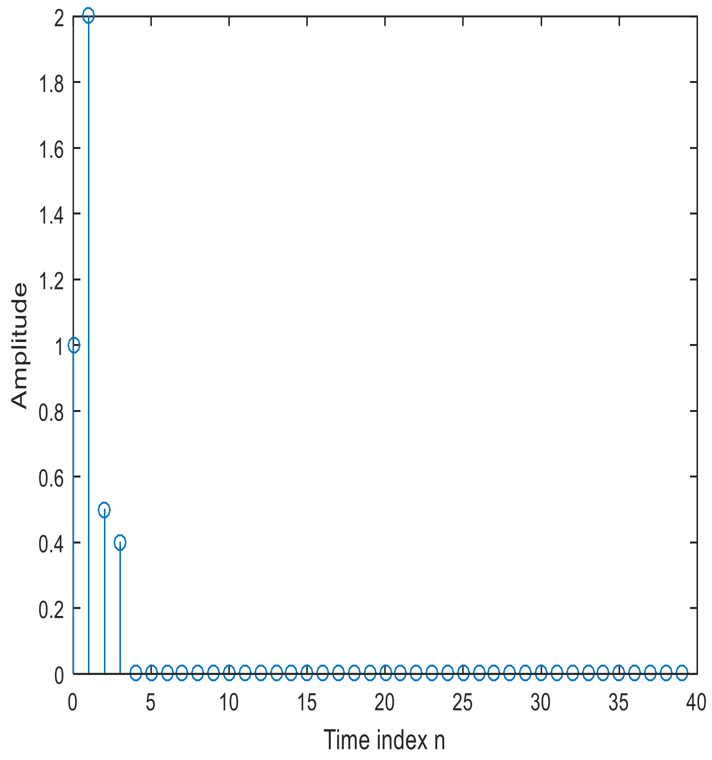
Impulse response





Input response





$y[n] = x[n]$ LTI

$\Updownarrow$

$h[n] = \delta[n]$

$\Updownarrow$

$\{\underset{\uparrow}{1}\}$

System A

$x[n] = \{\underset{\uparrow}{1}, 2\}$

$\Updownarrow$

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

Signal A

Signal A $\rightarrow$ $\boxed{\text{System A}}$ $\rightarrow$ Signal y_A

$y_A[n] = h[n] + 2h[n-1]$

$= \delta[n] + 2\delta[n-1]$

$\{\underset{\uparrow}{1}, 2\}$

$y_A[n] = x[n] \circledast h[n]$

n	0	1	2
$x[n]$	$\underset{\uparrow}{1}$	2	
$h[n]$	$\underset{\uparrow}{1}$		
	$\underset{\uparrow}{1 \times 1}$	1×2	
	$\underset{\uparrow}{1}$	2	

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

$\Updownarrow$

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

$\Updownarrow$

$\{\underset{\uparrow}{1}, 2\}$

System B

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

$\Updownarrow$

$x[n] = \{\underset{\uparrow}{1}\}$

Signal B

Signal B $\rightarrow$ $\boxed{\text{System B}}$ $\rightarrow$ Signal y_B

$y_B = \cancel{\delta[n]}\, h[n]$

$= \delta[n] + 2\delta[n-1]$

$\{\underset{\uparrow}{1}, 2\}$

$y_B = x[n] \circledast h[n]$

n	0	1	2
$x[n]$	$\underset{\uparrow}{1}$		
$h[n]$	$\underset{\uparrow}{1}$	2	
	$\underset{\uparrow}{1 \times 1}$		
		2×1	
	$\underset{\uparrow}{1}$	2	

$\{\underset{\uparrow}{1}, 2\}$

~~Signal B~~

Illustration of relationship between signal and system

2.5.3 Simple Interconnection Schemes

(Convolution Sum Properties)

