University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Digital Signal Processing
EEE 0713-3107

Noor Md Shahriar
Senior Lecturer, Deputy Head of Dept.
Dept. of Electrical & Electronic Engineering
University of Global Village, (UGV), Barishal
Contact: 01743-500587
E-mail: noor.shahriar1@gmail.com

'Imagination is more important than knowledge' – Albert Einstein

Basic Course Information

Course Title	Digital Signal Processing
Course Code	EEE- 0714-3107
Credits	03
CIE Marks	90
SEE Marks	60
Exam Hours	2 hours (Mid Exam) 3 hours (Semester Final Exam)
Level	5th Semester
Academic Session	Winter 2025

Digital Signal Processing (EEE-0714-3107)

3 Credit Course

Class:	**17 weeks (2 classes per week)** **Total Class Duration: 1 hrs.** **Total=34 Hours**
Preparation Leave (PL):	02 weeks
Exam:	04 weeks
Results:	02 weeks
Total:	**25 Weeks**

Attendance:

Students with more than or equal to 70% attendance in this course will be eligible to sit for the Semester End Examination (SEE). SEE is mandatory for all students.

Continuous Assessment Strategy

Quizzes

Altogether 4 quizzes may be taken during the semester, 2 quizzes will be taken for midterm and 2 quizzes will be taken for final term.

Assignment

Altogether 2 assignments may be taken during the semester, 1 assignments will be taken for midterm and 1 assignments will be taken for final term.

Presentation

The students will have to form a group of maximum 3 members. The topic of the presentation will be given to each group and students will have to do the group presentation on the given topic.

ASSESSMENT PATTERN

CIE- Continuous Internal Evaluation (90 Marks)

Bloom's Category Marks	Tests (45)	Quiz (15)	External Participation in Curricular/Co-Curricular Activities (15)
Remember	10	09	Bloom's Affective Domain: (Attitude or will) Attendance: 15 Viva-Voca: 5 Assignment: 5 Presentation: 5
Understand	8	06	
Apply	10		
Analyze	5		
Evaluate	7		
Create	5		

SEE- Semester End Examination (60 Marks)

Bloom's Category	Tests
Remember	**10**
Understand	**10**
Apply	**15**
Analyze	**10**
Evaluate	**10**
Create	**5**

COURSE LEARNING OUTCOME (CLO)

Course learning outcomes (CLO): After successful completion of the course students will be able to -

CLO-1	**Understand** the fundamentals of digital signal processing, including signal and system concepts, classification, and analog-to-digital conversion techniques.
CLO-2	**Analyze** different types of signals, their representations, and key properties such as energy, power, and manipulations.
CLO-3	**Apply** mathematical tools like convolution, Z-transform, and frequency-domain analysis to solve signal processing problems.
CLO-4	**Design** and **implement** digital filters (FIR and IIR) and apply advanced techniques like FFT for practical DSP applications.

SYNOPSIS / RATIONALE

Digital Signal Processing (DSP) is a critical field in electrical engineering, focusing on the manipulation and analysis of signals using digital techniques. This course provides students with a fundamental understanding of signal processing algorithms, methods, and applications. In an increasingly digital world, DSP plays a vital role in various domains such as telecommunications, audio processing, image processing, and biomedical engineering. By mastering DSP principles and techniques, students gain the skills necessary to design, implement, and optimize digital signal processing systems, contributing to advancements in technology and innovation across industries.

Course Objective

- **Understand** fundamental concepts and principles of digital signal processing.
- **Analyze** and **interpret** signals in time and frequency domains.
- **Design** and **implement** digital filters for signal processing applications.
- **Apply** Fourier analysis and Z-transform techniques to analyze signals.
- **Design** FIR & IIR Digital Filters & **analyze.**

COURSE OUTLINE

Sl.	Content of Course	Hrs	CLOs
1	**Introduction to DSP:** Signals, systems and signal processing, Basic Elements of DSP, Advantages and Disadvantages of DSP, Application of DSP, Types of Signal, A/D Conversion, Problems	4	CLO1, CLO2
2	**Discrete Time Signals & Systems:** Representation of discrete time signals, Some elementary discrete time signals, Classification of discrete time (DT) signals, Manipulation of DT signals, Classification of Discrete Time System, Convolution sum, Correlation	4	CLO2, CLO3
3	**Analysis of DT Linear Time-Invariant System** Sampling theorem, aliasing, quantization error, Nyquist rate problems.	4	CLO2, CLO3

COURSE OUTLINE

Sl.	Content of Course	Hrs	CLOs
5	**Z-Transform**: Z-transform, Physical significance of z-transform, Region of convergence (ROC), Z-transform of some basic causal and anti-causal signals, Properties of z-transform, Pole-zero Plot, Inverse z-transform	4	CLO3, CLO4
6	**Frequency Analysis**: FIR System, structures for FIR System, Direct form realization, Examples related to FIR system implementation, IIR system, Structures for FIR System, Direct form structures of IIR system, DFT, DTFT, FFT algorithms, Circular Convolution	6	CLO4, CLO5
7	**Digital Filters**: Filter kernel, classification, FIR and IIR design, kernel conversion, spectral inversion.	6	CLO4, CLO5

COURSE SCHEDULE

Week	Topic	Teaching Learning Strategy	Assessment Strategy	Corresponding CLOs
1	**Introduction to Digital Signal Processing (DSP)**: Definition of Signal, System, Basic Block Diagram, Advantages, Limitations & Applications of DSP	- Lecture, multimedia presentation.	Class participation, Group Discussion, Q&A.	CLO1
2	**Signal Classification & Analog-to-Digital Conversion**: Types of signals, steps to convert Analog to Digital signals.	- Interactive lecture; problem-solving session.	Group Discussion Q&A.	CLO1, CLO2
3	**Sampling Theorem and Quantization**: Alias frequency, quantization error, SQNR; Nyquist rate problems.	- Example problems and graphical representation.	Group Discussion Q&A.	CLO2, CLO3
4	**Representation Methods of Signals**: Various representation methods, elementary signals.	- Class examples, signal sketching exercises.	Class Test-1	CLO2

COURSE SCHEDULE

Week	Topic	Teaching Learning Strategy	Assessment Strategy	Corresponding CLOs
5	**Energy and Power of Signals**: Determining energy, power, and signal classification.	- Lecture and guided practice; worked examples.	- Problem-solving exercise.	CLO 2, CLO 3
6	**Signal Manipulation**: Basic operations like shifting, scaling, and folding of signals.	- Practical demonstrations and group exercises.	- In-class problems on signal operations.	CLO 2, CLO 3
7	**Discrete Systems**: Block diagram representation, system classification (linear/nonlinear, causal/noncausal).	- Diagrammatic explanations; group discussions.	Problem-solving exercise, Q&A, Class Participation	CLO 1, CLO 4
8	**Linear Convolution and Correlation**: Cross-correlation, auto-correlation.	- Hands-on practice with numerical problems.	Class Test-2	CLO 3
9	**Introduction to Z-Transform**: Significance, ROC, and basic concepts.	- Lecture and practice session on Z-transform.	Assignment-1	CLO 3, CLO 4

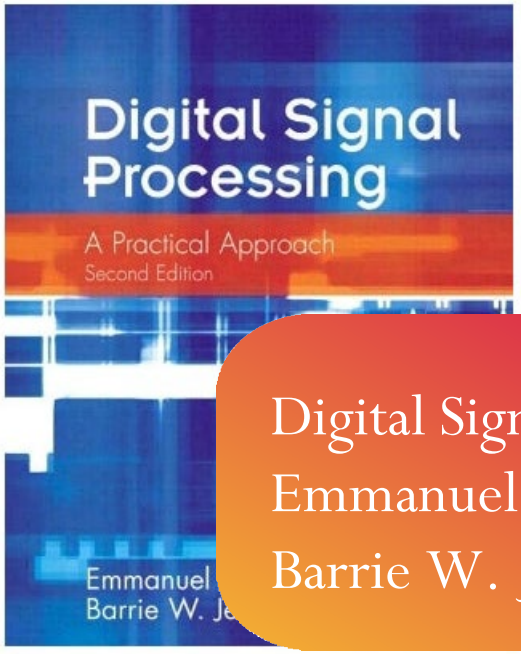
COURSE SCHEDULE

Week	Topic	Teaching Learning Strategy	Assessment Strategy	Corresponding CLOs
10	**Properties & Inverse Z-Transform**: Key properties and methods for inverse Z-transform.	- Problem-solving exercises.	- Homework on inverse Z-transform.	CLO 3, CLO 4
11	**FIR and IIR Systems**: Structure and implementation basics.	- Lecture, block diagram examples.	- Lab assignment on FIR and IIR systems.	CLO 4
12	**DFT, DTFT, and Circular Convolution**: Understanding frequency domain analysis.	- Case studies and numerical problems.	- Quiz on frequency-domain transformations.	CLO 3, CLO 4
13	**Radix-2 FFT Algorithm**: 8-point DIT-FFT butterfly algorithm.	- Simulation using MATLAB; example calculations.	- Lab assignment on FFT implementation.	CLO 4, CLO 5
14	**Digital Filters**: Advantages, disadvantages, applications, classification, filter kernel.	- Interactive lecture with real-life examples.	- Short essay on filter applications.	CLO 4, CLO 5

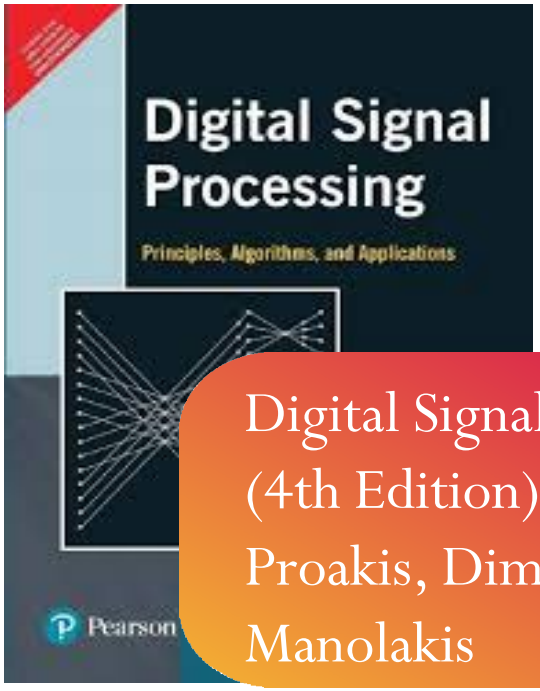
COURSE SCHEDULE

Week	Topic	Teaching Learning Strategy	Assessment Strategy	Corresponding CLOs
15	**Filter Kernel Conversion**: Spectral inversion.	- Demonstrations and exercises.	Class Test-3	CLO 4
16	**FIR Filter Design**: Design techniques and practical considerations.	- Design session with tools like MATLAB.	- Lab-based FIR filter design task.	CLO 5
17	**IIR Filter Design**: Analysis, design methods, and simulation.	- Lecture, software-based design (MATLAB or Python).	Assignment-2	CLO 5
18	**Course Review and Final Examination**	- Revision and problem-solving workshop.	- Summative assessment (written exam).	CLO 1–5

REFERENCE BOOK



Digital Signal Processing - Emmanuel C. Ifeachor, Barrie W. Jervis



Digital Signal Processing (4th Edition), John G. Proakis, Dimitris K Manolakis

YouTube

Video Lecture Playlist
https://youtube.com/playlist?list=PLuh62Q4Sv7BUSzx5Jr8Wrxxn-U10qG1et&si=FTy0rlueWRQIWQYj

Bloom Taxonomy Cognitive Domain Action Verbs

Remembering (C1)	Choose • Define • Find • How • Label • List • Match • Name • Omit • Recall • Relate • Select • Show • Spell • Tell • What • When • Where • Which • Who • Why
Understanding (C2)	Classify • Compare • Contrast • Demonstrate • Explain • Extend • Illustrate • Infer • Interpret • Outline • Relate • Rephrase • Show • Summarize • Translate
Applying (C3)	Apply • Build • Choose • Construct • Develop • Experiment with • Identify • Interview • Make use of • Model • Organize • Plan • Select • Solve • Utilize
Analyzing (C4)	Analyze • Assume • Categorize • Classify • Compare • Conclusion • Contrast • Discover • Dissect • Distinguish • Divide • Examine • Function • Inference • Inspect • List • Motive • Relationships • Simplify • Survey • Take part in • Test for • Theme
Evaluating (C5)	Agree • Appraise • Assess • Award • Choose • Compare • Conclude • Criteria • Criticize • Decide • Deduct • Defend • Determine • Disprove • Estimate • Evaluate • Explain • Importance • Influence • Interpret • Judge • Justify • Mark • Measure • Opinion • Perceive • Prioritize • Prove • Rate • Recommend • Rule on • Select • Support • Value
Creating (C6)	Adapt • Build • Change • Choose • Combine • Compile • Compose • Construct • Create • Delete • Design • Develop • Discuss • Elaborate • Estimate • Formulate • Happen • Imagine • Improve • Invent • Make up • Maximize • Minimize • Modify • Original • Originate • Plan • Predict • Propose • Solution • Solve • Suppose • Test • Theory

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-1
Introduction of Digital Signal Processing

Course Code: EEE-0714-3103
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Week 1
Slide 16-23

Contents

- ✓ Signals, systems and signal processing.
- ✓ Basic Elements of DSP.
- ✓ Advantages and Disadvantages of DSP.
- ✓ Application of DSP.
- ✓ Types of Signal.
- ✓ A/D Conversion.
- ✓ Problems

Signals, Systems and Signal Processing

Signal

A signal is defined as a function representing any physical quantity that varies with time, space or any other independent variable or variables. It contains information about the behavior or nature of the phenomenon. Mathematically, we describe a signal as a function of one or more independent variables. For example,

$$S_1(t) = 5t$$

$$S(xy) = 3x + 2xy + 5y^2$$

- ✓ Speech, electrocardiogram and electroencephalogram signals are examples of information

  bearing signals that evolve as function of a single independent variable.
- ✓ Two dimensional signal An example of a signal that is a function of two independent variable is an image signal.
- ✓ A video signal is function of three independent variables.

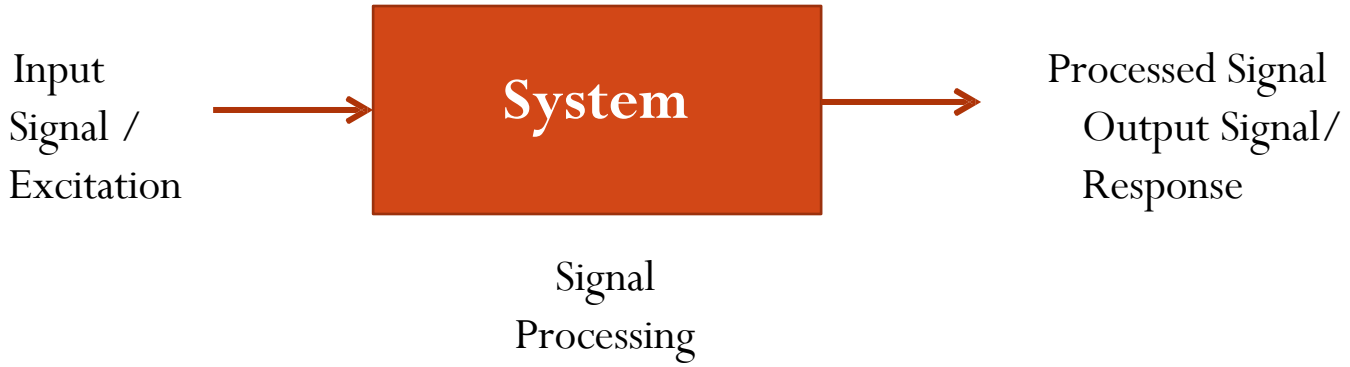
Signals, Systems and Signal Processing (Cont.)

System

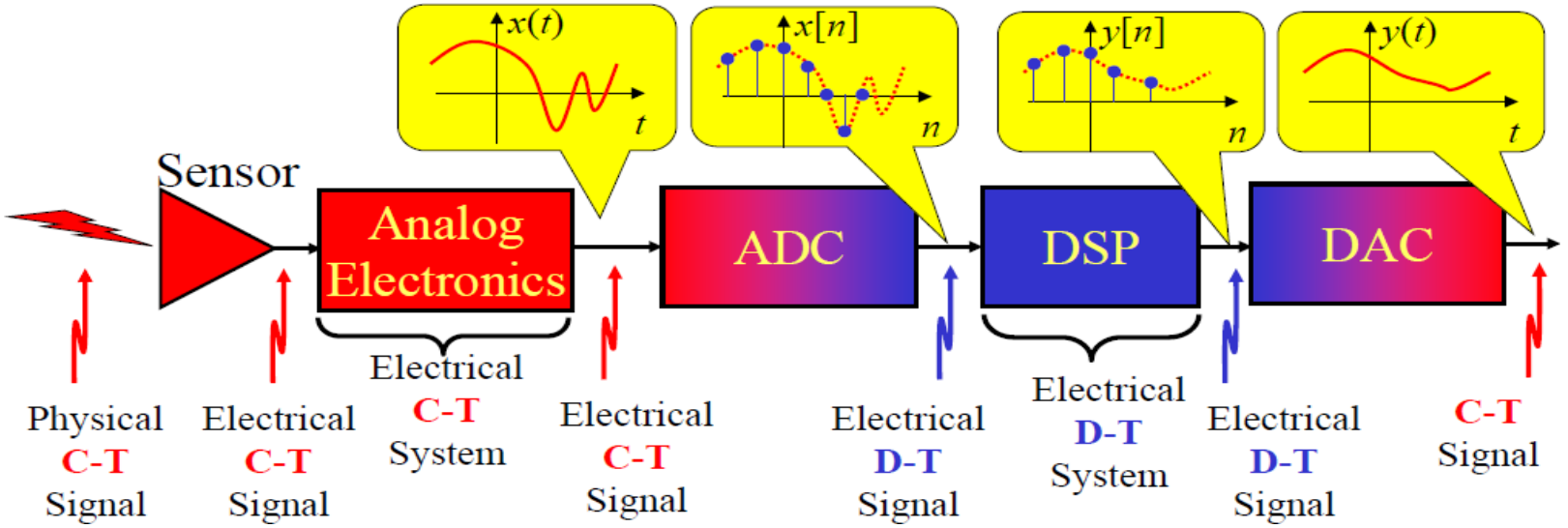
A system may be defined as a physical device that performs an operation on a signal. System is a mathematical model of a Physical process that relates the input (Excitation) to the Output (Response). For example, a filter used to reduce the noise and interference corrupting a desired information bearing signal is a system.

Signal Processing

When we pass a signal through a system, we say that we have processed the signal.



Basic Elements of a DSP System



A/D Converter: Digital Signal Processing provides an alternative method for processing the analog signal. To perform the processing digitally, there is a need for an interface between the analog signal and the digital processor. This interface is called analog-to-digital (A/D) converter.

DSP: The digital signal processor may be a large programmable digital computer or a small microprocessor programmed to perform the desired operation in the input signal.

D/A Converter: The digital output from the digital signal processor is to be given to the user in analog form. This is done by another interface called a digital-to-analog(D/A) converter.

Advantages of Digital over Analog Signal Processing

1)DSP Systems are reconfigurable: A digital programmable system allows flexibility in configuring the digital signal processing operations simply by changing the program. Reconfiguration of an analog system usually a redesign of hardware followed by testing and verification to see that it operates properly.

2)Accuracy Consideration: Tolerances in analog circuit components make it extreme difficult for the system designer to control the accuracy of an analog signal processing system. On the other hand, a digital system provides much better control of accuracy requirements.

3)Storing Data: Digital signals are easily stored on magnetic media (tape or disk) without deterioration or loss of signal fidelity beyond that introduce in the A/D conversion. As a consequence, the signals become transportable and can be processed off-line in a remote laboratory.

Advantages of Digital over Analog Signal Processing (Cont.)

4)Signal Processing Algorithm: The digital signal processing method also allows for the implementation of more sophisticated signal processing algorithms. It is usually very difficult to perform precise mathematical operations on a signal in analog form, but these same operations can be routinely implemented on a digital computer using software.

5)Cost: In some cases a digital implementation of the signal processing system is cheaper than its analog counterpart.

6)Effect of Noise: Digital Signal can convey information with greater noise immunity.

7)Electromagnetic interference: There is minimum electromagnetic interference in digital technology.

8)Security & bandwidth : It is more secure and higher rate transmission with wider bandwidth.

Disadvantages of Digital Signal Processing

1. **Speed of operation:** One practical limitation is the speed of operation of A/D converters and digital signal processors. We shall see that signals having extremely wide band widths require fast-sampling rate A /D converters and fast digital signal processors. Hence there are analog signals with large bandwidths for which a digital processing approach is beyond the state of the art of digital hardware.
2. **Reconstruction:** The process of reconstructing analog signal from the digital signal is very difficult.
3. **Expensive for small applications.**
4. **Finite precision effect.**

Application of DSP

DSP

- **Space**
  - -Space photograph enhancement
  - -Data compression
  - -Intelligent sensory analysis by remote space probes
- **Medical**
  - -Diagnostic imaging (CT, MRI, ultrasound, and others)
  - -Electrocardiogram analysis
  - -Medical image storage/retrieval
- **Commercial**
  - -Image and sound compression for multimedia presentation
  - -Movie special effects
  - -Video conference calling
- **Telephone**
  - -Voice and data compression
  - -Echo reduction
  - -Signal multiplexing
  - -Filtering
- **Military**
  - -Radar
  - -Sonar
  - -Ordnance guidance
  - -Secure communication
- **Industrial**
  - -Oil and mineral prospecting
  - -Process monitoring & control
  - -Nondestructive testing
  - -CAD and design tools
- **Scientific**
  - -Earthquake recording & analysis
  - -Data acquisition
  - -Spectral analysis
  - -Simulation and modeling

Current Research Trends in DSP

Week 2
Slide 25-30

Classification of Signals: Multichannel and Multidimensional

Multichannel Signal

In some application, signals are generated by multiple source or multiple sensors. Such signals, in turn, can be represented in vector form. We refer to such a vector of signals as a multichannel signal.

In electrocardiogram, for example, 3-lead and 12-electrocardiogram (ECG) are often used in practice which result in a 3 channel and 12 channel signals.

Multidimensional Signal

If the signal is a function of a single independent variable, the signal is called a one dimensional signal. On the other hand, a signal is called M-dimensional if its value is a function of M independent variable.

Black and white picture is an example of a two-dimensional signal, since the intensity or brightness I(x,y) at each point is a function of two independent variables.

Black and white TV picture me be treated as a three-dimensional signal.

Color TV picture is a three-channel and three-dimensional signal.

$$I(x, y, t) = \begin{bmatrix} I_r(x, y, t) \\ I_g(x, y, t) \\ I_b(x, y, t) \end{bmatrix}$$

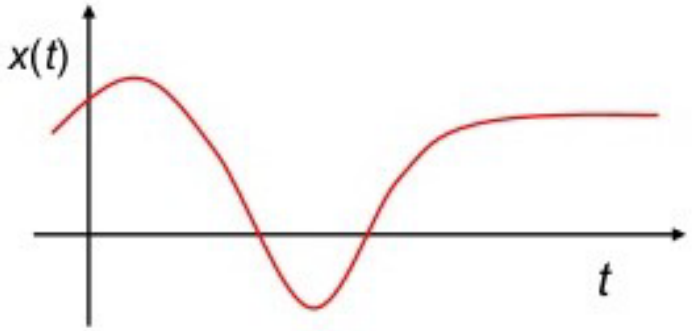
Classification of Signals: Continuous Time and Discrete Time Signal

Continuous Time Signal

Continuous signals or analog signals are defined for every value of time and they take on values in the continuous interval (a,b), where **a** can be $-\infty$ and **b** can be $+\infty$

Examples: $x_1(t) = \cos(\pi t)$

$x_2(t) = e^{-|t|}$ $\quad where\ t = -\infty < t < \infty$



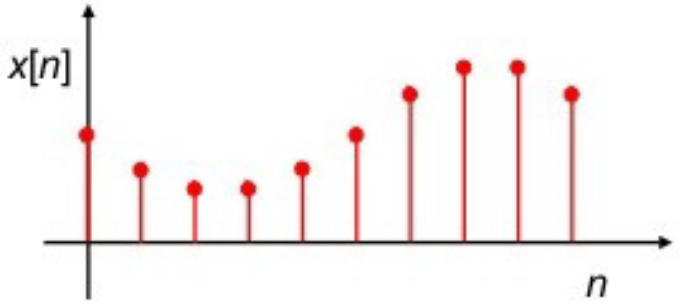
Discrete time Signal

Discrete time signals are defined only at certain specific values of time. These time instants need not be equidistant, but in practice they are usually taken at equally spaced intervals.

Examples:

$$x_1(n) = \begin{matrix} 0.8^n, & n \geq 0 \\ 0, & otherwise \end{matrix}$$

n is integer number.



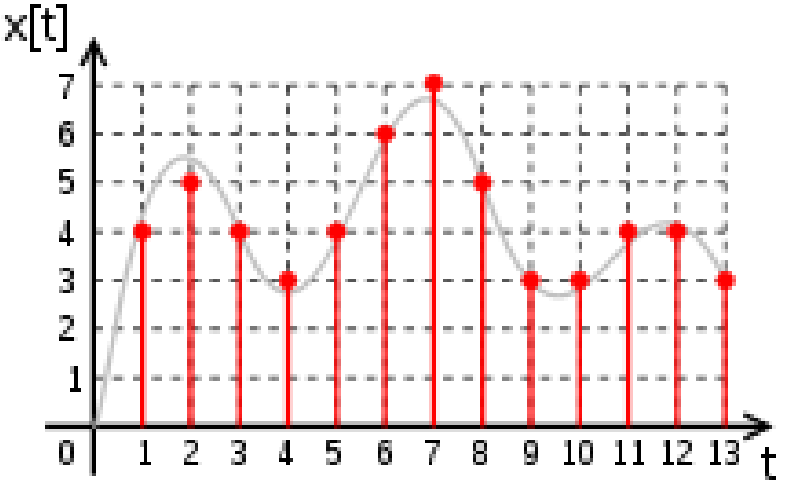
Classification of Signals: Continuous Valued and Discrete Valued

The values of a continuous time or discrete time signal can be continuous or discrete.

Continuous Valued Signal: If a signal takes on all possible values on a finite or an infinite range, it is said to be a continuous valued signal.

Discrete Valued Signal: Alternatively, if the signal takes on values from a finite set of possible values, it is said to be discrete-valued signal.

Digital Signal: A discrete time signal having a set of discrete values is called a digital signal. In order for a signal to be processed digitally, it must be discrete in time and its values must be discrete (i.e. it must be digital signal)



Classification of Signals: Deterministic and Random Signal

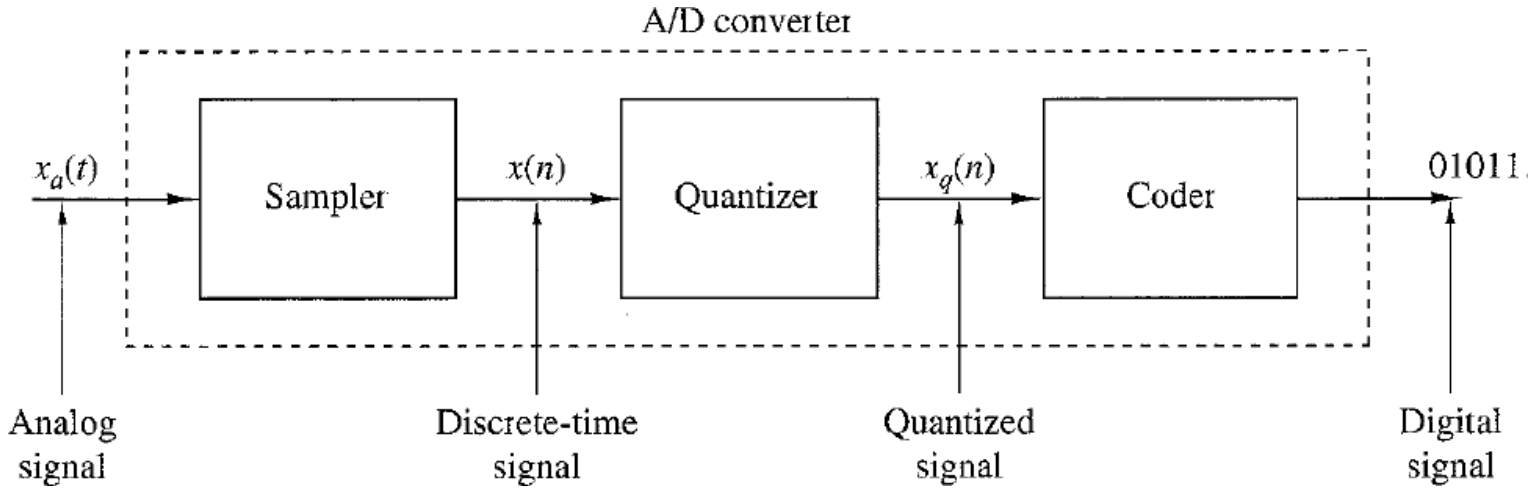
Deterministic (Predictable/Wanted) Signal : Any signal that can be uniquely described by an explicit mathematical expression, a table of data or a well defined rule is called deterministic. This term is used to emphasize the fact that all past, present and future values of the signal are known precisely without any uncertainty.

Random Signal (Unpredictable/ unwanted/ noise): In many practical application the signals can not be described to any reasonable degree of accuracy by explicit mathematical formulas, or such description is too complicated to be any practical use.

The lack of such a relationship implies that such signals evolve in time is an unpredictable manners. We refer to these signals as random.

The o/p of noise generation, the speech signal are example of random signals.

Analog to digital Conversion



Sampling : This is the conversion o f a continuous-time signal into a discrete time signal obtained by taking " samples'" of the continuous-time signal at discrete-time instants. Thus, if $x_a(t)$ is the input to the sampler, the output is $x_a(nT) = x(n)$, where T is called the sampling interval.

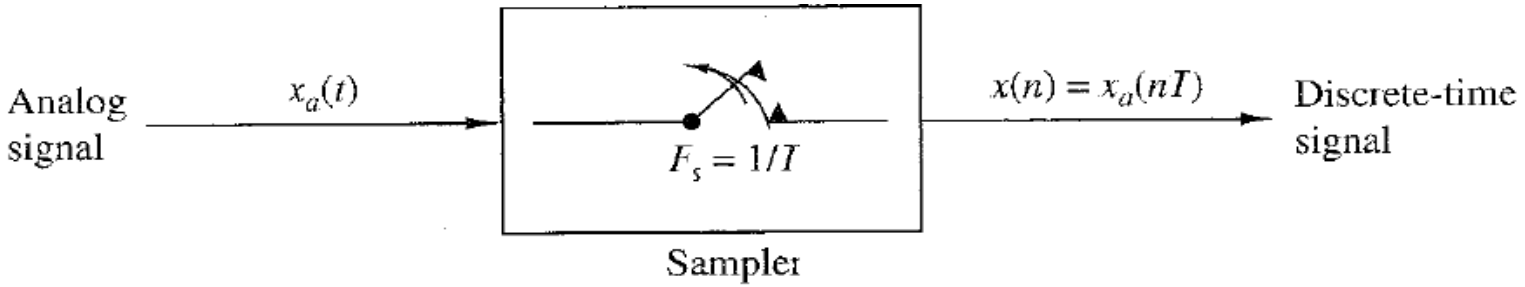
Quantization : This is the conversion o f a discrete-time continuous-valued signal $x(n)$ in to a discrete-time, discrete-valued (digital) signal $x_q(n)$. The value of each signal sample is represented by a value selected from a finite set o f possible values.

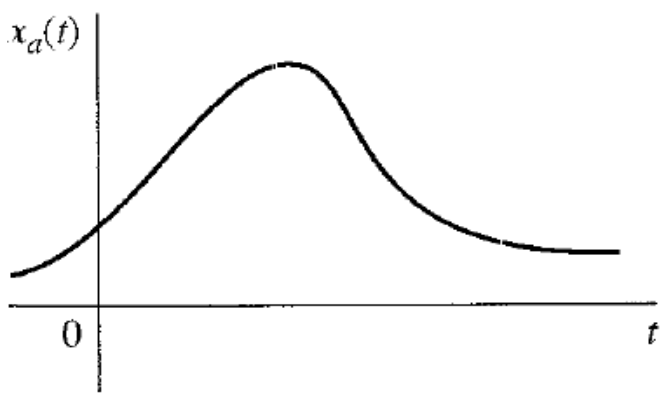
Coding: In the coding process, each discrete value $x_q(n)$ is represented by a b-bit binary sequence.

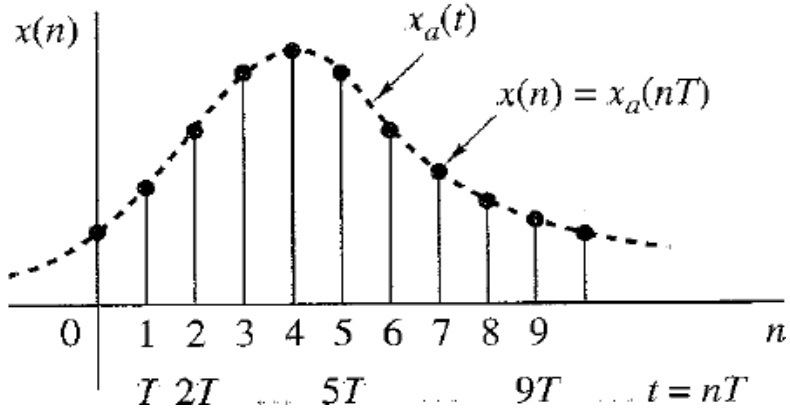
Sampling of Analog Signals

Periodic or uniform sampling is described by the relation- $x(n) = x_a(nT), \quad -\infty \leq n \leq \infty$

where x(n) is the discrete-time signal obtained by " taking samples" of the analog signal $x_a(t)$ at every T seconds. The time interval T between successive samples is called the sampling period or sample interval and its reciprocal $^1\ _T = F_s$ is called the sampling rate (samples per second) or the sampling frequency (hertz).







Week 3
Slide 32-45

Sampling of Analog Signals (Cont.)

Periodic sampling establishes a relationship between the time variables t and n of continuous-time and discrete-time signals.

$$t = nT = \frac{n}{F_s}$$

If the analog signal $x_a(t) = A\cos(2\pi Ft + \theta)$

Sampled periodically at a rate $F_s = {}^{1}\ _{T}$, the digital signal can be expressed as

$$x_a(nT) \equiv x(n) = A\cos(2\pi FnT + \theta)$$

$$= A\cos\left(\frac{2\pi nF}{F_s} + \theta\right)$$

From above relationship between the frequency variable F (or Ω) for analog signals and the frequency variable f (or ω) for discrete-time signals.

$$f = \frac{F}{F_s} \qquad \omega = \Omega T$$

The frequency variable f of discrete signal is sometimes called Relative normalized frequency

Alias of Frequency

Consider two sinusoidal analog Signals: $x_1(t) = \cos 2\pi(10)t$

$$x_2(t) = \cos 2\pi(50)t$$

If they are sampled at a rate $F_s = 40\ Hz$, The corresponding discrete time signal will be:

$$x_1(n) = \cos 2\pi \left(\frac{10}{40}\right) n = \cos \frac{\pi}{2} n$$

$$x_2(n) = \cos 2\pi \left(\frac{50}{40}\right) n = \cos \frac{5\pi}{2} n$$

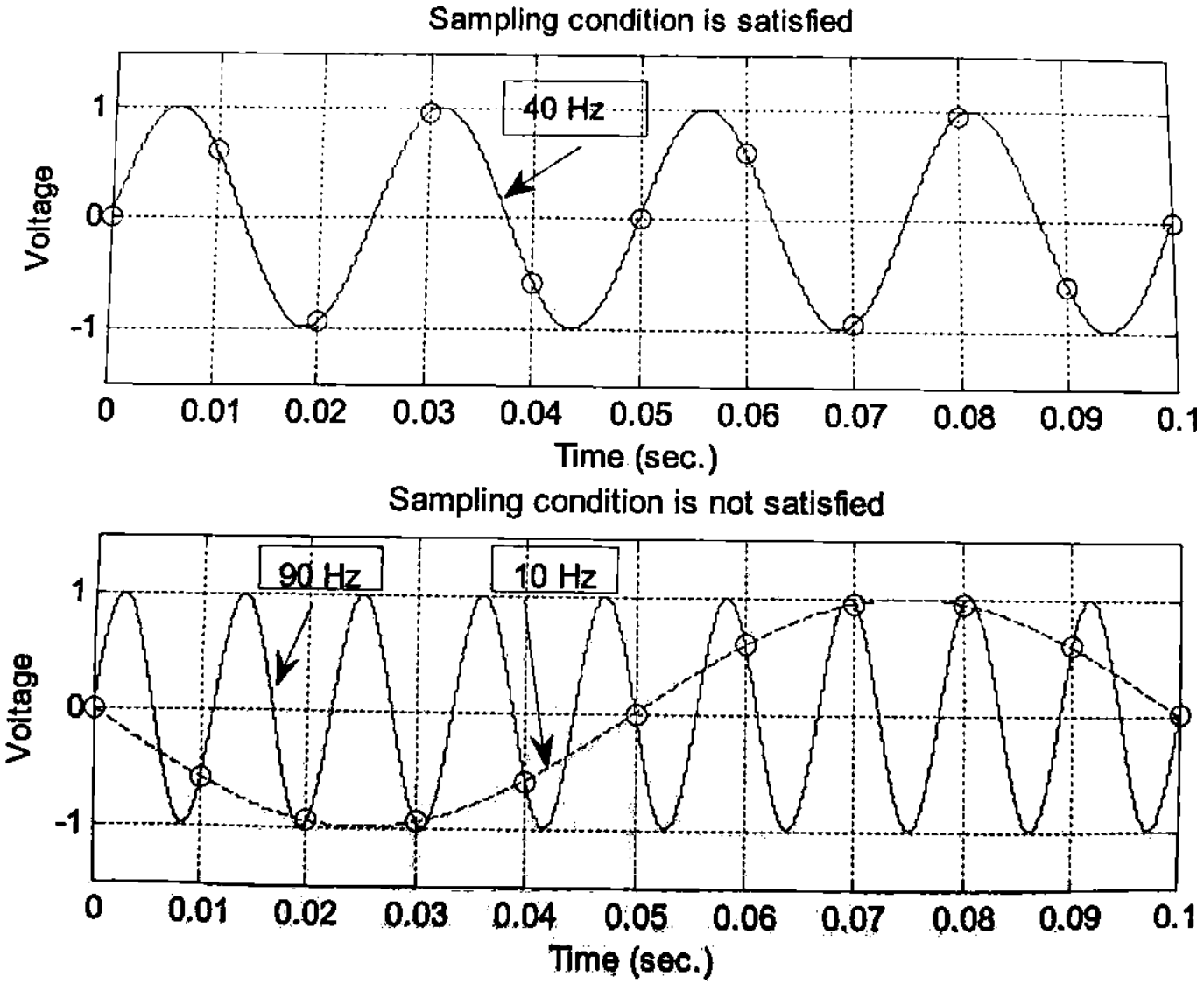
However, $\cos 5\pi n/2 = \cos(2\pi n + \pi n/2) = \cos \pi n/2$. Hence $x_2(n) = x_1(n)$

Thus the sinusoidal signals are identical and consequently, indistinguishable.

Since $x_2(t)$ yields exactly the same values as $x_1(t)$ when the two are sampled at $F_s = 40$ samples per second, we say that the frequency $F_2 = 50\ Hz$ is an **alias** of the frequency $F_1 = 10\ Hz$ at the sampling rate of 40 samples per second.

It is important to note that F_2 is not only the alias of F_1. In fact at the sampling rate of 40 samples per second, the frequency $F_3 = 90\ Hz, F_4 = 130$ Hz ..... So on are also an alias of F_1.
In general, all of the sinusoids $\cos 2\pi\,(F_1 + 40K)t\,, k = 1,2,3\ldots.$, sampled at 40 samples per second are the aliases of $F_1 = 10\ Hz$.

Alias of Frequency (What should be the sampling Rate?)



Example 1.4.2 (Proakis)

Consider the analog signal

$$x_a(t) = 3\cos 100\pi t$$

(a) Determine the minimum sampling rate required to avoid aliasing.

(b) Suppose that the signal is sampled at the rate $F_s = 200$ Hz. What is the discrete-time signal obtained after sampling?

(c) Suppose that the signal is sampled at the rate $F_s = 75$ Hz. What is the discrete-time signal obtained after sampling?

(d) What is the frequency $0 < F < F_s/2$ of a sinusoid that yields samples identical to those obtained in part (c)?

Solution.

(a) The frequency of the analog signal is $F = 50$ Hz. Hence the minimum sampling rate required to avoid aliasing is $F_s = 100$ Hz.

(b) If the signal is sampled at $F_s = 200$ Hz, the discrete-time signal is

$$x(n) = 3\cos\frac{100\pi}{200}n = 3\cos\frac{\pi}{2}n$$

(c) If the signal is sampled at $F_s = 75$ Hz, the discrete-time signal is

$$x(n) = 3\cos\frac{100\pi}{75}n = 3\cos\frac{4\pi}{3}n = 3\cos\left(2\pi - \frac{2\pi}{3}\right)n = 3\cos\frac{2\pi}{3}n$$

Example 1.4.2 (Proakis)- Cont.

(d) For the sampling rate of $F_s = 75$ Hz, we have

$$F = f F_s = 75 f$$

-The frequency of the sinusoid in part (c) is $f = \frac{1}{3}$. Hence

$$F = 25 \text{ Hz}$$

Clearly, the sinusoidal signal

$$y_a(t) = 3 \cos 2\pi F t$$

$$= 3 \cos 50\pi t$$

sampled at $F_s = 75$ samples/s yields identical samples. Hence $F = 50$ Hz is an alias of $F = 25$ Hz for the sampling rate $F_s = 75$ Hz.

Sampling Theorem

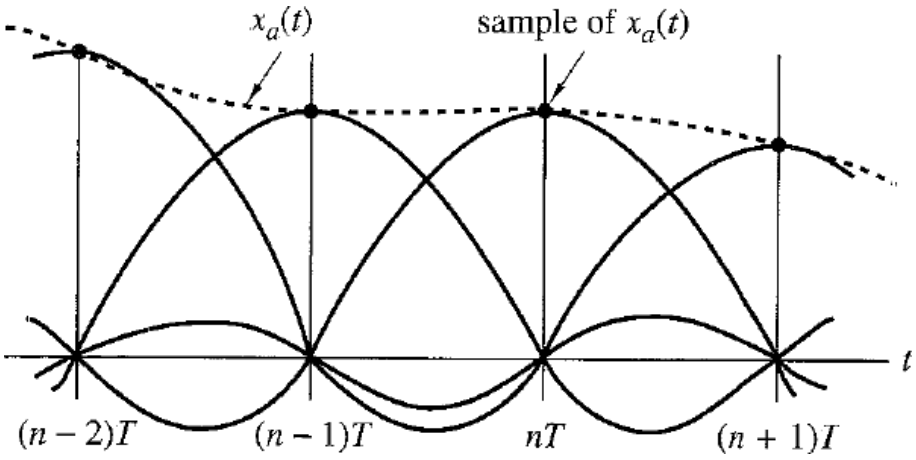
If the highest frequency contained in an analog signal $x_a(t)$ is $\boldsymbol{F_{max} = B}$ and the signal is sampled at a rate $\boldsymbol{F_s > 2F_{max} \equiv 2B}$. then $x_a(t)$ can be exactly recovered from it sample values using the interpolation function

$$g(t) = \frac{\sin 2\pi Bt}{2\pi Bt}$$

$x_a(t)$ may be expressed as $$x_a(t) = \sum_{n=-\infty}^{\infty} x_a\left(\frac{n}{F_s}\right) g\left(t - \frac{n}{F_s}\right)$$

Where $x_a\left(\frac{n}{F_s}\right) = x_a(nT) \equiv x(n)$ are the samples of $x_a(t)$

The minimum sampling rate of a signal to recover it from its sample value is $\boldsymbol{F_N = 2F_{max} = 2B}$ **is called the Nyquist rate.**



Quantization of Continuous-Amplitude Signal

Quantization: The process of converting a discrete-time continuous-amplitude signal into a digital signal by expressing each sample value as a finite (instead of an infinite) number of digits is called quantization.

Quantization error: The error introduced in representing the continuous-valued signal by a finite set of discrete value levels is called quantization error or quantization noise.

The quantization error is a sequence $e_q(n)$ defined as the difference between the quantized value ($x_q(n) = Q[x(n)]$) and the actual sample value-

$$e_q(n) = x_q(n) - x(n)$$

Resolution: The values allowed in the digital signal are called the quantization levels, whereas the distance Δ between two successive quantization levels is called the quantization step size or resolution.

The quantization error *eq (n)* in round ing is limited to the range of $-\Delta_2$ to $+\Delta_2$, that is,

$$-\frac{\Delta}{2} \leq e_q(n) \leq \frac{\Delta}{2}$$

In other words, the instantaneous quantization error cannot exceed half of the quantization step.

Quantization of Continuous-Amplitude Signal (Cont.)

If x_{min} and x_{max} represent the minimum and maximum value of x(n) and L is the number of quantization levels, then

$$\Delta = \frac{x_{max} - x_{min}}{L - 1}$$

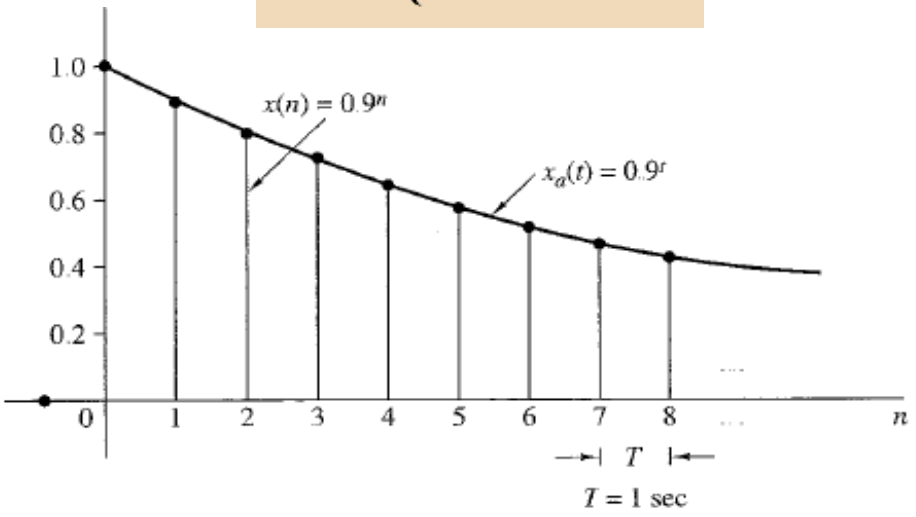
$x_{max} - x_{min}$ is known as the dynamic range of signal

For the example in Fig,
x_{max}=1 and $x_{min} = 0, L = 11$
So, $\Delta = 0.1$

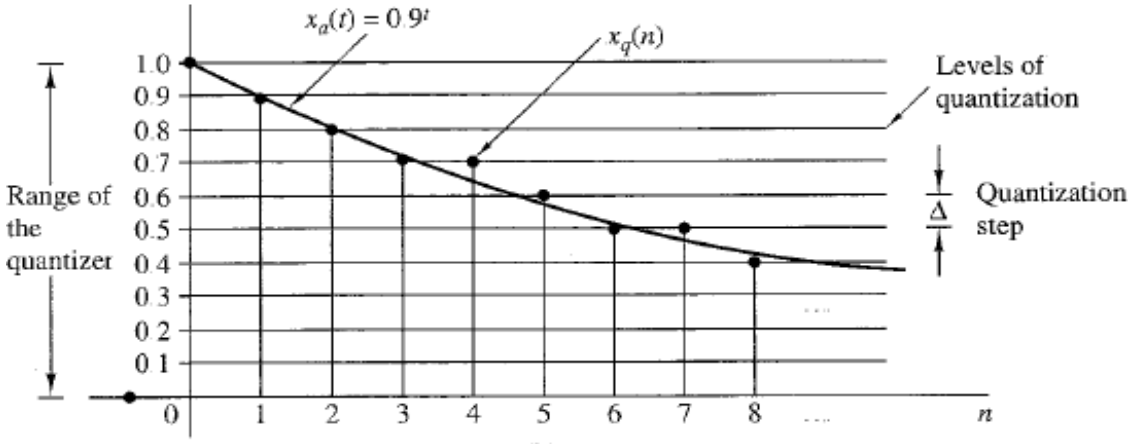
Note that if the dynamic range is fixed, in creasing the number of quantization levels, L results in a decrease o f the quantization step size. Thus the quantization error decreases and the accuracy o f the quantizer increases.