- **Commutative property :**

$$x[n] \circledast h[n] = h[n] \circledast x[n]$$

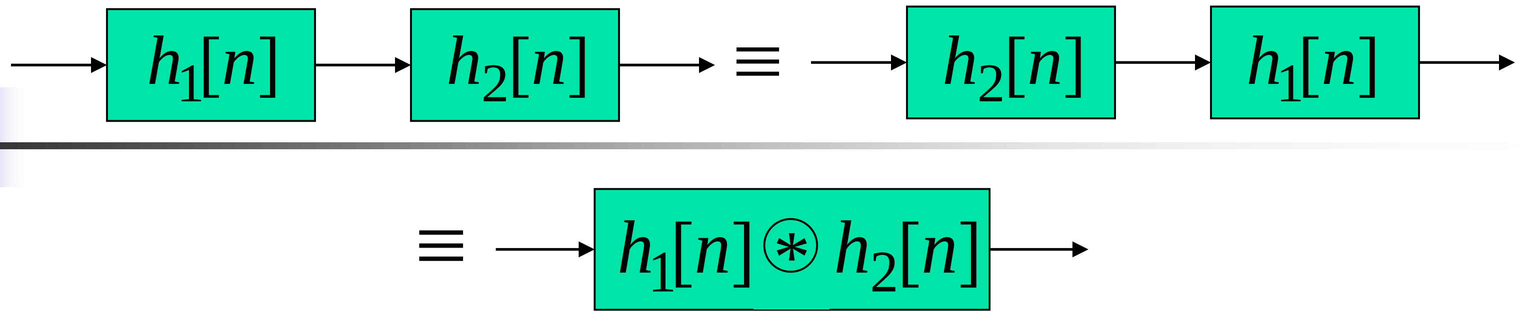
- **Associative property :**

$$(x[n] \circledast h[n]) \circledast y[n] = x[n] \circledast (h[n] \circledast y[n])$$

- **Distributive property :**

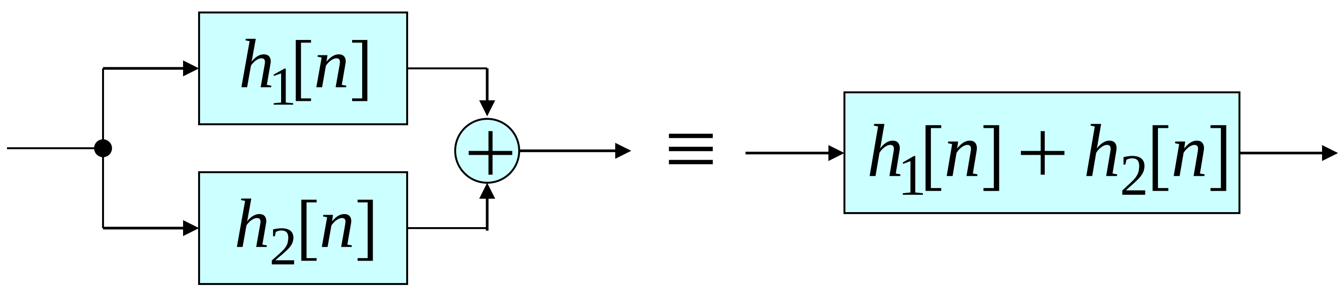
$$x[n] \circledast (h[n] + y[n]) = x[n] \circledast h[n] + x[n] \circledast y[n]$$

- **Cascade Connection**



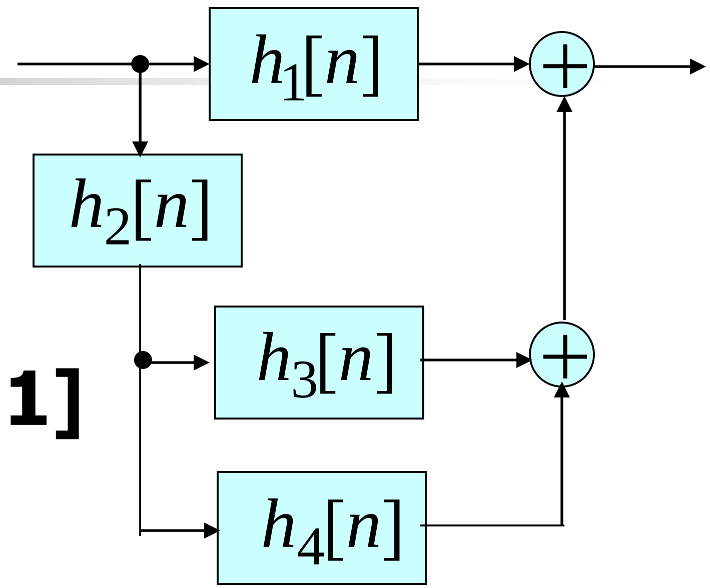
$$h[n] = h_1[n] \circledast h_2[n]$$

- **Parallel Connection**



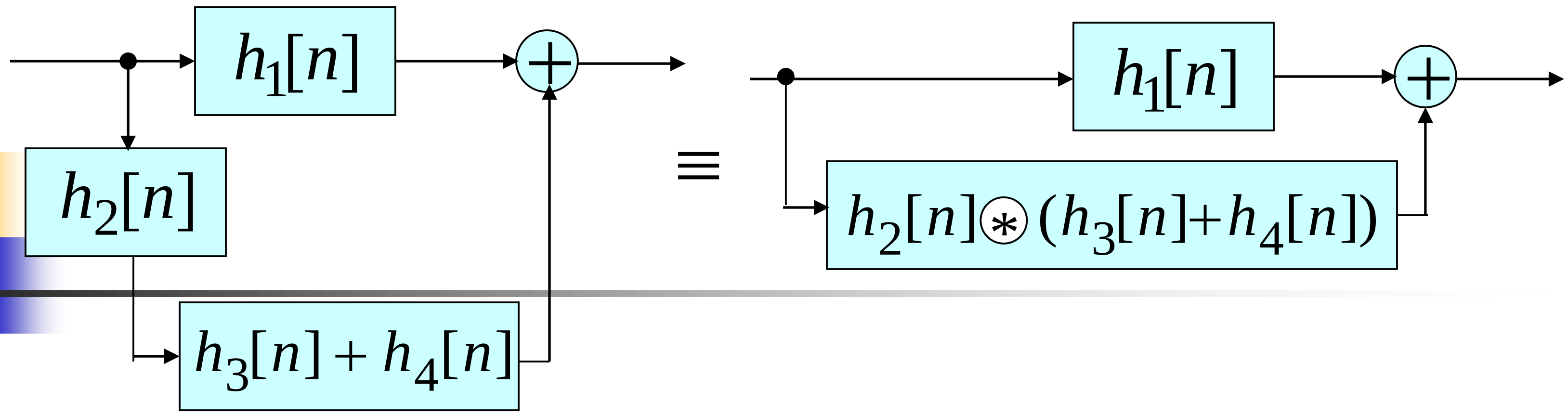
$$h[n] = h_1[n] + h_2[n]$$

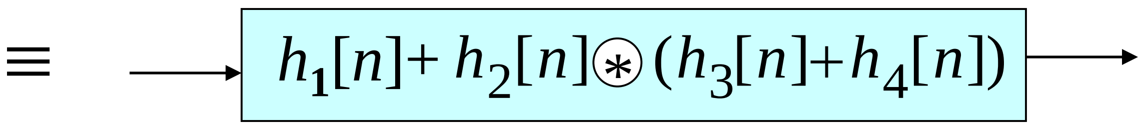
Consider the discrete-time system where



$$h_1[n] = \delta[n] + 0.5\delta[n-1]$$
$$h_2[n] = 0.5\delta[n] - 0.25\delta[n-1]$$
$$h_3[n] = 2\delta[n]$$
$$h_4[n] = -2(0.5)^n\mu[n]$$

- **Simplifying the block-diagram we obtain**





$$
\begin{aligned}
h[n] &= h_1[n]+h_2[n]\circledast(h_3[n]+h_4[n]) \\
&= h_1[n]+h_2[n]\circledast h_3[n]+h_2[n]\circledast h_4[n]
\end{aligned}
$$