Illustrating the quantization process for the function:

$$x(n) = \begin{cases} 0.9^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

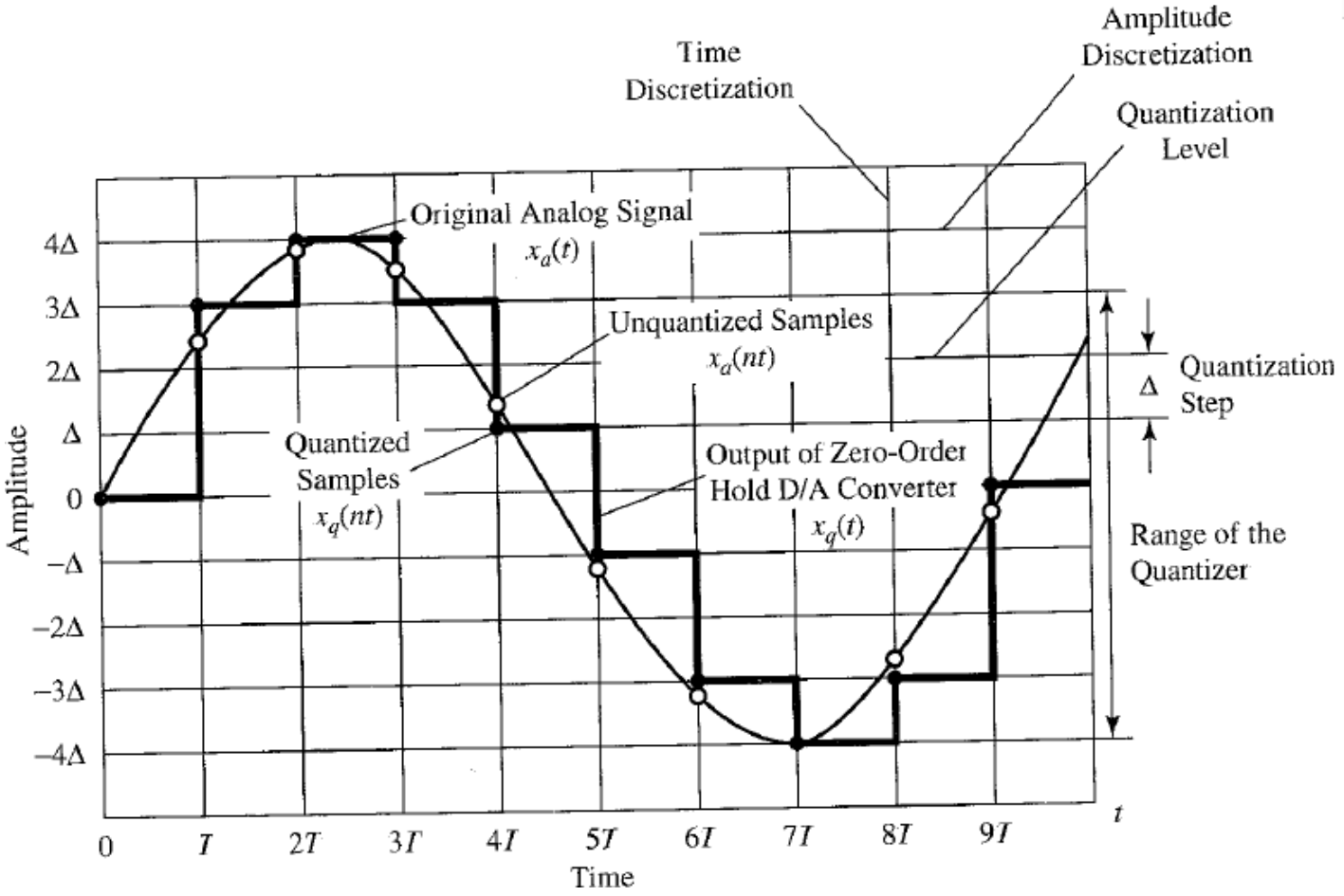


(a)



(b)

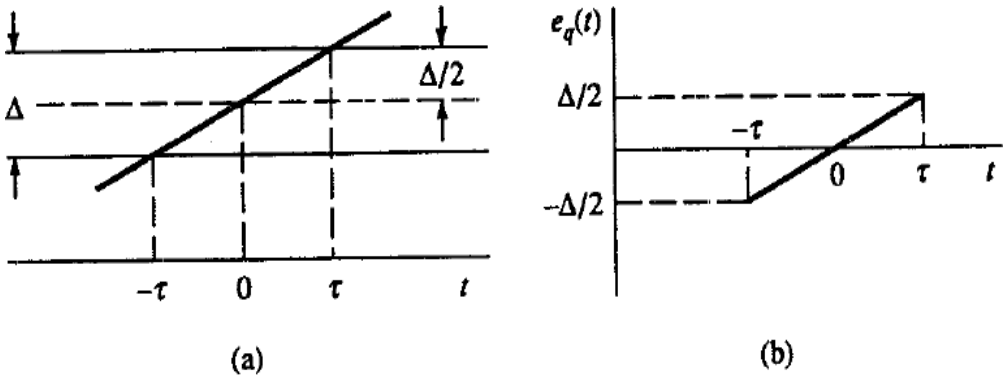
Quantization of Sinusoidal Signal



The analog sinusoidal signal, $x_a(t) = A\cos\Omega_0 t$

The discrete sinusoidal signal, $\mathrm{x(n)} = x_a(nT)$

Quantization of Sinusoidal Signal (Cont.)



The quantization error $e_q(t) = x_a(t) - x_q(t)$.

$$P_q = \frac{1}{2\tau}\int_{-\tau}^{\tau} e_q^2(t)\,dt = \frac{1}{\tau}\int_{0}^{\tau} e_q^2(t)\,dt$$

Since $e_q(t) = (\Delta/2\tau)t$, $-\tau \le t \le \tau$, we have

$$P_q = \frac{1}{\tau}\int_{0}^{\tau}\left(\frac{\Delta}{2\tau}\right)^2 t^2\,dt = \frac{\Delta^2}{12}$$

Here, $x_{min} = -A$ and $x_{max} = A$, If the quantizer has b bit accuracy, $\Delta = \frac{A+A}{2^b} = \frac{2A}{2^b}$

$$P_q = \frac{A^2/3}{2^{2b}}$$

Quantization of Sinusoidal Signal (Cont.)

The average power of the signal $x_a(t)$ is

$$P_x = \frac{1}{T_p}\int_0^{T_p} (A\cos\Omega_0 t)^2\, dt = \frac{A^2}{2}$$

The quality of the output of the A/D converter is usually measured by the *signal-to-quantization noise ratio (SQNR)*, which provides the ratio of the signal power to the noise power:

$$\text{SQNR} = \frac{P_x}{P_q} = \frac{3}{2}\cdot 2^{2b}$$

Expressed in decibels (dB), the SQNR is

$$\text{SQNR(dB)} = 10\log_{10}\text{SQNR} = 1.76 + 6.02b$$

This implies that the SQNR increases approximately 6 dB for every bit added to the word length , that is. for each doubling of the quantization levels. Although this formula was derived for sinusoidal signals, but similar result holds for every signal whose dynamic range spans the range of the quantizer. This relationship is extremely important because it dictates the number of bits required by a specific application to assure a given signal-to noise ratio. For example, most compact disc players use a sampling frequency of 44.1 kHz and 16-bit sample resolution , which implies a SQNR of more than 96 dB.

Example 1.4.4 (Proakis)

Consider the analog signal

$$x_a(t) = 3\cos 2000\pi t + 5\sin 6000\pi t + 10\cos 12{,}000\pi t$$

(a) What is the Nyquist rate for this signal?

(b) Assume now that we sample this signal using a sampling rate $F_s = 5000$ samples/s. What is the discrete-time signal obtained after sampling?

(c) What is the analog signal $y_a(t)$ that we can reconstruct from the samples if we use ideal interpolation?

Solution.

(a) The frequencies existing in the analog signal are

$$F_1 = 1 \text{ kHz}, \qquad F_2 = 3 \text{ kHz}, \qquad F_3 = 6 \text{ kHz}$$

Thus $F_{\max} = 6$ kHz, and according to the sampling theorem,

$$F_s > 2F_{\max} = 12 \text{ kHz}$$

The Nyquist rate is

$$F_N = 12 \text{ kHz}$$

Example 1.4.4 (Proakis)- Cont.

(b) Since we have chosen $F_s = 5$ kHz, the folding frequency is

$$\frac{F_s}{2} = 2.5 \text{ kHz}$$

and this is the maximum frequency that can be represented uniquely by the sampled signal. By making use of (1.4.2) we obtain

$$x(n) = x_a(nT) = x_a\left(\frac{n}{F_s}\right)$$

$$= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(\tfrac{3}{5}\right)n + 10\cos 2\pi\left(\tfrac{6}{5}\right)n$$

$$= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(1 - \tfrac{2}{5}\right)n + 10\cos 2\pi\left(1 + \tfrac{1}{5}\right)n$$

$$= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(-\tfrac{2}{5}\right)n + 10\cos 2\pi\left(\tfrac{1}{5}\right)n$$

$$x(n) = 13\cos 2\pi\left(\tfrac{1}{5}\right)n - 5\sin 2\pi\left(\tfrac{2}{5}\right)n$$

Since, $F_s = 5\ KHz$, the folding frequency is $F_s/2 = 2.5\ KHz$. This is the maximum frequency that can be represented uniquely by the sampled signal. The frequency F_1 is less than $F_s/2$ and thus is not affected by aliasing. However the other two frequencies are below the folding frequency and they will be changed by the aliasing effect.

Example 1.4.4 (Proakis)- Cont.

(c) Since the frequency components at only 1 kHz and 2 kHz are present in the sampled signal, the analog signal we can recover is

$$ya(t) = 13\cos 2000\pi t - 5\sin 4000\pi t$$

which is obviously different from the original signal $x_a(t)$. This distortion of the original analog signal was caused by the aliasing effect, due to the low sampling rate used.

****

Solve the exercise problems related to the topics discussed in the lecture.

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering

Segment-2
Discrete-Time Signals and Systems

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ Representation of discrete time signals.
- ✓ Some elementary discrete time signals
- ✓ Classification of discrete time (DT) signals.
- ✓ Manipulation of DT signals
- ✓ Classification of Discrete Time System
- ✓ Convolution sum
- ✓ Correlation

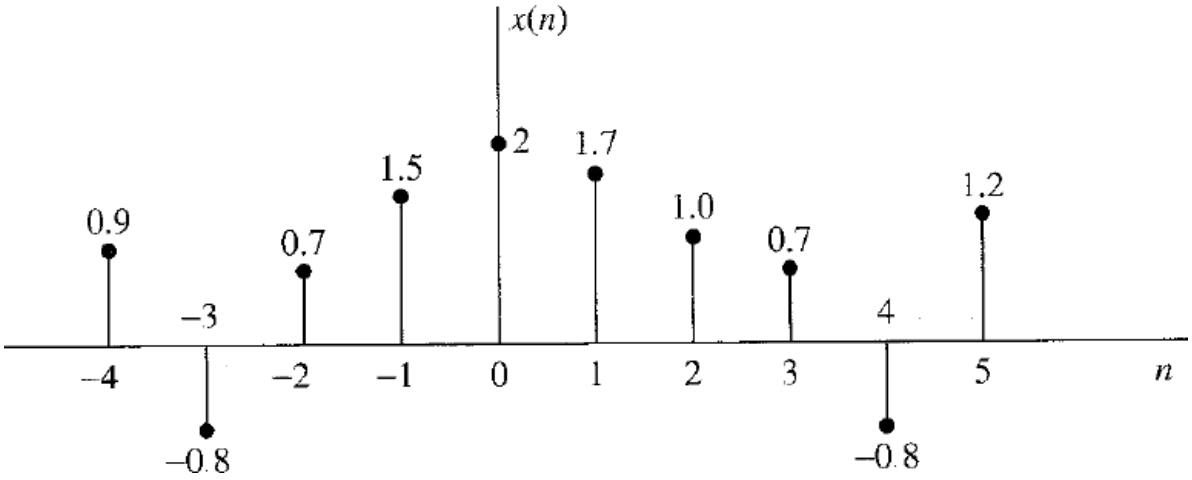
Text Book:

Digital Signal Processing (4th Edition), John G. Proakis, Dimitris K Manolakis

Week 4
Slide 49-54

Representation of Discrete-Time Signals

1. Graphical Representation



2. Functional Representation

$$x(n) = \begin{cases} 1, & \text{for } n = 1, 3 \\ 4, & \text{for } n = 2 \\ 0, & \text{elsewhere} \end{cases}$$

3.Tabular Representation

n	...	−2	−1	0	1	2	3	4	5	...
$x(n)$	...	0	0	0	1	4	1	0	0	...

Representation of Discrete-Time Signals (Cont.)

4. Sequence Representation

An infinite-duration signal or sequence with the time origin (n=0) indicated by the symbol ↑ is represented as

$$x(n) = \{\ldots 0, \underset{\uparrow}{0}, 1, 4, 1, 0, 0, \ldots\}$$

$$x(n) = \{\underset{\uparrow}{0}, 1, 4, 1, 0, 0, \ldots\}$$

A finite duration sequence can be represented as

$$x(n) = \{3, -1, \underset{\uparrow}{-2}, 5, 0, 4, -1\}$$

Whereas a finite-duration sequence that satisfies the condition $x(n) = 0$ for n<0 can be represented as

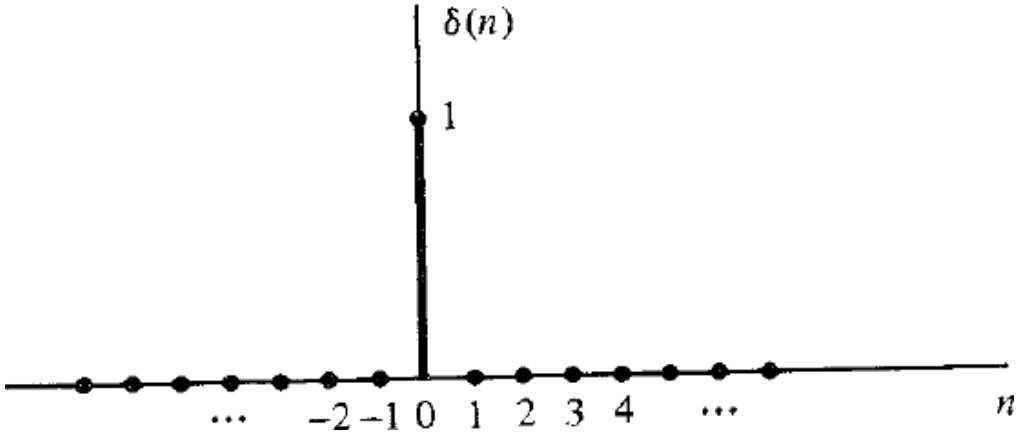
$$x(n) = \{\underset{\uparrow}{0}, 1, 4, 1\}$$

Some Elementary Discrete –Time Signals

1. Unit Sample Sequence

The unit sample sequence is denoted as $\delta(n)$ and is defined as

$$\delta(n) \equiv \begin{cases} 1, & \text{for } n = 0 \\ 0, & \text{for } n \neq 0 \end{cases}$$

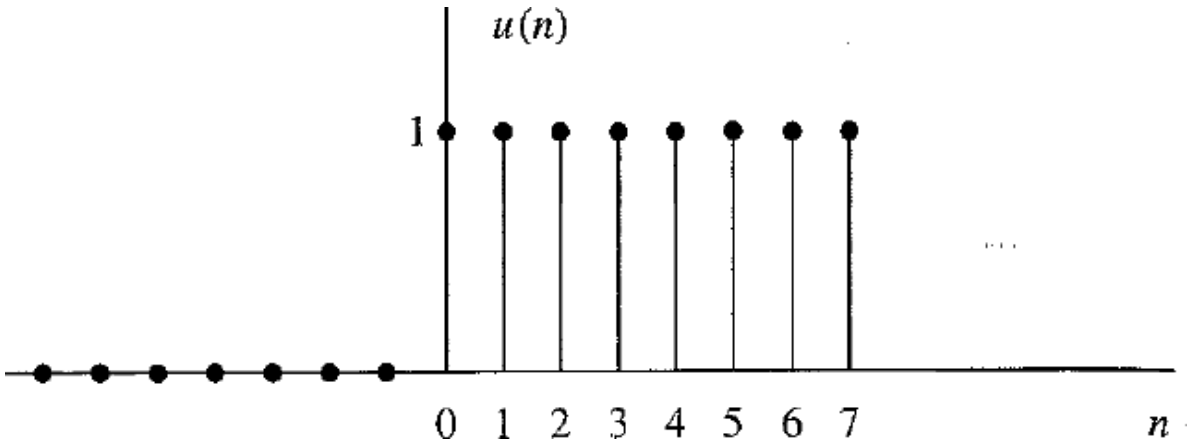


Some Elementary Discrete –Time Signals (cont.)

2. Unit Step signal

The unit step signal is denoted as u(n) and is defined as

$$u(n) \equiv \begin{cases} 1, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases}$$

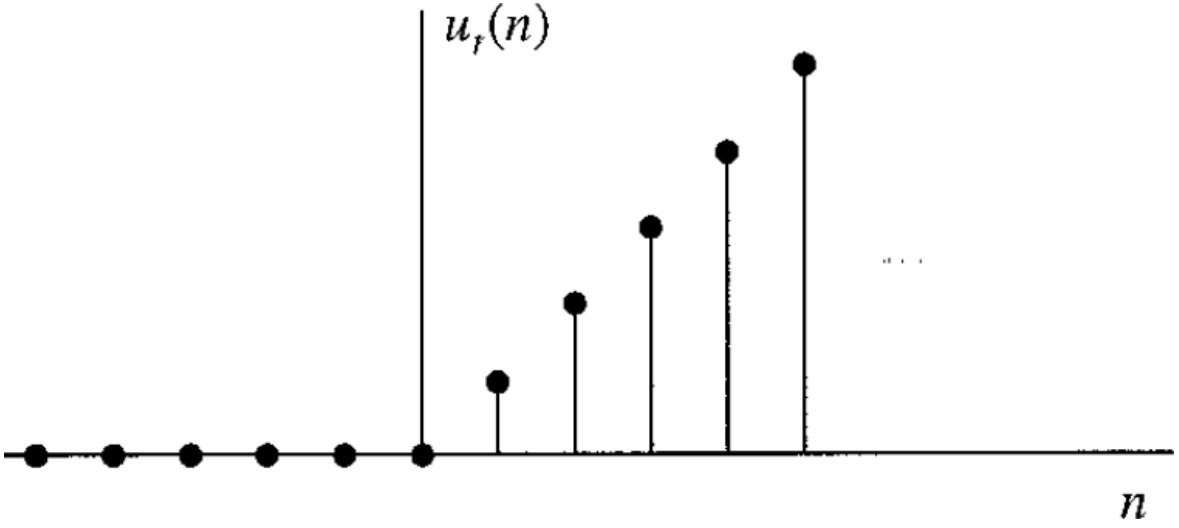


Some Elementary Discrete –Time Signals (cont.)

3. Unit ramp signal

The unit ramp signal is denoted as $u_r(n)$ and is defined as

$$u_r(n) \equiv \begin{cases} n, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases}$$



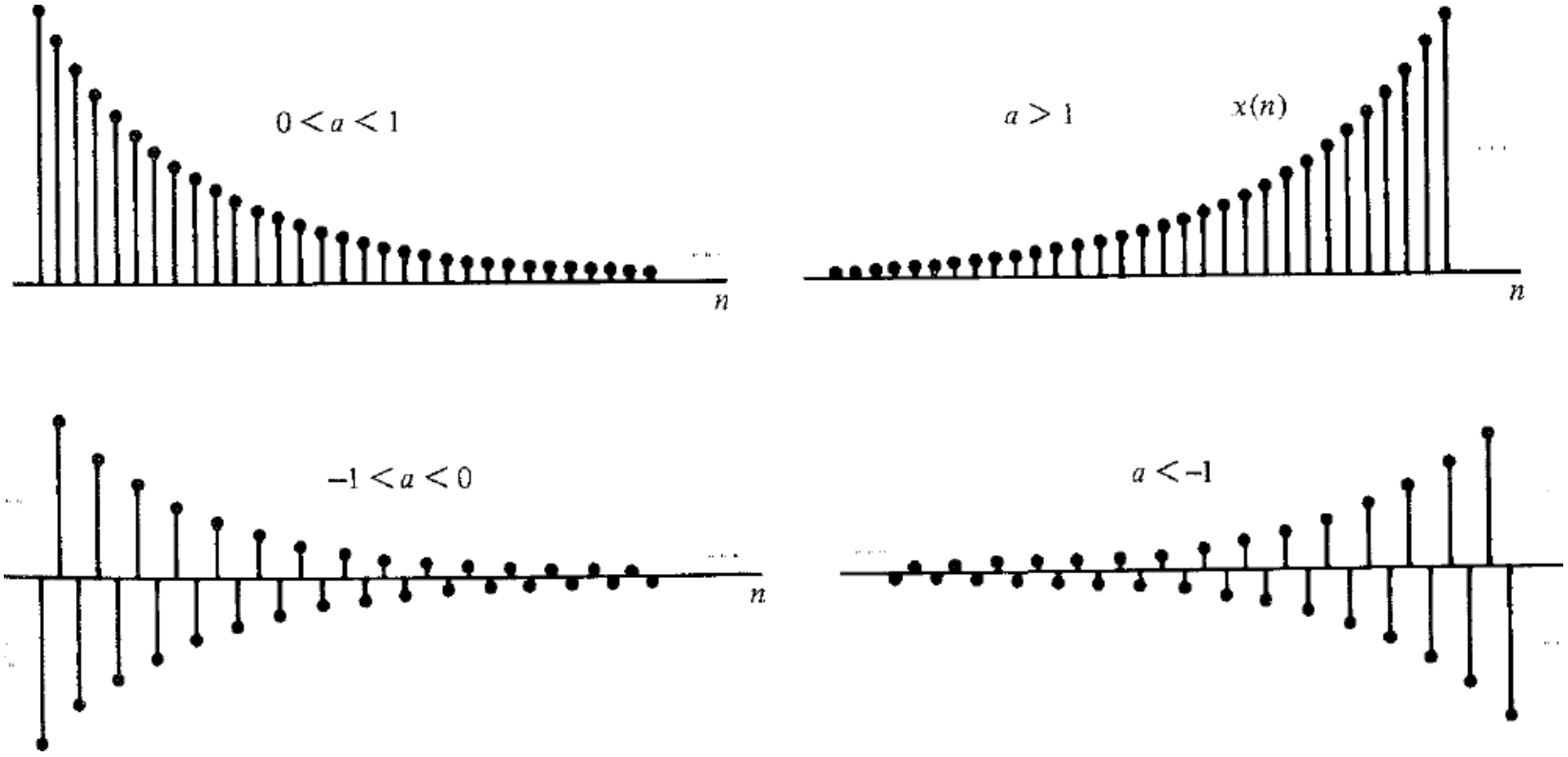
Some Elementary Discrete –Time Signals (cont.)