Now,

$$
\begin{aligned}
h_2[n]\circledast h_3[n] &= \left(\tfrac{1}{2}\delta[n]-\tfrac{1}{4}\delta[n-1]\right)\circledast 2\delta[n] \\
&= \delta[n]-\tfrac{1}{2}\delta[n-1]
\end{aligned}
$$

$$h_2[n] * h_4[n] = (\tfrac{1}{2}\delta[n] - \tfrac{1}{4}\delta[n-1]) \circledast \left(-2(\tfrac{1}{2})^n \mu[n]\right)$$

$$= -(\tfrac{1}{2})^n \mu[n] + \tfrac{1}{2}(\tfrac{1}{2})^{n-1} \mu[n-1]$$

$$= -(\tfrac{1}{2})^n \mu[n] + (\tfrac{1}{2})^n \mu[n-1]$$

$$= -(\tfrac{1}{2})^n \delta[n] = -\delta[n]$$

Therefore,

$$h[n] = \delta[n] + \frac{1}{2}\delta[n-1] + \delta[n] - \frac{1}{2}\delta[n-1] - \delta[n] = \delta[n]$$

2.5.4 Classification of LTI Discrete-Time Systems

Based on Impulse Response Length -

- **If the impulse response $h[n]$ is of finite length, i.e.,**

$$h[n] = 0 \quad \text{for } n < N_1, N_2 < n \text{ and } N_1 < N_2$$

then it is known as a Finite Impulse Response (FIR) discrete-time system

- **The convolution sum description here is**

$$y[n] = \sum_{k=N_1}^{N_2} h[k]x[n-k]$$

- **The output $y[n]$ of an FIR LTI discrete-time system can be computed directly from the convolution sum as it is a finite sum of products**
- **Examples of FIR LTI discrete-time systems are the *moving-average system* and the *linear interpolators***

- **If the impulse response is of infinite length, then it is known as an Infinite Impulse Response (IIR) discrete-time system**
- **The class of IIR systems we are concerned with in this course are characterized by *linear constant coefficient difference equations***

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

Stability in terms of impulse response

A stable system:

$h[n]$ is absolutely summable.

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

Causality in terms of impulse response

A causal system:

$$h[n] = 0, n < 0$$

Example consider a LTI system shown as below:

$$h[n] = \alpha^n \mu[n]$$

For this system

$$S = \sum_{n=-\infty}^{\infty} \left|\alpha^n\right| \mu[n] = \sum_{n=0}^{\infty} \left|\alpha\right|^n = \frac{1}{1-\left|\alpha\right|} \qquad if\ \left|\alpha\right| < 1$$

So ,if $|\alpha|$<1, above system is stable. Otherwise, the system is not stable.

2.6 Finite-Dimension LTI Discrete-Time Systems

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

order = max(*N*, *M*)

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

$y_{zi}[n]$ $y_{zs}[n]$

Zero-Input response **Zero-state response**

Zero-input response and zero-state response

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

Here, the *zero-input response* $y_{zi}[n]$ is obtained by solving the equation by setting the input $x[n]$ = 0, the *zero-state response* $y_{zs}[n]$ is obtained by solving the equation by applying the specified input with all initial conditions set to zero.

Zero-Input response: $y_{zi}[n]$

homogeneous equation

$$\sum_{k=0}^{N} d_k y[n-k] = 0$$

homogeneous solution **(同类解，相似解)**

$$y_h[n] = d_0\lambda^n + d_1\lambda_1^{n-1} + d_2\lambda_2^{n-2} + \ldots + d_N\lambda_N^{n-N}$$

Eigen root: $\lambda_1, \ \lambda_2, \ \ldots, \ \lambda_N$

Zero-Input response: ***Coefficient?***

$$y_{zi}[n] = d_0\lambda^n + d_1\lambda_1^{n-1} + d_2\lambda_2^{n-2} + \ldots + d_N\lambda_N^{n-N}, \quad n \geq 0$$

Example Known the difference equation as bellow:

$$y[n]+y[n-1]-6y[n-2]=x[n]$$

For a step input $x[n] = 8\mu[n]$ and with initial conditions $y[-1] = 1$ and $y[-2] = -1$, the total response $y[n]$ are wanted.

First, determine the *eigen value* of the difference equation.

Setting $x[n] = 0$ and $y[n] = \lambda^n$ in the difference equation we arrive at

$$\lambda^n + \lambda^{n-1} - 6\lambda^{n-2} = \lambda^{n-2}(\lambda^2 + \lambda^1 - 6)$$

$$= \lambda^{n-2}(\lambda + 3)(\lambda - 2) = 0$$

And hence its *eigen value* (roots) are:

$$\lambda_1 = -3, \qquad \lambda_2 = 2$$

Therefore we can get the *zero input response* as below:

$$y_{zi}[n] = \alpha_1(-3)^n + \alpha_2(2)^n$$

Zero input response

- α_1 **and** α_2 **can be determined by**

$$\begin{cases} y_{zi}[0] = \alpha_1(-3)^0 + \alpha_2(2)^0 = \alpha_1 + \alpha_2 \\ \qquad\quad = -y[-1] + 6y[-2] = -7 \\ y_{zi}[1] = \alpha_1(-3)^1 + \alpha_2(2)^1 = -3\alpha_1 + 2\alpha_2 \\ \qquad\quad = -y[0] + 6y[-1] = 13 \end{cases}$$

$$\alpha_1 = -5.4, \qquad \alpha_2 = -1.6$$

$$y_{zi}[n] = -5.4(-3)^n - 1.6(2)^n$$

Zero state response

Complementary solution: $y_c[n] = \alpha_1(-3)^n + \alpha_2(2)^n$

Particular solution: $\boldsymbol{y_p[n] = \beta \rightarrow \beta = -2,\ n \geq 0}$

Zero-state solution:

$$y_{zs}[n] = \alpha_1(-3)^n + \alpha_2(2)^n - 2$$

Zero all initial conditions :

$$\begin{cases} y[-1] = 0 \\ y[-2] = 0 \end{cases} \quad or \quad \begin{cases} y[0] = x[0] = 8 \\ y[1] = x[1] - y[0] = 0 \end{cases}$$

Zero state response

$$y_{zs}[n] = 3.6(-3)^n + 6.4(2)^n - 2$$

- **The zero-state response is determined by**

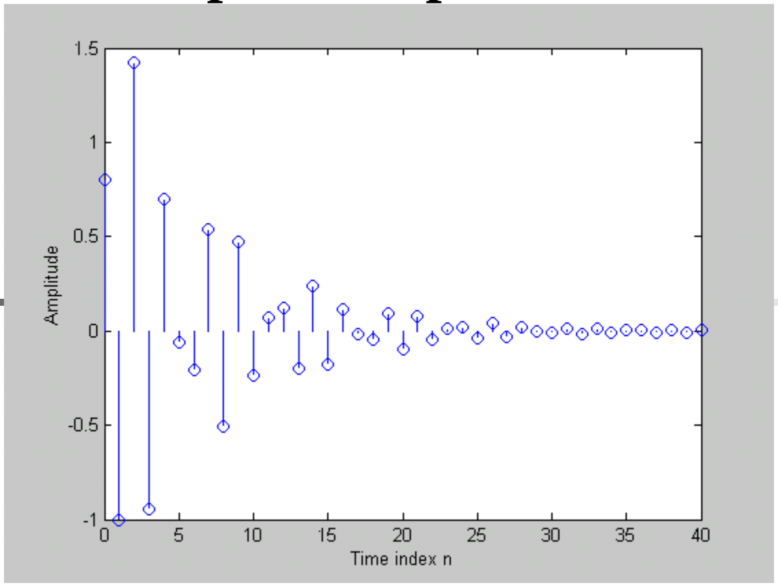
$$y_{zs}[n] = x[n] \circledast h[n]$$

Read Section 4.6 by yourself!!