4. Exponential signal

The exponential signal is a sequence of the form

$$x(n) = a^n \qquad \text{for all } n$$

If the parameter a is real, then x(n) is real.



Class Test Next Week

Syllabus: Slide 1-54

Week 5
Slide 56-64

Classification of Discrete –Time Signals

Energy Signal and Power Signal

The energy E of a signal x(n) defined as $$E \equiv \sum_{n=-\infty}^{\infty} |x(n)|^2$$

The energy of a signal can be finite or infinite. If E is finite (i.e. $0<E<\infty$), then x(n) is called an **energy signal**.

Many signals that posses infinite energy have a finite average power.The average power of a discrete-time signal x(n) is defined as

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

If we define the signal energy of x(n) over the finite interval $-N \le n \le N$ as $$E_N \equiv \sum_{n=-N}^{N} |x(n)|^2$$

Then we can express energy E as $$E \equiv \lim_{N\to\infty} E_N$$

So the average power of the signal x(n) as $$P \equiv \lim_{N\to\infty} \frac{1}{2N+1} E_N$$

Clearly, if E is finite, P=0. On the other hand, if E is infinite, the average power P may be either finite or infinite. If P is finite and nonzero, the signal is called a **power signal**.

Classification of Discrete –Time Signals (Cont.)

Energy Signal and Power Signal (Cont.)

Determine the power and energy of the following sequence.

$$x\ n\ = \alpha^n u(\ n)\ , where\ 0 < \alpha < 1$$

$$\sum_{n=-\infty}^{\infty} [x(n)]^2 = \sum_{n=-\infty}^{\infty} [a^n \mathrm{u}(n)]^2 = \sum_{n=0}^{\infty} [a^n]^2 = \sum_{n=0}^{\infty} [a^2]^n = \frac{1}{1-a^2}$$

Formula $$\left[\sum_{k=0}^{\infty} r^k = \frac{1}{1-r}, |r| < 1 \right]$$

Here, for $0 < \alpha < 1$ we can say $0 < E < \infty$

So, $x(n) = \alpha^n u(n)$ is an energy signal for $0 < \alpha < 1$

Classification of Discrete –Time Signals (cont.)

Determine the power and energy of the unit step sequence.

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=0}^{N} u^2(n)$$

$$= \lim_{N\to\infty} \frac{N+1}{2N+1} = \lim_{N\to\infty} \frac{1+1/N}{2+1/N} = \frac{1}{2}$$

Consequently, the unit step sequence is a power signal. Its energy is infinite.

- A signal can be an energy signal, a power signal or neither type. Unit ramp sequence is neither a power signal nor an energy signal.
- A signal can not be both an energy signal and a power signal.

Problem: Check Whether the following signals are energy or power signals

(i) $\delta(n)$ (ii) $x(n) = (\frac{1}{3})^n u(n)$ (iii) $x(n) = \sin(\frac{\pi}{4}n)$ (iv) $x(n)=e^{j\frac{\pi}{4}n}$

Classification of Discrete –Time Signals (cont.)

Periodic Signal and aperiodic signal

A signal x(n) is periodic with period N (N>0) if and only if

$$x(n+N) = x(n) \text{ for all } n$$

The smallest value of N that satisfies above relation is called the fundamental period. If there is no value of N that satisfies the above relation, the signal is called nonperiodic or aperiodic.

The sinusoidal signal in the form $x(n) = A\sin(2\pi f_0 n)$ is periodic when $f_0$is a rational number, that is, if $f_0$is expressed as

$$f_0 = \frac{k}{N}, \textit{where k and N are integers}$$

Periodic Signals are Power Signals

The energy of the periodic signal x(n) for $-\infty \leq n \leq \infty$ is infinite. On the other hand , the average power of the periodic signal is finite and is equal to the average power over a single period.Thus if x(n) is periodic signal with fundamental period N and takes on finite values , its power is given by

$$P = \frac{1}{N}\sum_{n=0}^{N-1} |x(n)|^2$$

Consequently the periodic signals are power signal.

Causality of Signal

Causal Signal

A continuous time signal $x(t)$ is called causal signal if the signal $x(t) = 0$ for $t < 0$. Therefore, a causal signal does not exist for negative time. The unit step signal u(t) is an example of causal signal as shown in Figure-1. Similarly, a discrete time sequence x(n) is called the causal sequence if the sequence x(n) = 0 for n < 0.

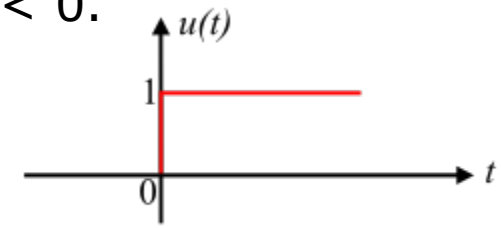


Figure-1

Anti-Causal Signal

A continuous-time signal x(t) is called the anti-causal signal if x(t) = 0 for t > 0. Hence, an anti-causal signal does not exist for positive time. The time reversed unit step signal u(-t) is an example of anti-causal signal (see Figure-2).

Similarly, a discrete time sequence x(n) is said to be anti-causal sequence if the sequence x(n) = 0 for $t > 0$.

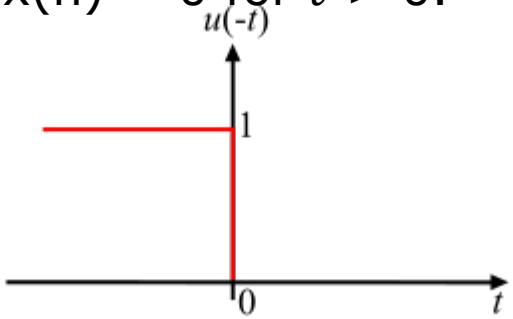


Figure-2

Causality of Signal (cont.)

Non-Causal Signal

A signal which is not causal is called the non-causal signal. Hence, by the definition, a signal that exists for positive as well as negative time is neither causal nor anti-causal, it is non-causal signal. The sine and cosine signals are examples of non-causal signal (see Figure-3).

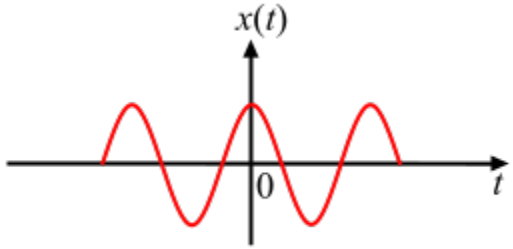


Figure-3

Classification of Discrete –Time Signals (cont.)

Symmetric (even) signal and Antisymmetric (odd) Signal

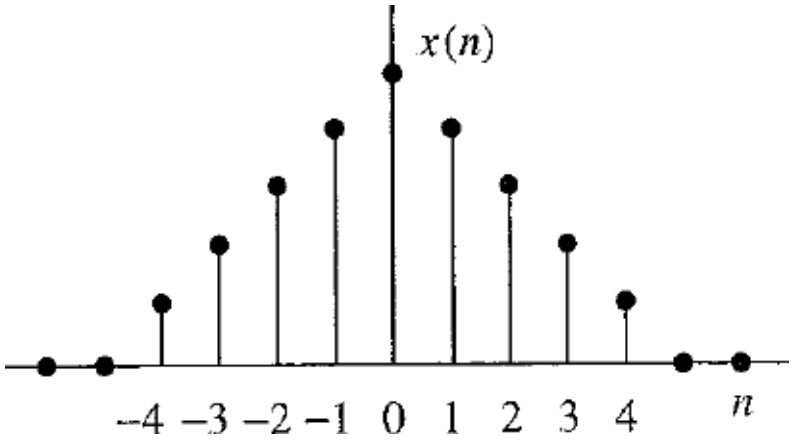
A real valued signal x(n) is called symmetric (even) if

$$x(-n)=x(n) \quad \text{for all } n.$$

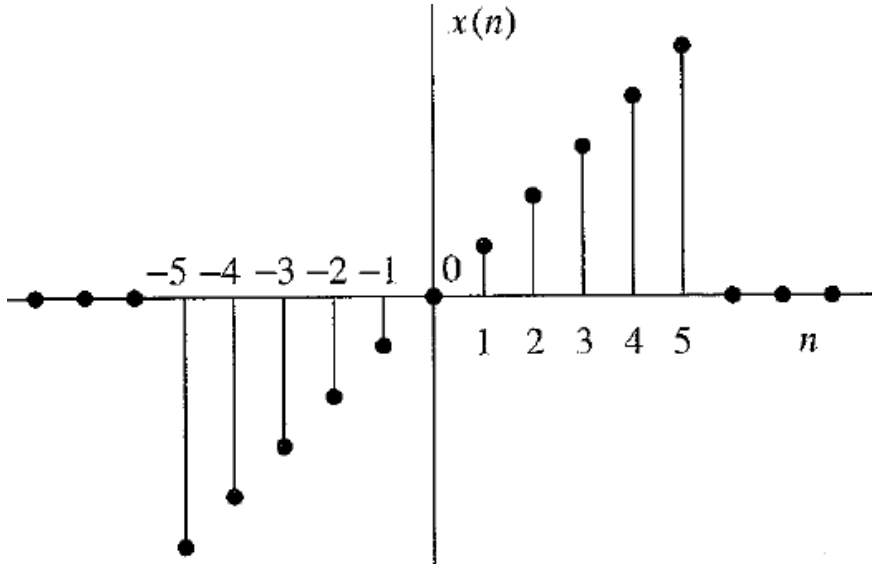
On the other hand, a signal x(n) is called antisymmetric (odd) if

$$x(-n)=-x(n) \quad \text{for all } n$$

If x(n) is odd, x(0)=0



Even Signal



Odd Signal

Classification of Discrete –Time Signals (cont.)

Symmetric (even) signal and Antisymmetric (odd) Signal

Many signal are neither even nor odd. Any arbitrary signal can be expressed as the sum of two signal components, one of which is even and the other odd.

The even signal components is expressed as

$$x_e(n) = \frac{1}{2}[x(n) + x(-n)]$$

The odd signal components is expressed as

$$x_o(n) = \frac{1}{2}[x(n) - x(-n)]$$

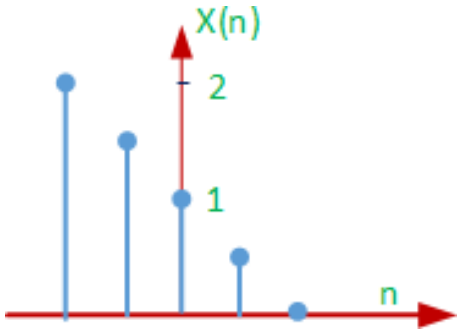
The signal x(n) is expressed as-

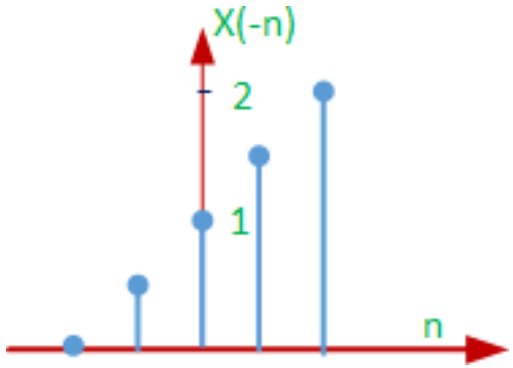
$$x(n) = x_e(n) + x_o(n)$$

Classification of Discrete –Time Signals (cont.)

Symmetric (even) signal and Antisymmetric (odd) Signal

Find the even and odd components of following signal x(n).



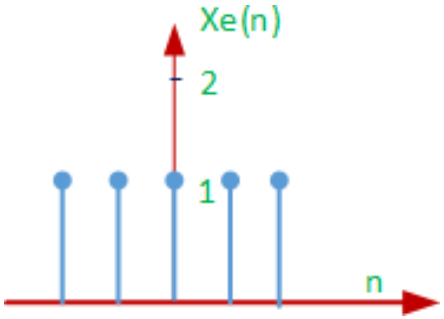


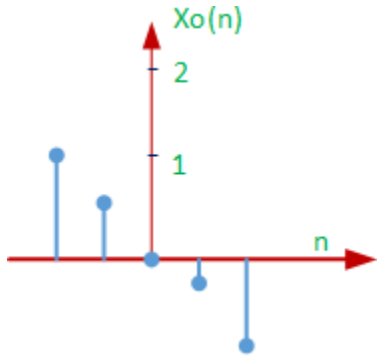
$x(n)=\{2,1.5,1,0.5,0\}$ (↑ at 1)

$x(-n)=\{0,0.5,1,1.5,2\}$ (↑ at 1)

$$x_e(n)=\frac{1}{2}[x(n)+x(-n)]=\{1,1,1,1,1\}$$ (↑ at 1)

$$x_e(n)=\frac{1}{2}[x(n)-x(-n)]=\{1,0.5,0,-0.5,-1\}$$ (↑ at 0)





Task: Find the even and odd parts of the following signals:

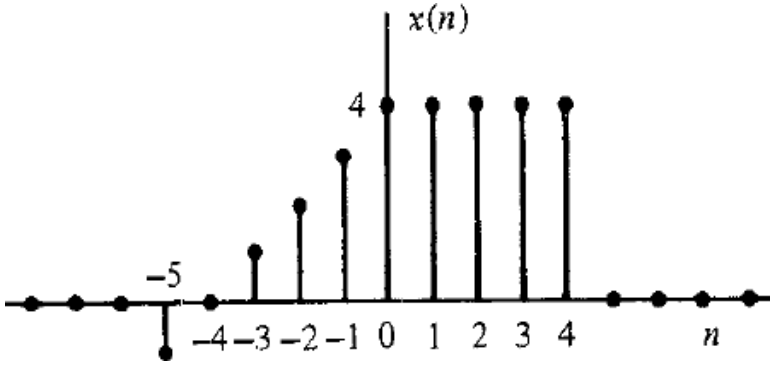
(a) x(n)=u(n) (b) $x(n)=\alpha^n u(n)$

Week 6
Slide 66-71

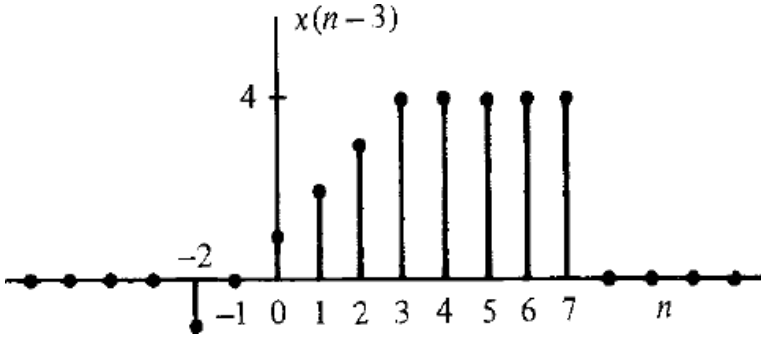
Simple manipulations of Discrete-Time Signals

Time Shifting

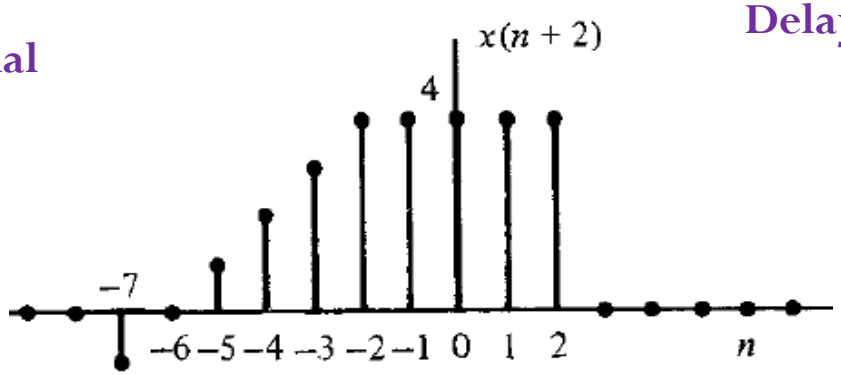
- A signal x(n) may be shifted in time by replacing the independent variable n by (n-k), where k is integer.
- If k is a positive integer, the time shift results in a delay of the signal by k units of time.
- If k is negative integer, the time shift results in an advance of the signal by |k| units in time.



Original Signal



Delay/Right Shift

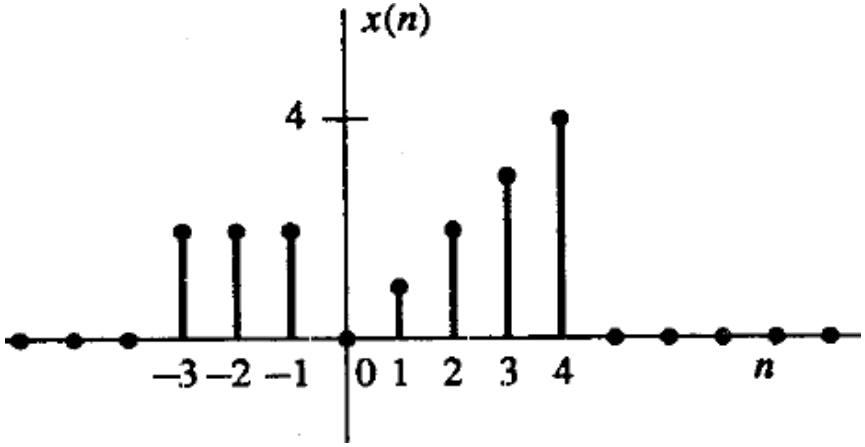


Advances/Left Shift

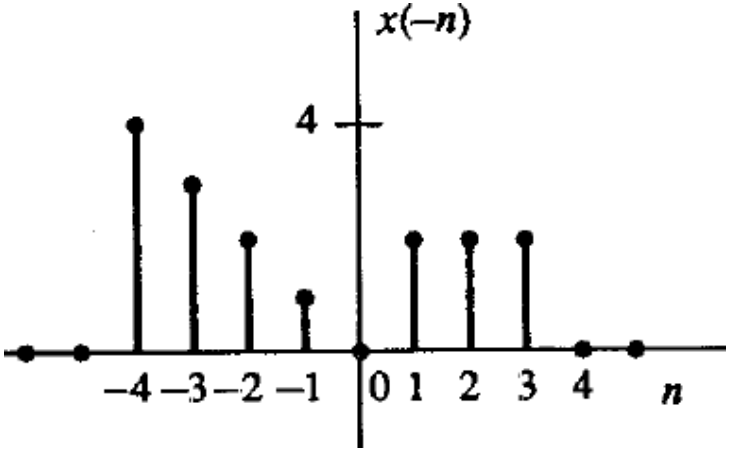
Simple manipulations of Discrete-Time Signals (Cont.)

Folding

Another useful modification of the time base is to replace the independent variable n by –n. The result of this operation is **a folding or a reflection of the signal** about the time origin n=0



Original Signal



Folded signal

Simple manipulations of Discrete-Time Signals (Cont.)

Shifting and Folding

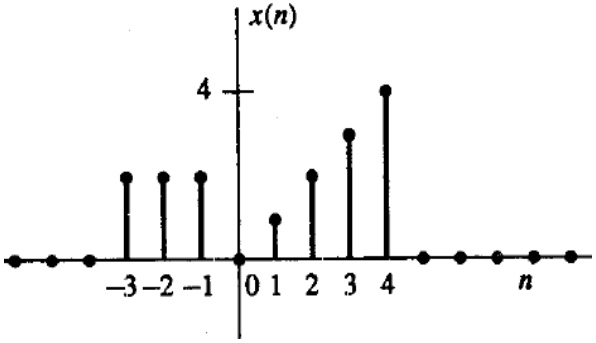
It is important to note that the operations of folding and time delaying (or advancing) a signal are not commutative. If we denote the time-delay operation by TD and folding operation by FD, we can write-

$$\mathrm{TD}_k[x(n)] = x(n-k), \qquad k > 0$$

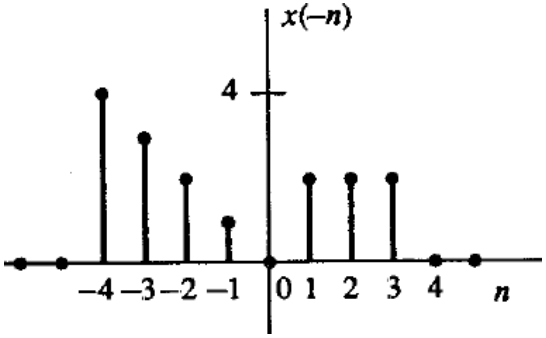
$$\mathrm{FD}[x(n)] = x(-n)$$

$$\mathrm{TD}_k\{\mathrm{FD}[x(n)]\} = \mathrm{TD}_k[x(-n)] = x(-n+k)$$

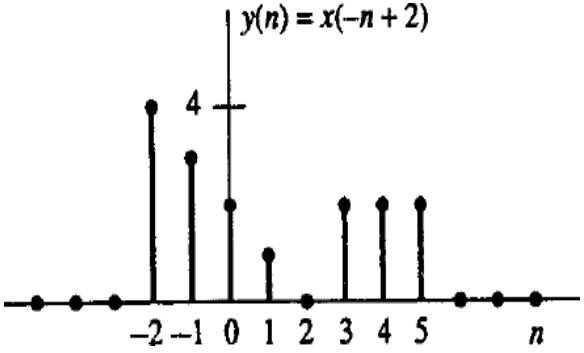
$$\mathrm{FD}\{\mathrm{TD}_k[x(n)]\} = \mathrm{FD}[x(n-k)] = x(-n-k)$$



Original Signal



Folded signal



Folded and shifted signal

Simple manipulations of Discrete-Time Signals (Cont.)

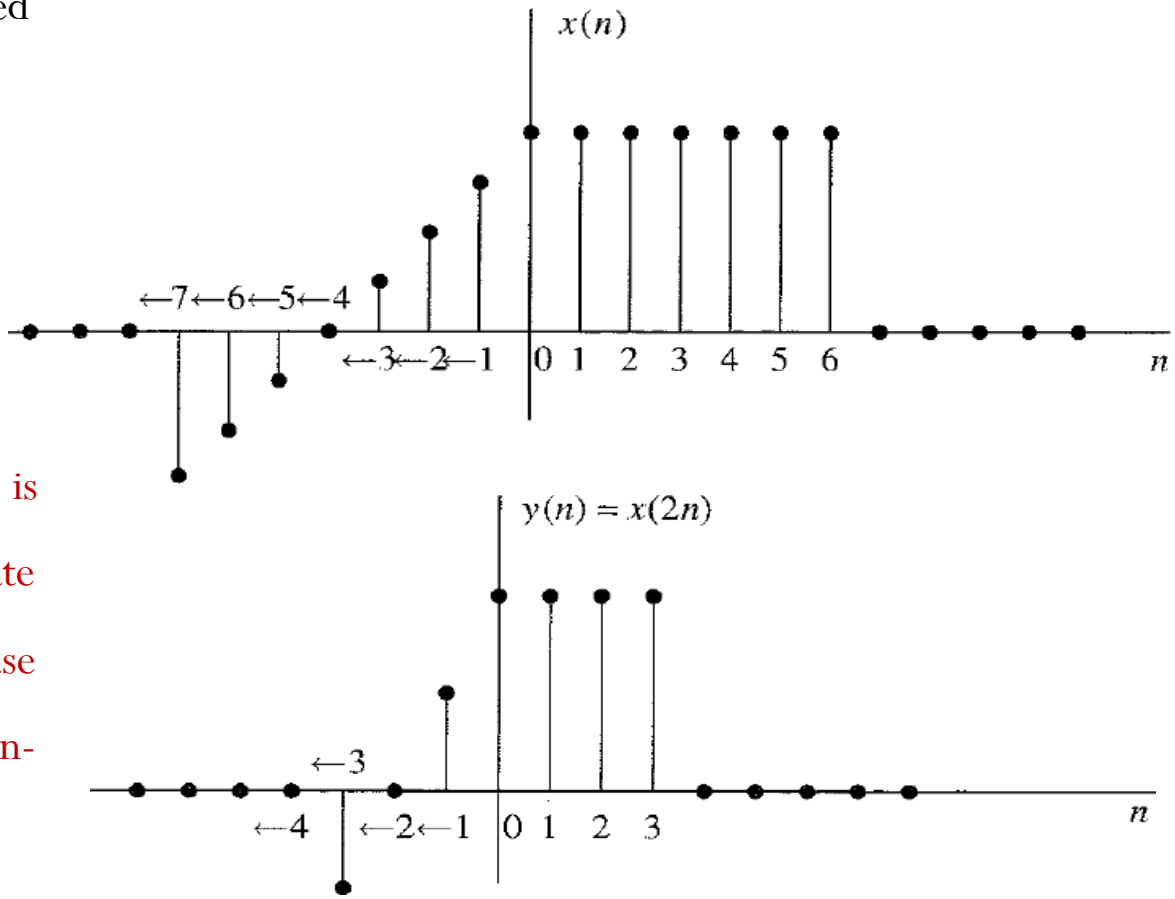
Time Scaling or Down sampling

Another modification of independent variable involves replacing n by μn, where μ is an integer. We refer to this time-base modification as time scaling or down-sampling.

If the signal x(n) was originally obtained by sampling an analog signal $x_a(t)$, then $x(n) = x_a(nT)$, where T is the sampling interval.

Now, $y(n) = x(2n) = x_a(2Tn)$.

Hence the time-scaling operation is equivalent to changing the sampling rate from 1/T to 1/μT, that is, to decrease the rate by a factor μ. This is a down-sampling operation.



Simple manipulations of Discrete-Time Signals (Cont.)

Addition, Multiplication and Scaling of sequence

Amplitude Scaling of a signal by a constant A is accomplished by multiplying the value of every signal sample A. Consequently, we obtain

$$y(n) = Ax(n), \qquad -\infty < n < \infty$$

The addition (sum) of two signals $x_1(n)$ and $x_2(n)$ is a signal $y(n)$, whose value at any instant is equal to the sum of the values of these two signals at that instant, that is,

$$y(n) = x_1(n) + x_2(n), \qquad -\infty < n < \infty$$

The product of two signals is similarly defined on a sample-to-sample basis as

$$y(n) = x_1(n)x_2(n), \qquad -\infty < n < \infty$$

Input-Output Description of System

Determine the response of the following sytems to the input signal

$$x(n) = \begin{cases} |n|, & -3 \le n \le 3 \\ 0, & \text{otherwise} \end{cases}$$

(a) $y(n) = x(n)$ (identity system)

(b) $y(n) = x(n-1)$ (unit delay system)

(c) $y(n) = x(n+1)$ (unit advance system)

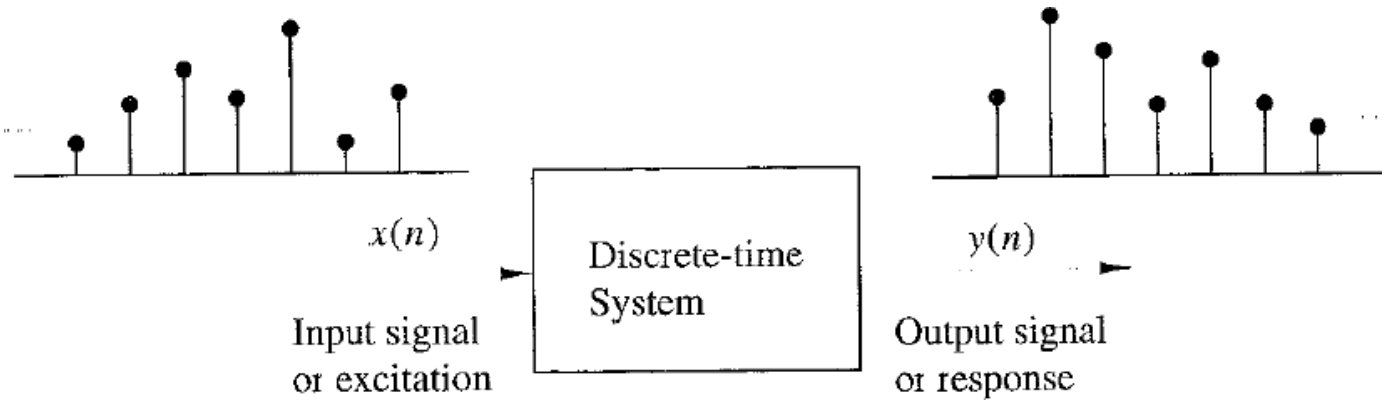
(d) $y(n) = \frac{1}{3}[x(n+1) + x(n) + x(n-1)]$ (moving average filter)

(e) $y(n) = \text{median}\{x(n+1), x(n), x(n-1)\}$ (median filter)

(f) $y(n) = \sum_{k=-\infty}^{n} x(k) = x(n) + x(n-1) + x(n-2) + \cdots$ (accumulator)

Week 7
Slide 73-89

Block diagram representation of Discrete Time System



$$y(n) \equiv \tau[x(n)]$$

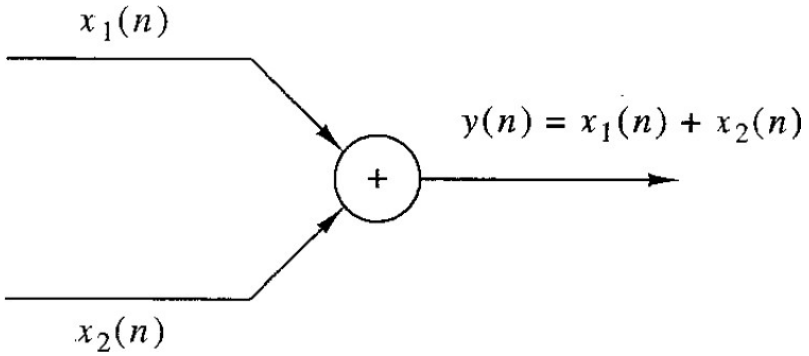
$$y(n) \overset{\tau}{} x(n)$$

The above expressions represent that x(n) is transformed by the system into a signal y(n) where the symbol τ denotes the transformation (also called an operator) or processing performed by the system on x(n) to produce y(n).

Block diagram representation of DT System

An adder System

An adder system that performs the addition of two signal sequences to form another (the sum) sequence y(n). Note that it is not necessary to store either one of the sequence in order to perform the addition i.e. the addition operation is memoryless.



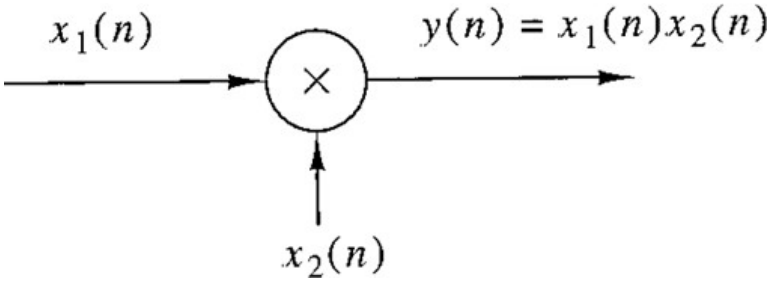
Constant Multiplier

This operation apply a scale factor on the input x(n). It is also memoryless operation.

Block diagram representation of DT System (Cont.)

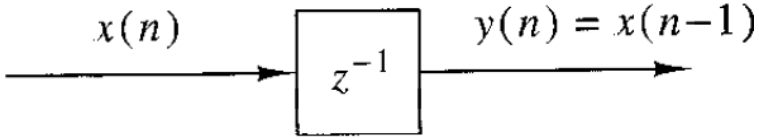
A signal Multiplier

It is also Memoryless system that multiplies two signal sequences to form another and is represented by following block-



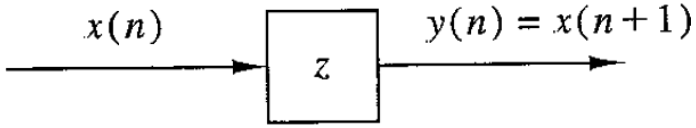
Unit delay element

In unit delay system if input signal is x(n), the output is x(n-1). In fact, the sample x(n-1) is stored in memory at time n-1 and it is recalled from memory at time n to form y(n)=x(n-1). Thus this basic building block requires memory.



Unit advance element

A unit advance moves the input x(n) ahead by one sample in time to yield x(n=1). Such advances impossible in real time, since, it involves looking into future of the signal. On the other hand, if we store the signal in the memory of the computer, we can recall any sample at any time.

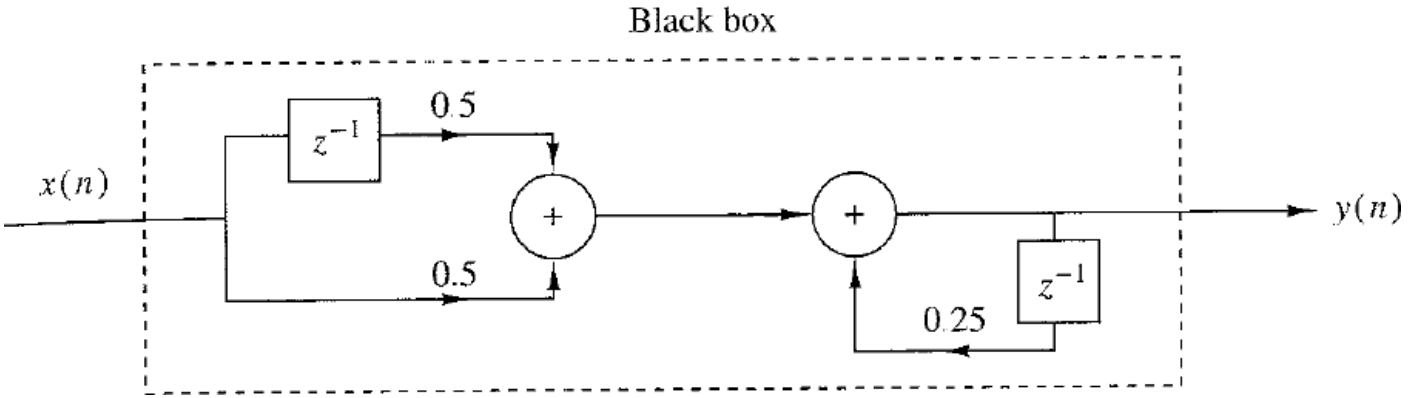


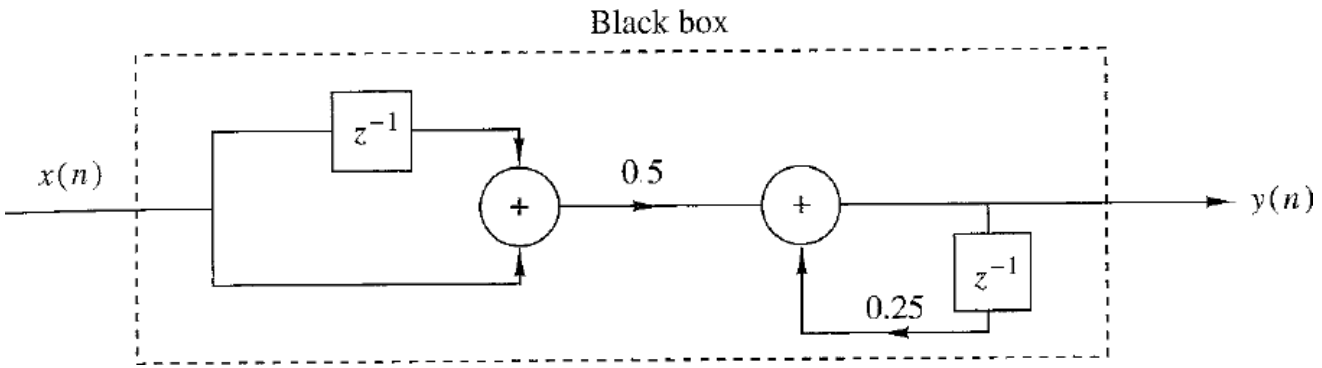
Block diagram representation of DT System (Cont.)

Problem: Using basic building blocks sketch the block diagram representation of the discrete-time system described by the input-output relation

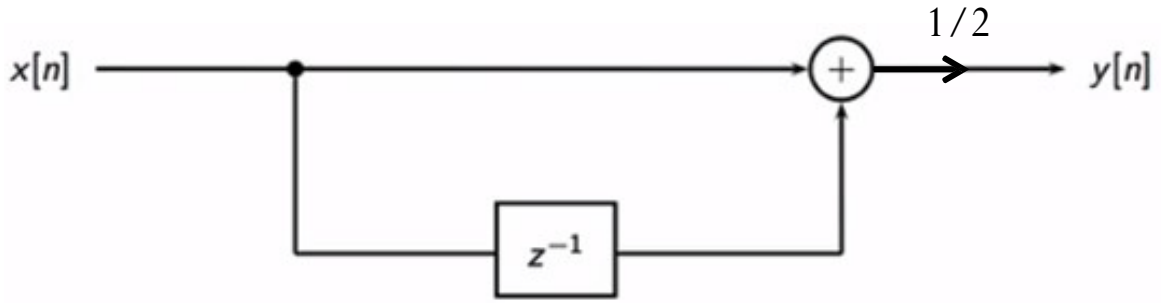
$$y(n) = \frac{1}{4}y(n-1) + \frac{1}{2}x(n) + \frac{1}{2}x(n-1)$$

Where x(n) is the input and y(n) is the output of the system.



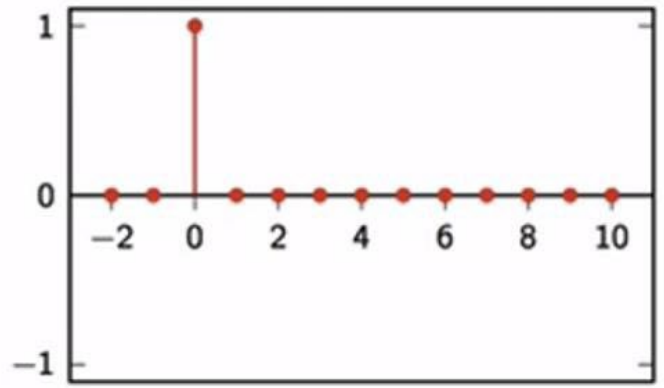


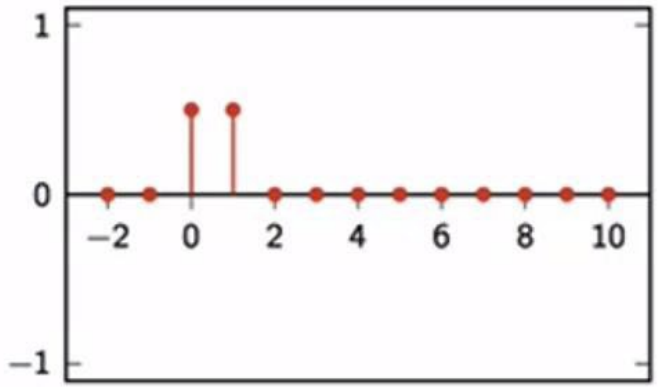
Block diagram representation of DT System (Cont.)



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

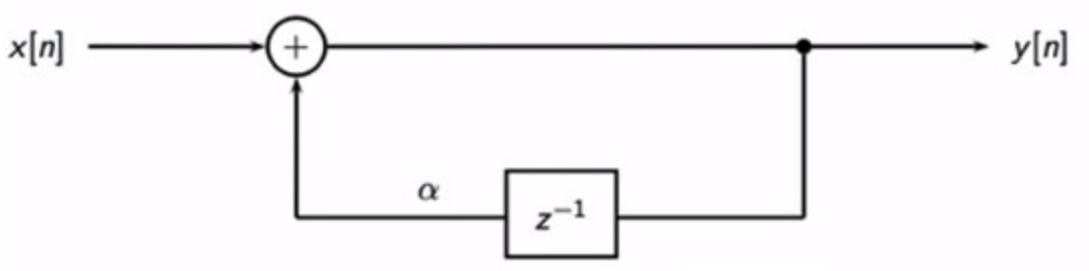
$x[n] = \delta[n]$





Block diagram representation of DT System (Cont.)

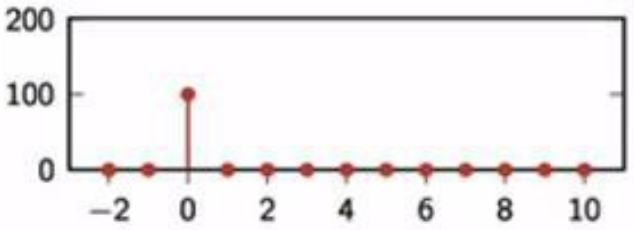
Problem: Find the output of the following system, if $\alpha = 1.05$ and input $x(n) = 100\,\delta(n)$

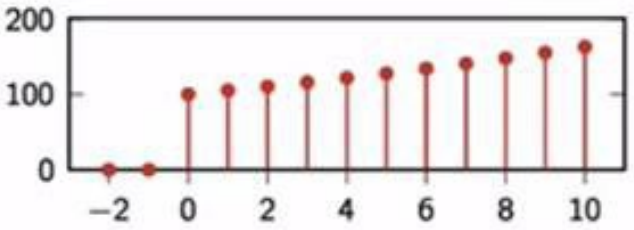


$$y(n) = x(n) + \alpha y(n-1)$$

$x[n] = 100\,\delta[n]$

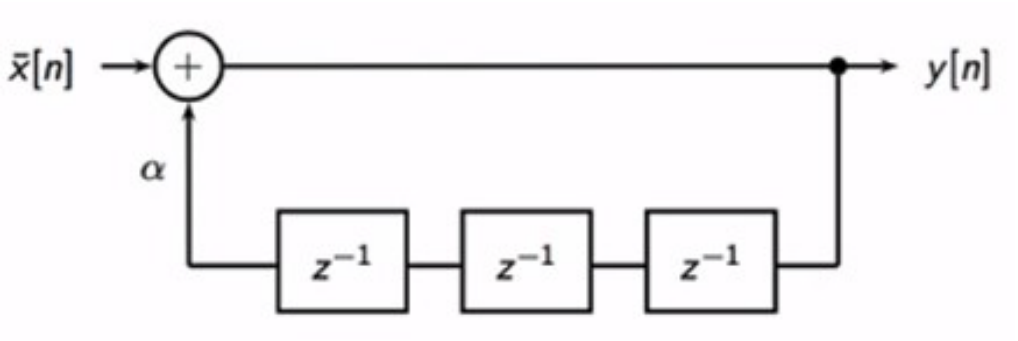
- $y[0] = 100$
- $y[1] = 105$
- $y[2] = 110.25$, $y[3] = 115.7625$ etc.
- In general: $y[n] = (1.05)^n 100\,u[n]$



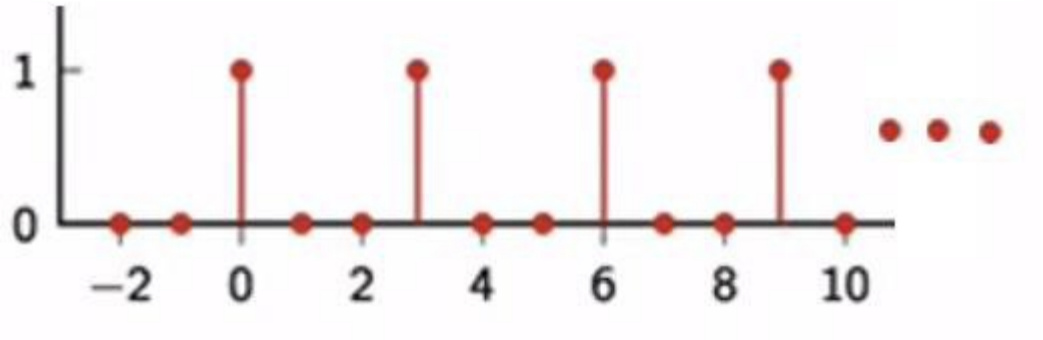


Block diagram representation of DT System (Cont.)

Problem: Find the output of the following system, if $\alpha = 1$ and input $x(n) = \delta(n)$. Assume initially all input, output and memory blocks are 0.



$$y(n) = x(n) + \alpha y(n-3)$$



What happens if $\alpha = 0.9$??

Classification of DT Systems

Static vs dynamic System

- ✓ A discrete-time system is called static or memoryless if its output at any instant n depends at most on the input samples at the same time, but not on past or future samples of the input. In other cases, the system is said to be dynamic or to have memory.
- ✓ If the output of a system at time n is completely determined by the input samples in the interval from n-N to n ($N \geq 0$), the system is said to have memory of duration N.
- ✓ If N=0, the system is static. If 0<N<∞, the system is said to have finite memory, whereas if N= ∞, the system is said to have infinite memory.

Example of Static Memory

$$y(n) = ax(n)$$

$$y(n) = nx(n) + bx^3(n)$$

Example of Dynamic Memory

$$y(n) = x(n) + 3x(n-1)$$ Finite Memory

$$y(n) = \sum_{k=0}^{n} x(n-k)$$ Finite Memory

$$y(n) = \sum_{k=0}^{\infty} x(n-k)$$ Infinite Memory

Classification of DT Systems (Cont.)

Time variant vs time invariant System

A system is called time-invariant if its input-output characteristics do not change with time. In other words, a relaxed system τ is time invariant or shift invariant if and only if

$$x(n) \xrightarrow{\mathcal{T}} y(n)$$

implies that

$$x(n-k) \xrightarrow{\mathcal{T}} y(n-k)$$

for every input signal x(n) and every time shift k. Otherwise the system is said to be time variant.

Identifying a system as Time variant or time invariant

Step-1: Excite the system with an arbitrary input sequence x(n), which produces an output denoted as y(n).

Step-2: Delay the input sequence by some amount k and recompute the output which is written as

$$y(n,k) = \mathcal{T}[x(n-k)]$$

Step-3: Delay output y(n) obtained in step-1 by some amount k to find y(n-k).

Now if y(n,k)=y(n-k), for all possible values of k, the system is time invariant.

But if the output $y(n,k) \neq y(n-k)$, even for one value of k, the system is time variant.

Classification of DT Systems (Cont.)

Time variant vs time invariant System: Determine if the following systems are time invariant or time variant. (Example 2.2.4, Proakis)

The system is described by

$$y(n) = \mathcal{T}[x(n)] = x(n) - x(n-1)$$

Input delayed by k unit and applied to the system results

$$y(n,k) = x(n-k) - x(n-k-1)$$

If y(n) delayed by k unit, we get

$$y(n-k) = x(n-k) - x(n-k-1)$$

System follows that y(n,k)=y(n-k).

Therefore, the system is time invariant.