The MATLAB program of changing input

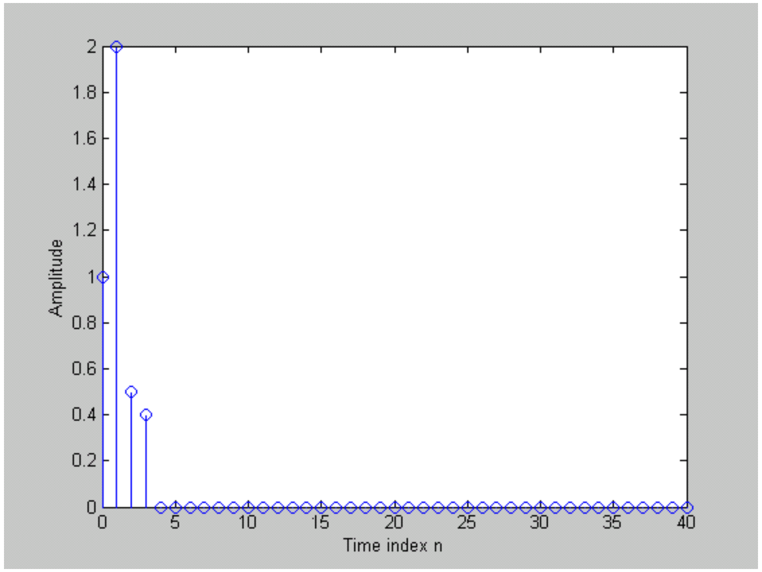
```
% Example 2.37 change input
% Illustration of Impulse Response Computation
N = input('Desired impulse response length = ');
p = input('Type in the vector p =');
d = input('Type in the vector d =');
% x = [1 zeros(1,N-4)];impulse
x = [1 2 0.5 0.4 zeros(1,N-4)];
y = filter(p,d,x);
figure(1);
k = 0:1:N-1;
stem(k,y)
xlabel('Time index n'); ylabel('Amplitude')
figure(2);
k = 0:1:N-1;
stem(k,x)
xlabel('Time index n'); ylabel('Amplitude')
```

Impulse response

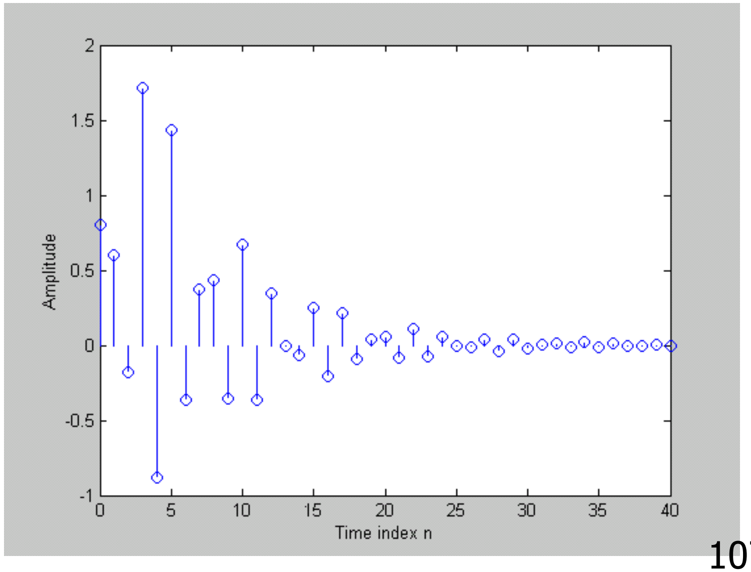


By changing the input signal ,we get response of the same system.

Input signal



response



Key points of Ch.2

1. Sequences in Common use
2. The Basic Operations On Sequences
3. Linear Time-Invariant System
   - Impulse and Step Responses
   - Convolution Sum
   - Generating sequences and analyzing system by MATLAB

Thanks!