The input-output equation for this system is

$$y(n) = \mathcal{T}[x(n)] = nx(n)$$

The response of this system to x(n-k) is

$$y(n,k) = nx(n-k)$$

If y(n) delayed by k unit, we get

$$\begin{aligned} y(n-k) &= (n-k)x(n-k) \\ &= nx(n-k) - kx(n-k) \end{aligned}$$

The system is time variant, since

$$y(n,k) \neq y(n-k),$$

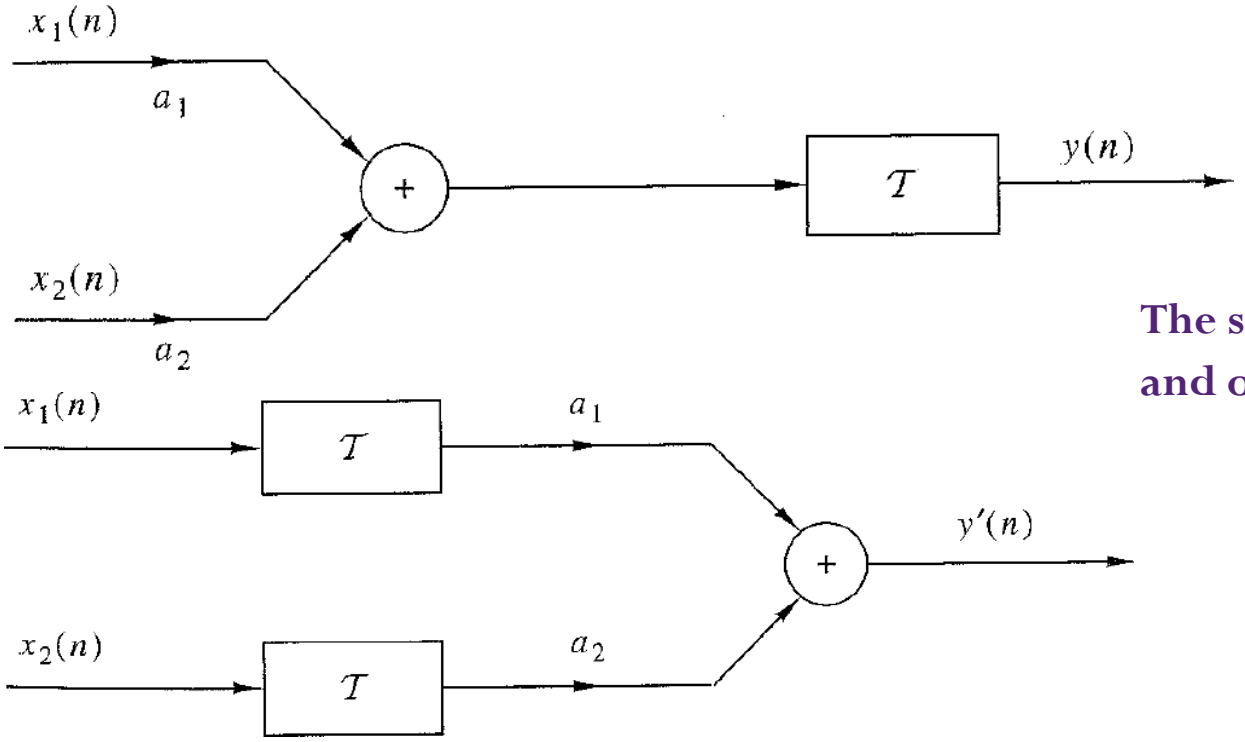
Classification of DT Systems (Cont.)

Linear Vs nonlinear System

A linear system is one that satisfies the superposition principle. Mathematically, a system is liner if and only if

$$\mathcal{T}[a_1 x_1(n) + a_2 x_2(n)] = a_1 \mathcal{T}[x_1(n)] + a_2 \mathcal{T}[x_2(n)]$$

For any arbitrary sequences $x_1(n)$ and $x_2(n)$, and any arbitrary constant a_1 and a_2



The system $\mathcal{T}$ is linear if and only if $y(n) = y'(n)$

See example 2.2.5 (Proakis)

Linearity of a System

Problem

Determine whether the following system described by the equation is linear or nonlinear.

$$\frac{dy(t)}{dt} + y(t) = x(t)$$

Using the superposition theorem, we can prove that the system is linear.

For input $x_1(t)$, the output is

$$\frac{dy_1(t)}{dt} + y_1(t) = x_1(t) \quad ..(1)$$

For input $x_2(t)$, the output is

$$\frac{dy_2(t)}{dt} + y_2(t) = x_2(t) \; ..(2)$$

$\text{Eq}^n(1) * a_1 + \text{Eq}^n(2) * a_2$

$$a_1\frac{dy_1(t)}{dt} + a_2\frac{dy_2(t)}{dt} + a_1y_1(t) + a_2y_2(t) = a_1x_1(t) + a_2x_2(t) \; ..(3)$$

Now Putting $a_1 = a_2 = 1$, $x_1(t) + x_2(t) = x(t)$ & $y_1(t) + y_2(t) = y(t)$ in $\text{Eq}^n(3)$ we get,

$$\frac{dy(t)}{dt} + y(t) = x(t)$$

Which is same as the original Equation. So, the System is **Linear**.

Linearity of a System

Problem

Determine whether the following system described by the equation is linear or nonlinear.

$$\frac{dy(t)}{dt} + y(t) + 2 = x(t)$$

Using the superposition theorem, we can prove that the system is linear.

For input $x_1(t)$, the output is

$$\frac{dy_1(t)}{dt} + y_1(t) + 2 = x_1(t) \quad ..(1)$$

For input $x_2(t)$, the output is

$$\frac{dy_2(t)}{dt} + y_2(t) + 2 = x_2(t) \quad ..(2)$$

$\text{Eq}^n(1) * a_1 + \text{Eq}^n(2) * a_2$

$$a_1\frac{dy_1(t)}{dt} + a_2\frac{dy_2(t)}{dt} + a_1y_1(t) + a_2y_2(t) + 2a_1 + 2a_2 = a_1x_1(t) + a_2x_2(t) \quad ..(3)$$

Now Putting $a_1 = a_2 = 1$, $x_1(t) + x_2(t) = x(t)$ & $y_1(t) + y_2(t) = y(t)$ in $\text{Eq}^n(3)$ we get,

$$\frac{dy(t)}{dt} + y(t) + 4 = x(t)$$

Which is not the same as the original Equation. So, the System is **Non-Linear**.

Classification of DT Systems (Cont.)

Causal vs Noncausal System

A system is said to be causal if the output of the system at any time n [i.e., y(n)] depends only on present and past inputs [i.e., x(n), x(n-1), x(n-2), ….] but does not depend on future inputs [i.e., x(n+1), x(n+2), ….]

Mathematically, $y(n) = F[x(n), x(n-1), x(n-2), \ldots]$

If a system does not satisfy this definition, it is called noncausal. Such a system has an output that depends not only on present and past inputs but also in future inputs.

Note:

It is apparent that in real-time signal processing applications we cannot observe future values of the signal, and hence a noncausal system is physically unrealizable (i.e., it can not be implemented). On the other hand if the system is recored so that the processing is done by off-line (nonreal time), it is possible to implement a non causal system.

Classification of DT Systems (Cont.)

Stable vs unstable System

An arbitrary relaxed system is said to be bounded input-bounded output (BIBO) stable if and only if every bounded input produces a bounded output at each and every instant.

The condition that the input sequence x(n)and the output sequence y(n) are bounded is translated mathematically to mean that there exist some finite numbers, say M_x and M_y, such that

$$|x(n)| \leq M_x < \infty, \qquad |y(n)| \leq M_y < \infty \qquad \text{for all n.}$$

If, for some bounded input sequence x(n), the output is unbounded (infinite), the system is classified as unstable.

Classification of DT Systems (Cont.)

Stable Vs unstable System (Cont.)

Example: Check whether the following system is stable or unstable.

(i) $y(n) = x^2(n)$ **(ii)** $y(n) = n\,x(n)$ **(iii)** $y(n) = \cos(n)x(n)$ **(iv)** $y(n) = \frac{x(n)}{\sin(n)}$

(i) $y(n) = x^2(n)$

For any bounded input x(n)= $B_x < \infty$,

$$y(n) = (B_x)^2 < \infty$$

At each and every instant (for any value of n), the output is bounded.

Therefore, the system is BIBO stable.

(ii) $y(n) = n\,x(n)$

For any bounded input x(n)= $B_x < \infty$,

$$y(n) = n(B_x)^2$$

When $n = \infty$, $y(n) = \infty$

At each and every instant (for any value of n), the output is not bounded.

Therefore, the system is BIBO unstable.

(iii) Stable

Hints: $y(n) = \cos(n)B_x$, $-1 < \cos(n) < 1$ *for any value of* n

(iv) Unstable

Hints: $y(n) = B_x/\sin(n)$, *when* $\sin(n) = 0, y(n) = \infty$

Stable Vs unstable System (Cont.)

EXAMPLE 2.2.7

Consider the nonlinear system described by the input–output equation

$$y(n) = y^2(n-1) + x(n)$$

As an input sequence we select the bounded signal

$$x(n) = C\delta(n)$$

where C is a constant. We also assume that $y(-1) = 0$. Then the output sequence is

$$y(0) = C, \quad y(1) = C^2, \quad y(2) = C^4, \quad \ldots, \quad y(n) = C^{2n}$$

Clearly, the output is unbounded when $1 < |C| < \infty$. Therefore, the system is BIBO unstable, since a bounded input sequence has resulted in an unbounded output.

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-3
Analysis of DT Linear Time-Invariant System

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Week 8
Slide 92-117

Analysis of DT Linear Time-Invariant System

There are two basic methods for analyzing the behavior or response of a linear system to a given input signal:

Method-1: This method is based on the direct solution of the input-output equation for the system which is called the difference equation.

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k)$$

Method-2:

In this method, the input signal x(n) is decomposed or resolved into a sum of elementary signals. The elementary signals are selected so that the response of the system to each signal component is easily determined.

Then, using the linearity property of the system, the response of the system to the elementary signals are added to obtain the total response of the system to the given input signal.

Analysis of DT Linear Time-Invariant System (Cont.)

Elaboration of 2nd method

Suppose that, the input signal x(n) is resolved into a weighted sum of elementary signal components $\{x_k(n)\}$

$$x(n) = \sum_k c_k x_k(n)$$

C_k is the set of amplitudes (weighting coefficients)

Suppose the response of the system to the elementary signal component $x_k(n)$ is $y_k(n)$

$$y_k(n) \equiv \mathcal{T}[x_k(n)]$$

Considering the linearity property total response of input signal x(n) is

$$y(n) = \mathcal{T}[x(n)] = \mathcal{T}\left[\sum_k c_k x_k(n)\right]$$

$$= \sum_k c_k \mathcal{T}[x_k(n)]$$

$$= \sum_k c_k y_k(n)$$

Choice of elementary signal

If we place no restriction on the characteristics of input signal, then most convenient way to express the input sequence as weighted sum of unit sample (impulse) sequence.

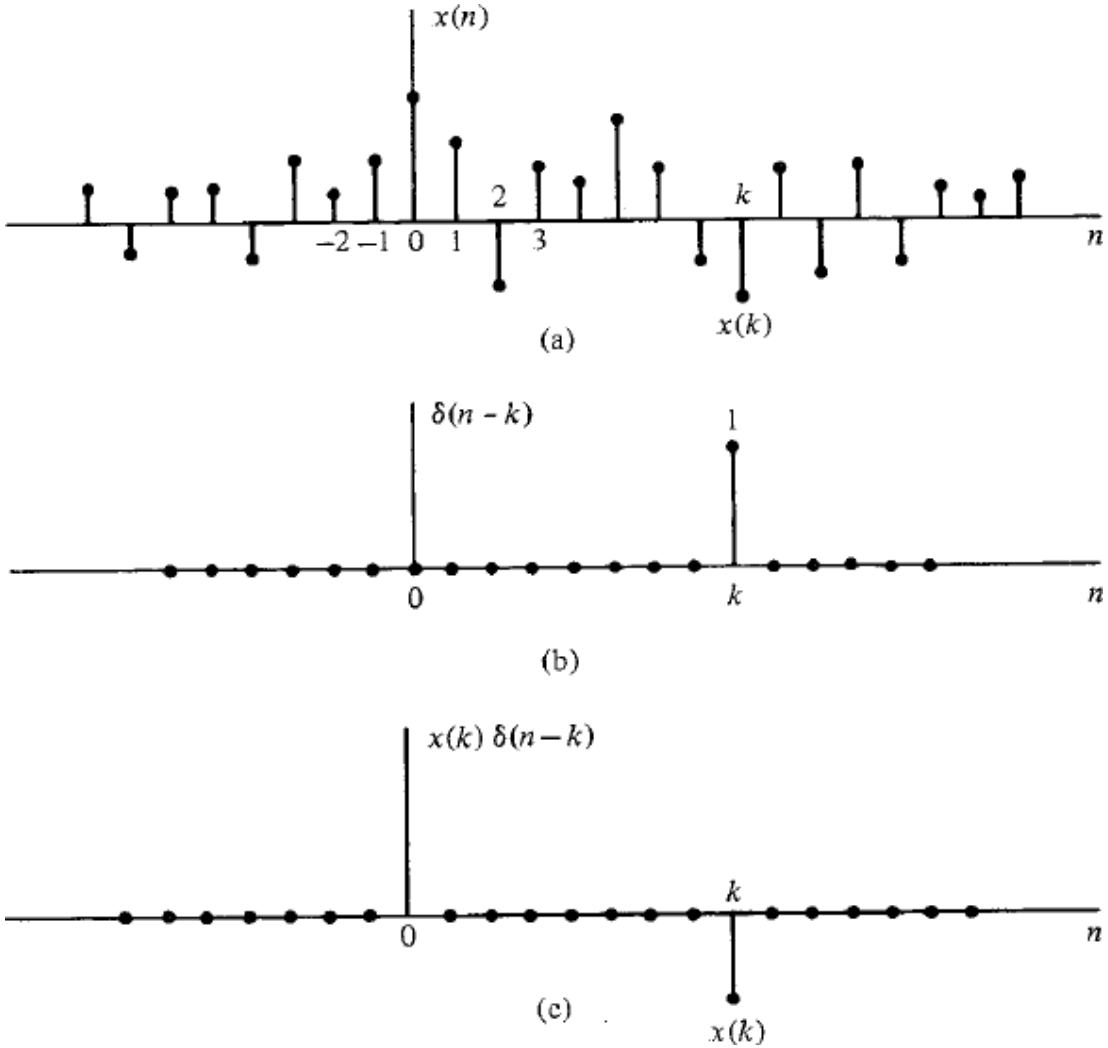
Analysis of DT Linear Time-Invariant System (Cont.)

Resolution of a Discrete Time Signal into Impulses

$$x_k(n) = \delta(n-k)$$

$$x(n)\delta(n-k) = x(k)\delta(n-k)$$

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k)$$



Analysis of DT Linear Time-Invariant System (Cont.)

Example 2.3.1: Consider the special case of a finite-duration sequence given as

$$x(n) = \{2, \underset{\uparrow}{4}, 0, 3\}$$

Resolve the sequence x(n) into a sum of weighted impulse sequences

Solution: Since the sequence x(n) is nonzero for the instants n=-1,0,2, we need three impulses at delays k= -1,0 and 2.

$$x(n) = 2\delta(n+1) + 4\delta(n) + 3\delta(n-2)$$

Analysis of DT Linear Time-Invariant System (Cont.)

Response of LTI System to arbitrary Inputs: The Convolution Sum

Response of the system for the unit sample sequence at n=k ; $y(n,k) \equiv h(n,k) = \mathcal{T}[\delta(n-k)]$

Resolving the sequence x(n) into a sum of impulse sequence, $x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k)$

The response of the system for input x(n)

$$y(n) = \mathcal{T}[x(n)] = \mathcal{T}\left[\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right]$$

$$= \sum_{k=-\infty}^{\infty} x(k)\mathcal{T}[\delta(n-k)]$$

$$= \sum_{k=-\infty}^{\infty} x(k)h(n,k)$$

Time- Invariance property

$$h(n) \equiv \mathcal{T}[\delta(n)]$$

Using Time Invariance property

$$h(n-k) = \mathcal{T}[\delta(n-k)]$$

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

Analysis of DT Linear Time-Invariant System (Cont.)

Response of LTI System to arbitrary Inputs: The Convolution Sum

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$ **The Convolution Sum**

The above equation that gives the response y(n) of the LTI system as a function of the input signal x(n) and unit sample (impulse) response h(n) is called a convolutional sum.

The convolution sum is used to compute the output of a LTI system for a given input x[n] and impulse response h[n].

The process of computing convolution involves the following four steps:

1. *Folding*. Fold $h(k)$ about $k = 0$ to obtain $h(-k)$.
2. *Shifting*. Shift $h(-k)$ by n_0 to the right (left) if n_0 is positive (negative), to obtain $h(n_0 - k)$.
3. *Multiplication*. Multiply $x(k)$ by $h(n_0 - k)$ to obtain the product sequence $v_{n_0}(k) \equiv x(k)h(n_0 - k)$.
4. *Summation*. Sum all the values of the product sequence $v_{n_0}(k)$ to obtain the value of the output at time $n = n_0$.

Analysis of DT Linear Time-Invariant System (Cont.)

Example (1) of Convolution Sum

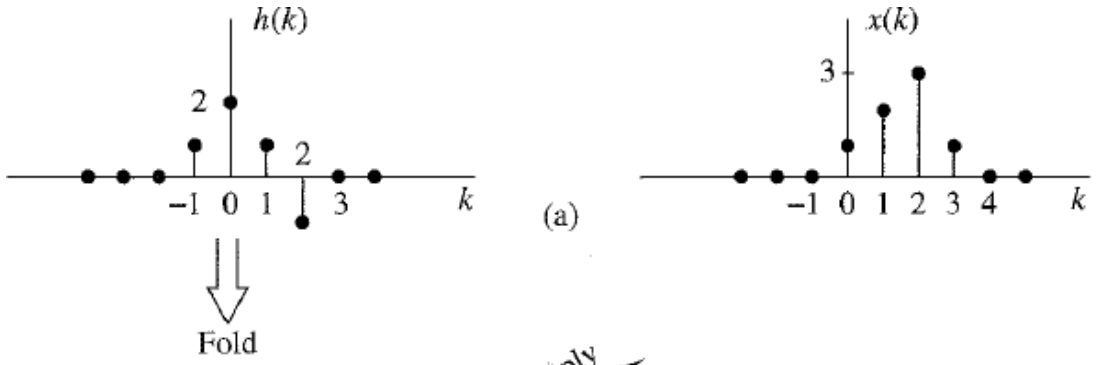
The impulse response of a linear time-invariant system is $h(n) = \{1, \underset{\uparrow}{2}, 1, -1\}$

Determine the response of the system to the input signal $x(n) = \{\underset{\uparrow}{1}, 2, 3, 1\}$

Solution

The Convolution Sum

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$



The output at n=0

$$y(0) = \sum_{k=-\infty}^{\infty} x(k)h(-k)$$

$$v_0(k) \equiv x(k)h(-k)$$

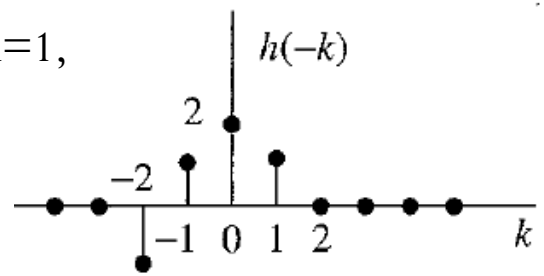
$$y(0) = \sum_{k=-\infty}^{\infty} v_0(k) = 4$$

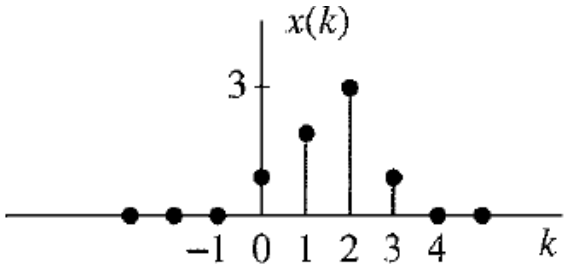
Analysis of DT Linear Time-Invariant System (Cont.)

Example (1) of Convolution Sum (Cont.)

The response of the system at n=1,

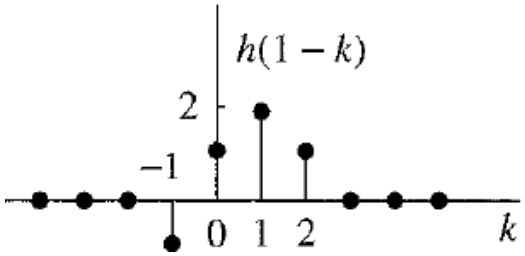
$$y(1) = \sum_{h=-\infty}^{\infty} x(k)h(1-k)$$



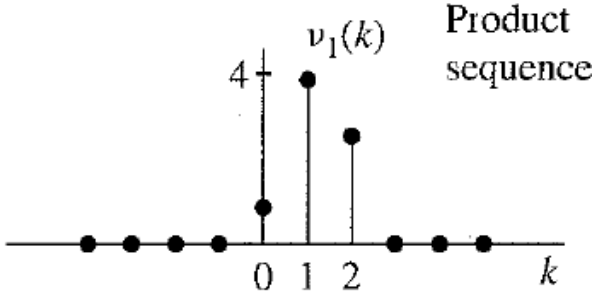


$$v_1(k) = x(k)h(1-k)$$

$$y(1) = \sum_{k=-\infty}^{\infty} v_1(k) = 8$$



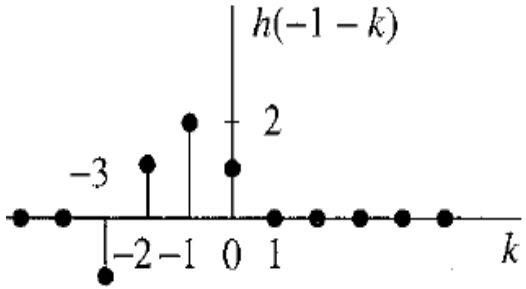
Multiply with $x(k)$



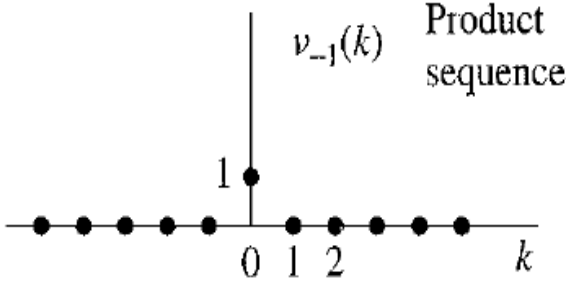
The response of the system at n=-1,

$$y(-1) = \sum_{k=-\infty}^{\infty} x(k)h(-1-k)$$

$$y(-1) = 1$$



Multiply with $x(k)$



Analysis of DT Linear Time-Invariant System (Cont.)

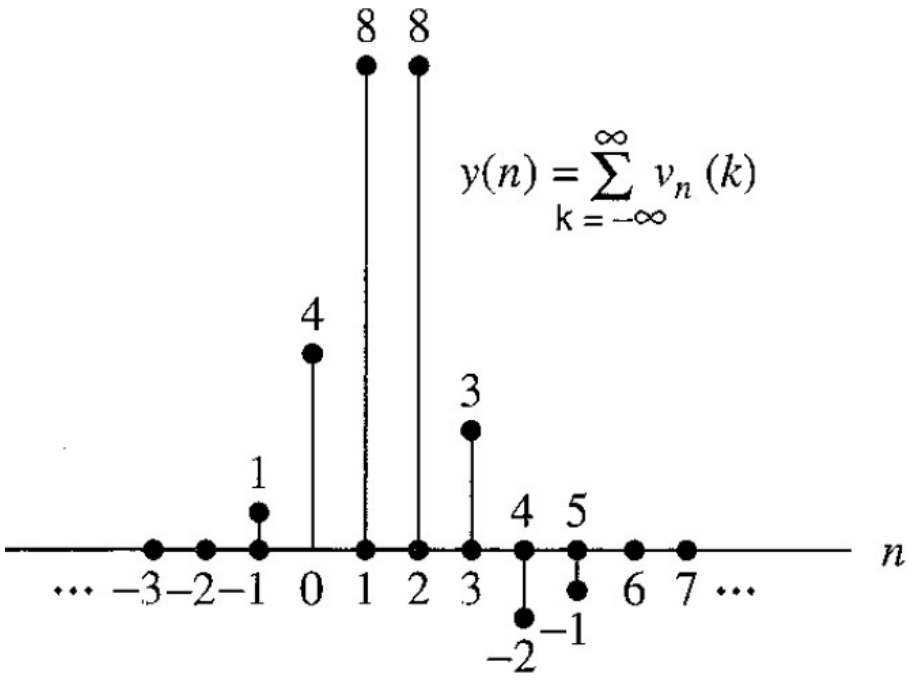
Example (1) of Convolution Sum (Cont.)

In similar manner, we obtained
$y(2)= 8, y(3)=3, y(4)= -2, y(5)=-1$
and $y(n)=0$, for $n>5$

Also, $y(-2)=0$ and $y(n)=0$ for $n<-1$

The enter response of the system ,

$$y(n) = \{\ldots, 0, 0, 1, \underset{\uparrow}{4}, 8, 8, 3, -2, -1, 0, 0, \ldots\}$$



Analysis of DT Linear Time-Invariant System (Cont.)

Commutative properties of Convolution Sum

We know the equation of convolution sum,

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

Defining a new index, m=n-k, we can write k=n-m. The above equation can be expressed as:

$$y(n) = \sum_{m=-\infty}^{\infty} x(n-m)h(m)$$

Since **m** is a dummy index, we may simply replace **m** by **k** so that

$$y(n) = \sum_{k=-\infty}^{\infty} x(n-k)h(k)$$

$$x(n) * h(n) = h(n) * x(n)$$

Analysis of DT Linear Time-Invariant System (Cont.)

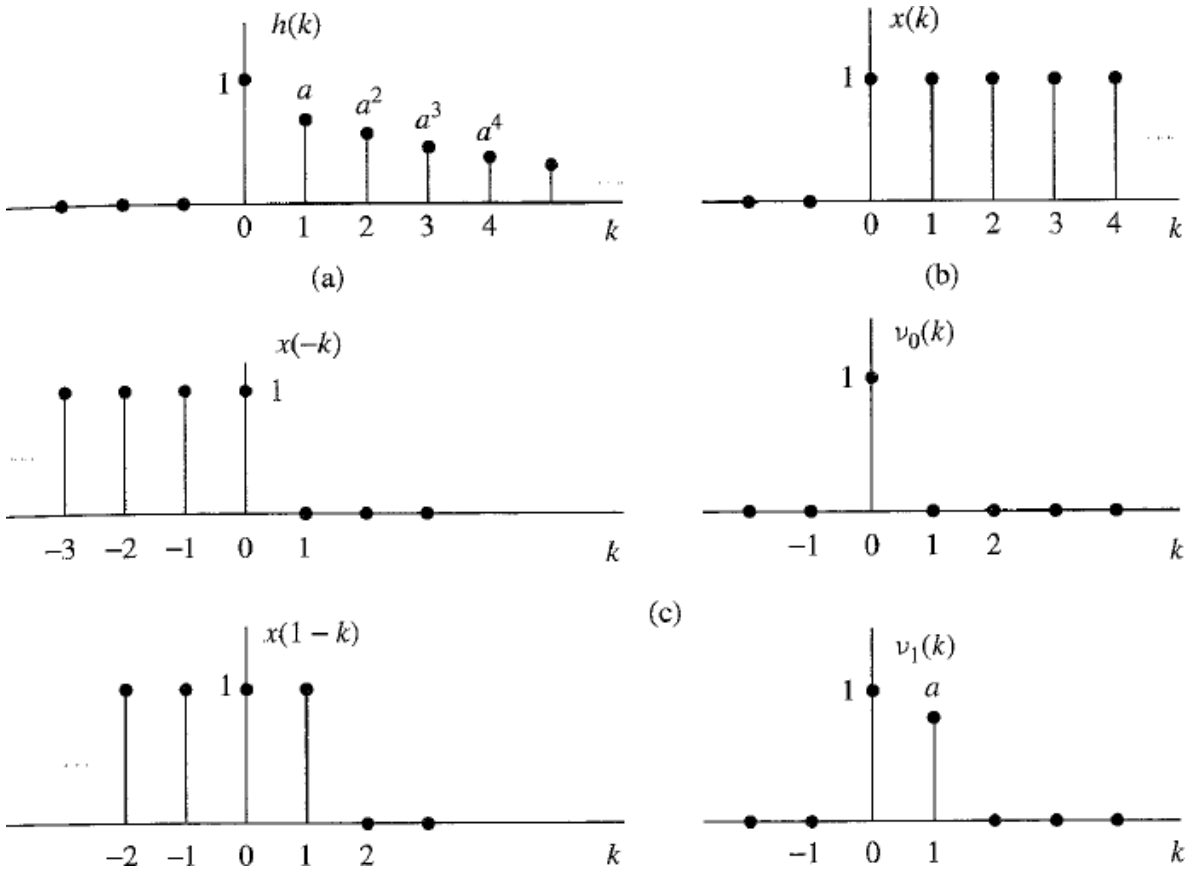
Example (2) of Convolution Sum

Determine the output y(n) of a relaxed linear time-invariant system with impulse response

$$h(n) = a^n u(n), |a| < 1$$

When the input is a unit step sequence, that is, x(n)=u(n)

Solution



$$y(0) = 1$$

$$y(1) = 1 + a$$

Analysis of DT Linear Time-Invariant System (Cont.)

Example (2) of Convolution Sum (Cont.)

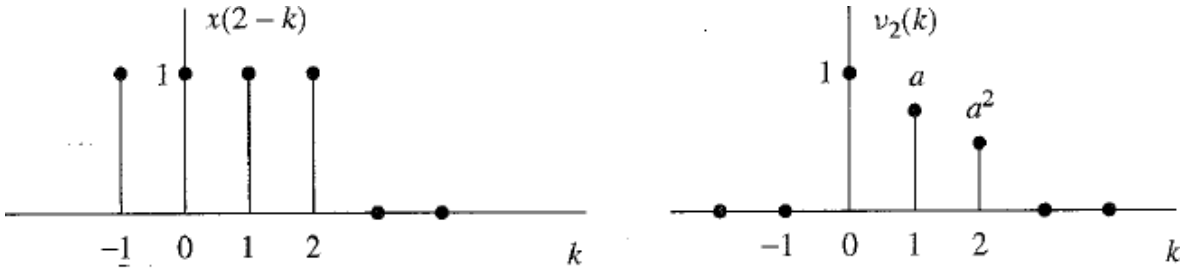
$$y(2) = 1 + a + a^2$$

Clearly, for n>0, the output is

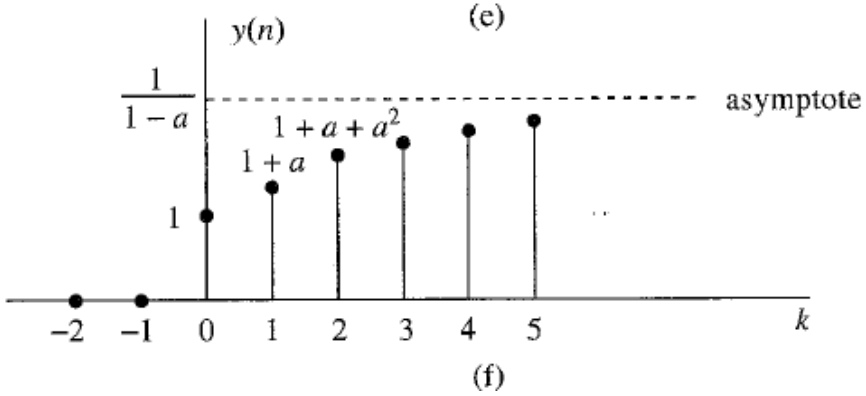
$$y(n) = 1 + a + a^2 + \cdots + a^n$$

$$= \frac{1 - a^{n+1}}{1 - a}$$

On the other hand for n<0, the output is y(n)=0



(e)



(f)

The final value of output as n approaches infinity is

$$y(\infty) = \lim_{n \to \infty} y(n) = \frac{1}{1 - a}$$

A plot of the output y(n) is illustrated in Fig (f)

Example (3) of Convolution Sum

Find the total response when the input function is $x(n) = \left(\frac{1}{2}\right)^n u(n)$ and the impulse response is given by $h(n) = \left(\frac{1}{3}\right)^n u(n)$

Applying the convolution formula,

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) = \sum_{k=-\infty}^{\infty} \left(\frac{1}{2}\right)^k u(k) \left(\frac{1}{3}\right)^{n-k} u(n-k)$$

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

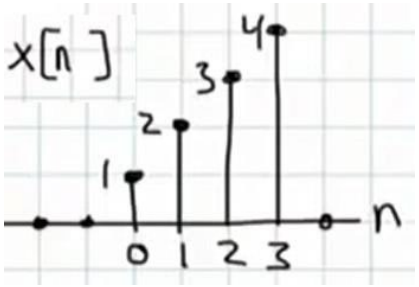
$$= \left(\frac{1}{3}\right)^n \frac{1-\left(\frac{3}{2}\right)^{n+1}}{1-\left(\frac{3}{2}\right)}$$

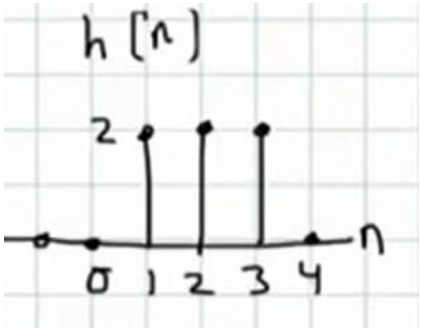
$$= (-2)\left(\frac{1}{3}\right)^n u(n) + 3\left(\frac{1}{2}\right)^n u(n)$$

Analysis of DT Linear Time-Invariant System (Cont.)

Practice Problems

1) The impulse response of a system is h(n)=u(n), find the output of the system when input x(n)=u(n).
2) The input and impulse response of a system is given below. Find the output of the system.





Analysis of DT Linear Time-Invariant System (Cont.)

System with Finite-Duration and Infinite-Duration Impulse Response

Linear time-invariant system may have finite duration impulse response (FIR) or infinite duration impulse response (IIR).

An FIR system has an impulse response that is zero outside of some finite time interval. For the causal FIR systems we can write:

$h(n)=0$, $n<0$ and $n \geq M$

The convolution formula for such system reduces to

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k)$$

Output at any time n is simply a weighted linear combination of the input signal samples x(n), x(n-1), …, x(n-M+1). An FIR system has a finite memory of length M samples.

An IIR linear time-invariant system has an infinite-duration impulse response. The output of IIR system based on convolution formula, is

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k)$$

In this case, the system output is a weighted [by the impulse response h(k)] linear combination of the input signal samples x(n), x(n-1), x(n-2), ….. Since this weighted sum involves the present and all the past input samples, we say the system has an infinite memory.

Recursive and Non-recursive discrete-time system

The convolution summation formula expresses the output of the linear time-invariant system explicitly and only in terms of the input signal. There are many systems where it is either necessary or desirable to express the output of the system not only in terms of the present and past values of the input, but also in terms of the already available past output values. In general, a system whose output y(n) at time n depends on any number of past output values y(n-1), y(n-2), …. is called a **recursive system**.

Example of recursive system:

Computation of cumulative average of a signal x(n) in the interval $0 \leq k \leq n$

$$y(n) = \frac{1}{n+1} \sum_{k=0}^{n} x(k), \qquad n = 0,1, \ldots \ldots$$

The computation of y(n) requires the storage of all the input samples x(k) for $0 \leq k \leq n$. Since n is increasing, memory requirements of the system grow linearly with time.

However y(n) can be computed more efficiently by utilizing the previous output value y(n-1)

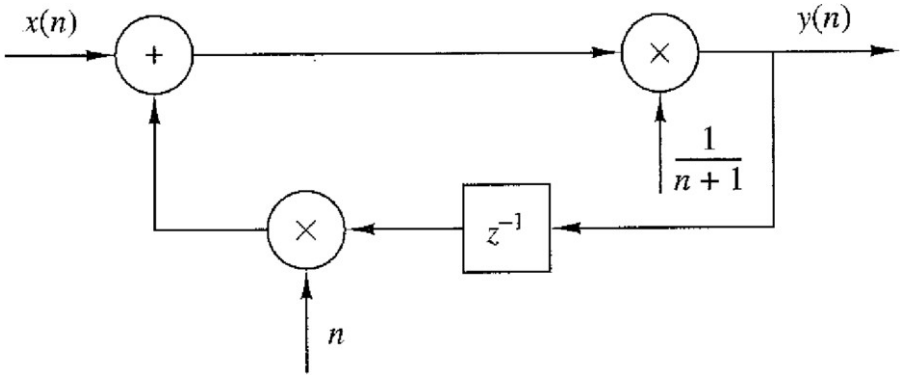
$$(n+1)y(n) = \sum_{k=0}^{n-1} x(k) + x(n) = ny(n-1) + x(n)$$

Hence,

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

Recursive and Non-recursive discrete-time system (Cont.)

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



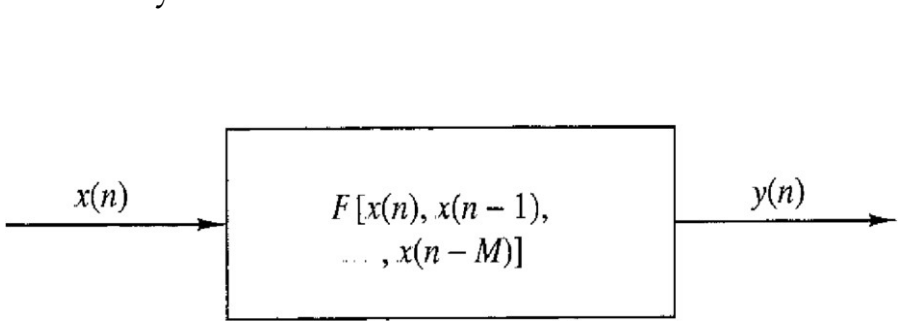
The output of a causal and practically realizable system recursive system can be expressed in general as

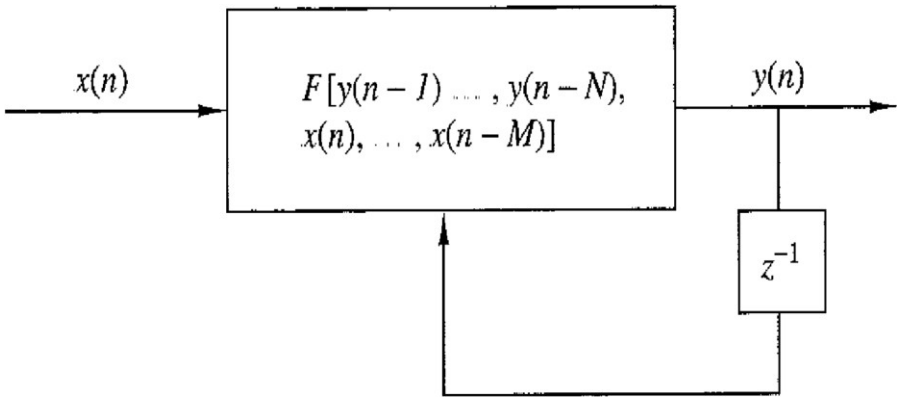
$$y(n) = F[y(n-1), y(n-2), \ldots, y(n-N), x(n), x(n-1), \ldots, x(n-M)]$$

If y(n) of a system depends only on the present and past inputs, then

$$y(n) = F[x(n), x(n-1), \ldots, x(n-M)]$$

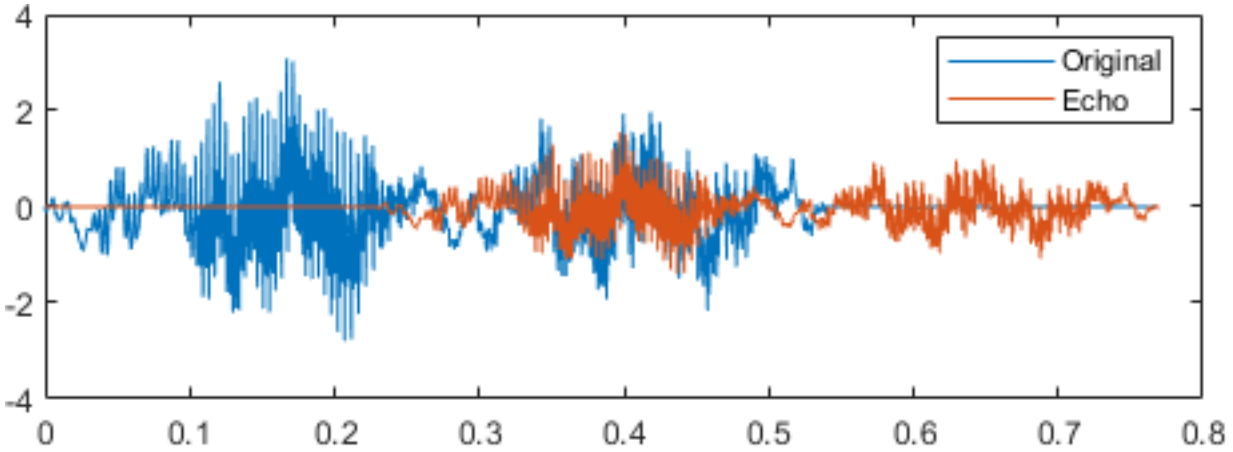
Such a system is **called non-recursive.**





Correlation of DT signals

- A mathematical operation that closely resembles convolution is correlation. In convolution the input and impulse response are involved whereas in correlation two signal sequences are involved.
- The correlation between the two signals is to measure the degree to which the two signals are similar and thus to extract some information that depends to a large extent on the application.
- Correlation of signals is often uncounted in radar, sonar, digital communications, geology and other areas in science and engineering.



Application of Correlation

Correlation in radar and active sonar applications

x(n) is the transmitted signal and **y(n)** is the received signal.

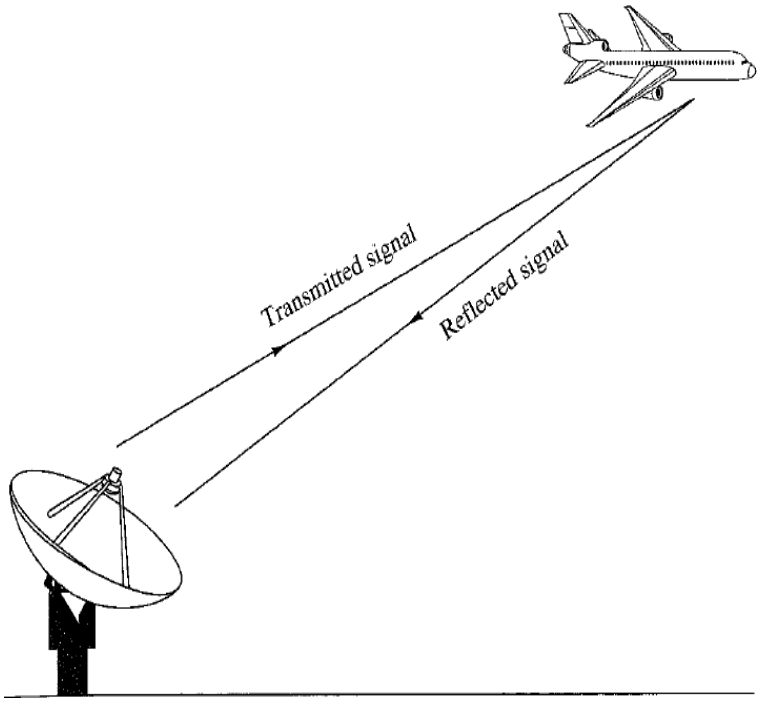
If a target is present y(n) will be

$$y(n) = \alpha x(n - D) + w(n)$$

Where α is some attenuation factor representing the signal loss involved in the round-trip transmission of the signal x(n), D is the round trip delay and w(n) represents additive noise that is picked up by the antenna and any noise generated by the electronic components and amplifier.

If there is no target,

y(n)=w(n)



Comparing two signal x(n) and y(n) radar detects whether a target is present or not and also calculate the distance if target is present. In practice, the signal x(n-D) is heavily corrupted by the additive noise to the point where a visual inspection of y(n) does not reveal the presence or absence of the desired signal reflected from the target. Correlation provides us with a means for extracting this important information from y(n).

Correlation in radar and active sonar applications (Cont.)

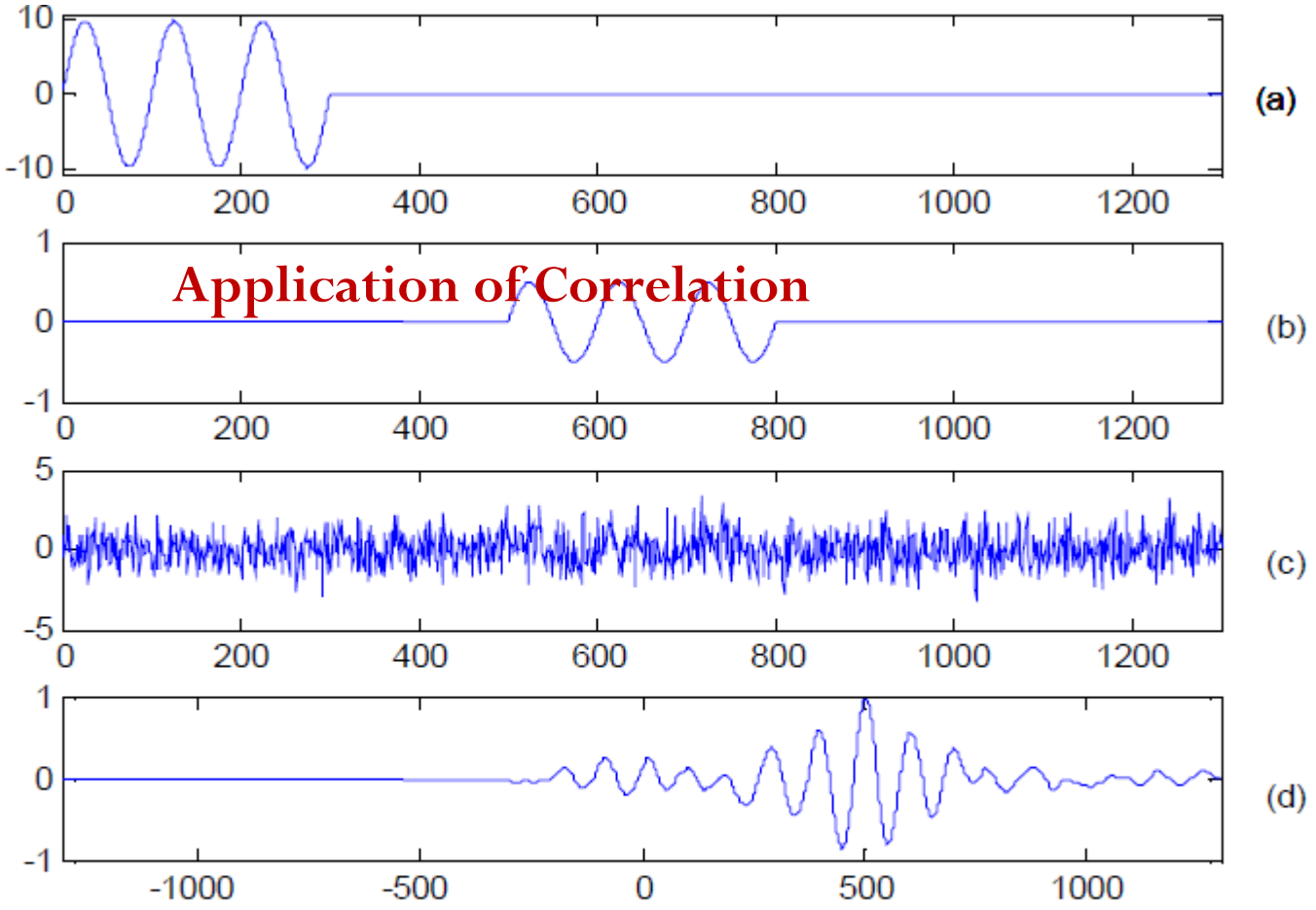


Figure :(a) Transmitted tone burst (b)Received weak echo(c)Received echo buried into background noise(d)CCF between (a) and (c) to locate a weak echo. It shows that after 500 units delay an echo arrives (location of the peak).

Application of Correlation (Cont.)

In digital Communication

Digital communication is another area where correlation is often used. In digital communications the information to be transmitted from one point to another is usually converted to binary form, that is, a sequence of zeros and ones, which are then transmitted to the intended receiver.

Signal sequence to transmit a logic 0: $x_0(n)$ for $0 \leq n \leq L-1$

Signal sequence to transmit a logic 1: $x_1(n)$ for $0 \leq n \leq L-1$

L represents the number of samples in each sequence.

The received signal can be represented as-

$$y(n) = x_i(n) + w(n), \qquad i = 0,1, \qquad 0 \leq n \leq L-1$$

w(n) represents the additive noise.

After receiving y(n), the receiver compares the received signal y(n) with both $x_0(n)$ and $x_1(n)$ to determine which of the two signals better matches y(n). The comparison process is performed by means of the correlation operation.

Crosscorrelation and Autocorrelation

Crosscorrelation

Suppose that we have two real signal sequence x(n) and y(n) each of which has finite energy. The crosscorrelation of x(n) and y(n) is a sequence $r_{xy}(l)$, which is defined as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l), \qquad l = 0, \pm 1, \pm 2, \ldots \ldots$$

Or, equivalently, as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n+l)y(n), \qquad l = 0, \pm 1, \pm 2, \ldots \ldots$$

Autocorrelation

In special case where both signal sequences are same (i.e. y(n)=x(n)), we have the autocorrelation of x(n), which is defined as the sequence

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n)x(n-l), \qquad l = 0, \pm 1, \pm 2, \ldots \ldots$$

Difference between Correlation and Convolution

- ✓ In the computation of convolution, one of the sequence is folded, then shifted, then multiplied by the other sequence to form the product sequence for that shift, and finally, the values of the product sequence are summed.
- ✓ Except for the folding operation, the computation of the crosscorrelation sequence involves the same operation: shifting one of the sequence, multiplying the two sequence, and summing over all values of the product sequence.
- ✓ So, if we first fold a sequence y(n) to y(-n) and find the convolution between two sequences x(n) and y(-n), it results crosscorrelation between x(n) and y(n).

$$r_{xy}(l) = x(l) * y(-l)$$

Example of determining Correlation Sequence

Determine the crosscorrelation sequence $r_{xy}(l)$ of the sequences

$$x(n) = \{2, -1, 3, 7, 1, 2, -3\}$$
$$\uparrow$$
$$y(n) = \{1, -1, 2, -2, 4, 1, -2, 5\}$$
$$\uparrow$$

Solution:

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l), \qquad l = 0, \pm 1, \pm 2, \ldots\ldots$$

n	-4	-3	-2	-1	0	1	2	3	l
x(n)	2	-1	3	7	1	2	-3	0	l=0
y(n)	1	-1	2	-2	4	1	-2	5	
∑x(n)y(n)	2	1	6	-14	4	2	6	0	7
y(n+1)	-1	2	-2	4	1	-2	5	0	l=-1
∑x(n)y(n+1)	-2	-2	-6	28	1	-4	-15	0	0
y(n+2)	2	-2	4	1	-2	5	0	0	l=-2
∑x(n)y(n+2)	4	2	12	7	-2	10	0	0	33

$r_{xy}(0) = 7$
$r_{xy}(-1) = 0$
$r_{xy}(-2) = 33$

Example of determining Correlation Sequence (Cont.)

n	-4	-3	-2	-1	0	1	2	l
x(n)	2	-1	3	7	1	2	-3	
y(n+3)	-2	4	1	-2	5			l=-3
∑x(n)y(n+3)	-4	-4	3	-14	5			-14
y(n+4)	4	1	-2	5	0			l=-4
∑x(n)y(n+4)	8	-1	-6	35	0			36
y(n+5)	1	-2	5	0				l=-5
∑x(n)y(n+5)	2	2	15					19
y(n+6)	-2	5						l=-6
∑x(n)y(n+6)	-4	-5						-9
y(n+7)	5							l=-7
∑x(n)y(n+7)	10							10

$r_{xy}(-3) = -14$

$r_{xy}(-4) = 36$

$r_{xy}(-5) = 19$

$r_{xy}(-6) = -9$

$r_{xy}(-7) = 10$

For n<-7

$r_{xy} = 0$

Example of determining Correlation Sequence (Cont.)

$r_{xy}(1) = 13$
$r_{xy}(2) = -18$
$r_{xy}(3) = 16$
$r_{xy}(4) = -7$
$r_{xy}(5) = 5$
$r_{xy}(6) = -3$
For n>6
$r_{xy} = 0$

The maximum similarity between two signals x(n) and y(n) obtained when y(n) is delayed by 4 positions.

n	-3	-2	-1	0	1	2	l
x(n)	-1	3	7	1	2	-3	
y(n-1)	1	-1	2	-2	4	1	l=1
∑x(n)y(n-1)	-1	-3	14	-2	8	-3	13
y(n-2)		1	-1	2	-2	4	l=2
∑x(n)y(n-2)		3	-7	2	-4	-12	-18
y(n-3)			1	-1	2	-2	l=3
∑x(n)y(n-3)			7	-1	4	6	16
y(n-4)				1	-1	2	l=4
∑x(n)y(n-4)				1	-2	-6	-7
y(n-5)					1	-1	l=5
∑x(n)y(n-5)					2	3	5
y(n-6)						1	l=6
∑x(n)y(n-6)						-3	-3

Therefore, the crosscorrelation sequence of $x(n)$ and $y(n)$ is

$$r_{xy}(l) = \{10, -9, 19, 36, -14, 33, 0, \underset{\uparrow}{7}, 13, -18, 16, -7, 5, -3\}$$

Class Test Next Week

Syllabus: Slide 56-118

Assignment: Given in Google Drive Folder

Submission Deadline : Before Mid-term Exam

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-4
Z-Transform

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ Z-transform
- ✓ Physical significance of z-transform
- ✓ Region of convergence (ROC).
- ✓ Z-transform of some basic causal and anticausal signals
- ✓ Properties of z-transform
- ✓ Pole-zero Plot
- ✓ Inverse z-transform

Text Book:

Digital Signal Processing (4th Edition), John G. Proakis, Dimitris K Manolakis

Week 9
Slide 121-135

Z-Transform

The direct Z-Transform

The z-transform of a discrete-time signal x(n) is defined as the power series

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$$

Where z is a complex variable. The relation sometimes called the direct z-transform because it transforms the time-domain signal x(n) into its complex-plane representation X(z).

$$\mathbf{z = re^{j\omega}}$$

$$\boldsymbol{z^{-n} = r^{-n}e^{-j\omega n}}$$

$$\boldsymbol{= r^{-n}[cos(\omega n) - jsin(\omega n)]}$$

Where,
'r' is a real number
$e^{j\omega}$ is Euler's Number
ω is the angular frequency in radians per sample.

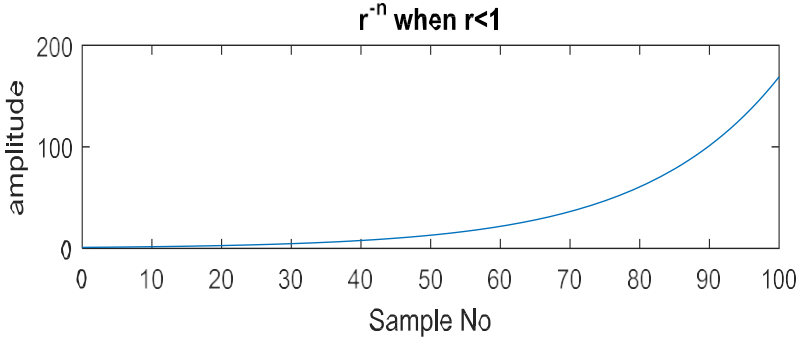
The z-transform of a discrete-time signal x(n) is denoted by

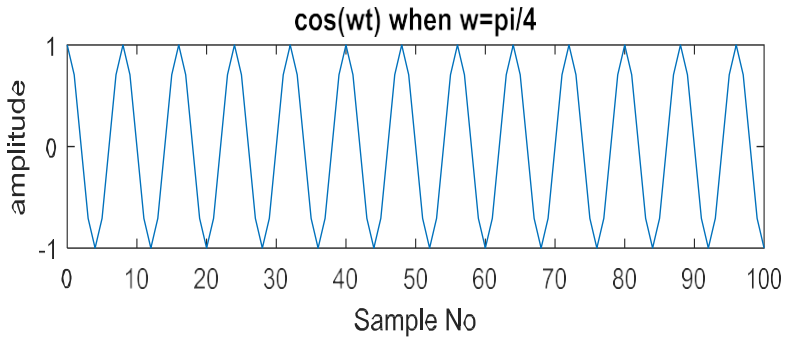
$$X(z) \equiv Z\{x(n)\}$$

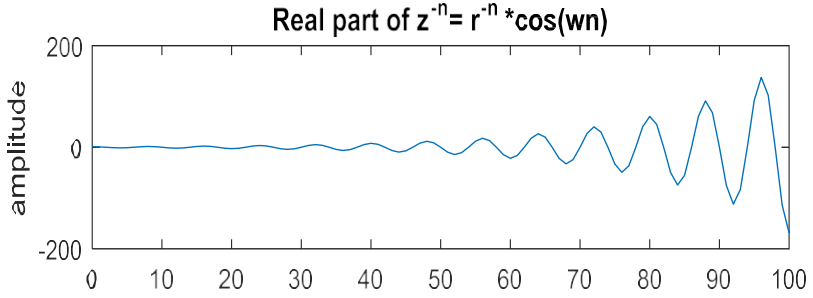
Whereas the relationship between x(n) and x(z) is indicated by

$$x(n) \overset{Z}{} X(z)$$

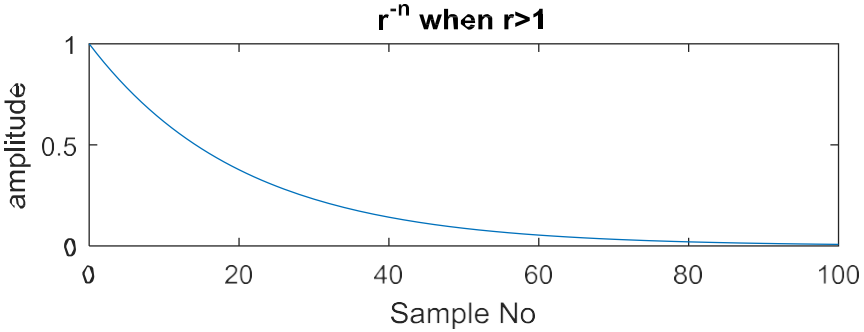
Significance of z^{-n}

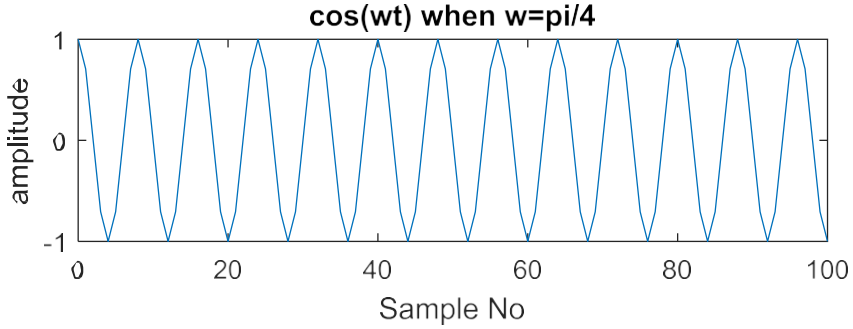


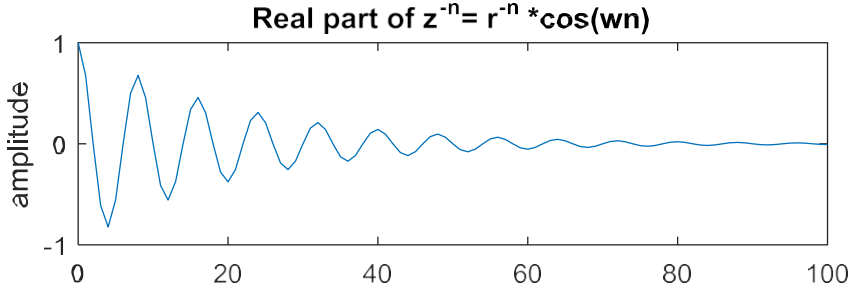




Plot of $r^{-n}cos\,(\omega n)$ when r<1 and $\omega = \frac{\pi}{4}$ per samples

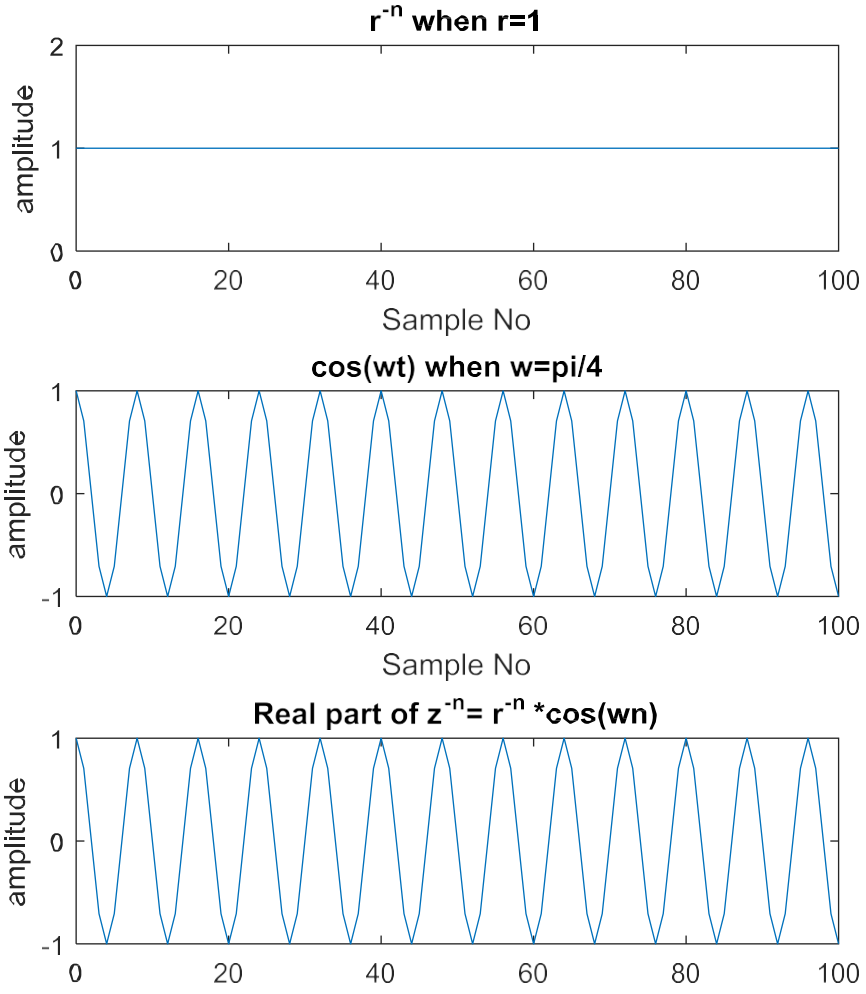






Plot of $r^{-n}cos\,(\omega n)$ when r>1 and $\omega = \frac{\pi}{4}$ per samples

Significance of z^{-n}



Plot of $r^{-n}cos\,(\omega n)$ when r=1 and $\omega = \frac{\pi}{4}$ per samples

$$z^{-n} = r^{-n}e^{-j\omega n}$$

$$= r^{-n}[cos(\omega n) - jsin(\omega n)]$$

When,

r>1, z^{-n} has exponentially decreasing oscillation

r<1, z^{-n} has exponentially increasing oscillation

r=1, z^{-n} has oscillation of constant amplitude.

So we can say

z^{-n} represents a set of oscillating components of constant or increasing or decreasing amplitude based on value of z

$$z = re^{j\omega} = a + jb$$

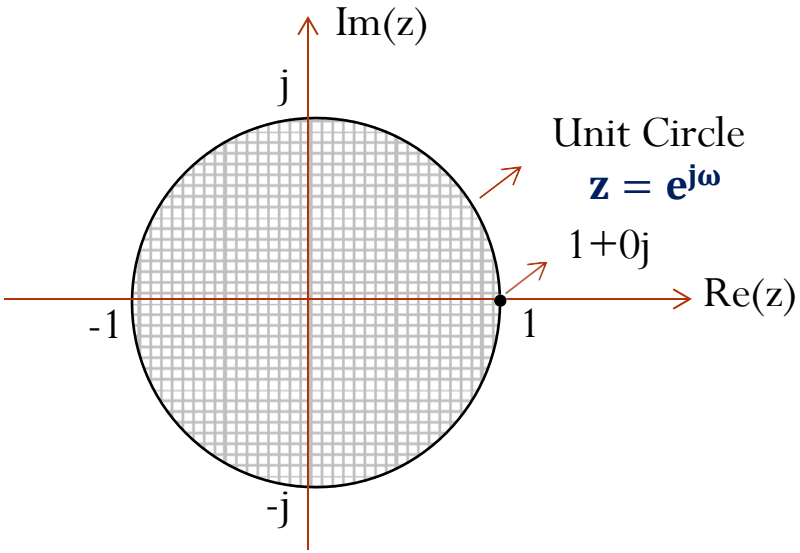
r is the magnitude or modulus of z controls the magnitude (increasing/decreasing/constant) of oscillation $r = |z| = \sqrt{a^2 + b^2}$

ω is the angle or argument or phase controls frequency of oscillation.

$$\omega = \angle z = \tan^{-1}\frac{b}{a}$$

Significance of Z-Transform

Argand Diagram



- Any point in the unit circle is associated with a complex number z which has a magnitude of 1 (z^{-n} has oscillation of constant amplitude)
- If z lies on the real axis of the argand diagram the signal z^{-n}won't oscillate.
  - * At point 1+0j, z^{-n} has a constant amplitude of 1.
  - * Increase exponentially, if it lies between 0 and 1.
  - * decrease exponentially if it lies after 1.
- If z lies outside of unit circle (but not in real axis) z^{-n} has oscillation with exponentially decreasing amplitude.
- If z lies inside of unit circle (but not in real axis) z^{-n} has oscillation with exponentially increasing amplitude.

Significance of Z-Transform

Significance of Z-Transform

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$$

In the above equation, x[n] is multiplied by z^{-n}(set of oscillating components of constant or increasing or decreasing amplitude based on value of z), very similar to correlation.

Z-transform is the measure of similarities of discrete time sequence x[n] with all the frequency of oscillations associate with z^{-n}. Z-transform identifies the presence of exponentially increasing or decreasing oscillations in the signal x[n].

Z-Transform of Impulse response

Taking the z-transform of a systems impulse response we get the following

- By identifying the presence of increasing and decreasing oscillations in the impulse response of a system we can determine if the system is stable or unstable.
- By identifying the presence of sinusoids in the impulse response of a system we can determine the systems frequency response. Note that a impulse input has all types of frequency components.

Region of Convergence (ROC) in Z-Transform

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n}$$

- Since the z-transform is an infinite power series, it exists only for those value of z for which this series converges.
- The region of convergence (ROC) of X(z) is the set of all values of z for which X(z) attains a finite value. Thus any time we cite a z-transform we should also indicates its ROC.

Poles and Zeros

When $X(z)$ is a rational function, i.e., a ration of polynomials in z, then:

1. The roots of the numerator polynomial are referred to as the zeros of $X(z)$, and
2. The roots of the denominator polynomial are referred to as the poles of $X(z)$.

Note that no poles of $X(z)$ can occur within the region of convergence since the z-transform does not converge at a pole.

Furthermore, the region of convergence is bounded by poles.

EXAMPLE 3.1.1

Determine the z-transforms of the following *finite-duration* signals.

(a) $x_1(n) = \{\underset{\uparrow}{1}, 2, 5, 7, 0, 1\}$

(b) $x_2(n) = \{1, 2, \underset{\uparrow}{5}, 7, 0, 1\}$

(c) $x_3(n) = \{\underset{\uparrow}{0}, 0, 1, 2, 5, 7, 0, 1\}$

(d) $x_4(n) = \{2, 4, \underset{\uparrow}{5}, 7, 0, 1\}$

(e) $x_5(n) = \delta(n)$

(f) $x_6(n) = \delta(n-k), k > 0$

(g) $x_7(n) = \delta(n+k), k > 0$

Solution. From definition (3.1.1), we have

(a) $X_1(z) = 1 + 2z^{-1} + 5z^{-2} + 7z^{-3} + z^{-5}$, ROC: entire z-plane except $z = 0$

(b) $X_2(z) = z^2 + 2z + 5 + 7z^{-1} + z^{-3}$, ROC: entire z-plane except $z = 0$ and $z = \infty$

(c) $X_3(z) = z^{-2} + 2z^{-3} + 5z^{-4} + 7z^{-5} + z^{-7}$, ROC: entire z-plane except $z = 0$

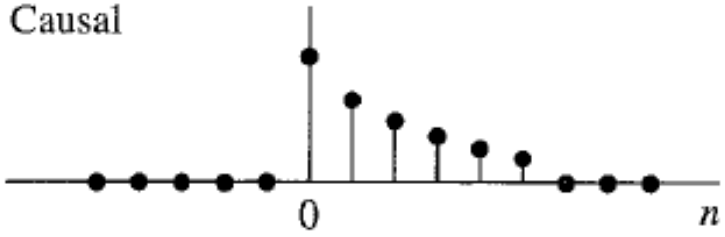
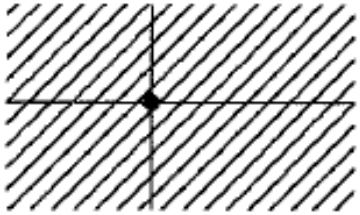
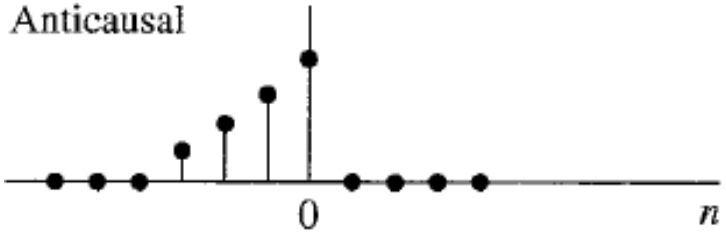
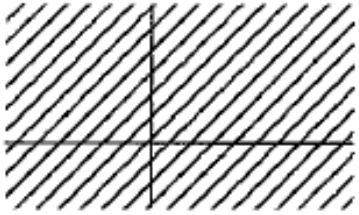
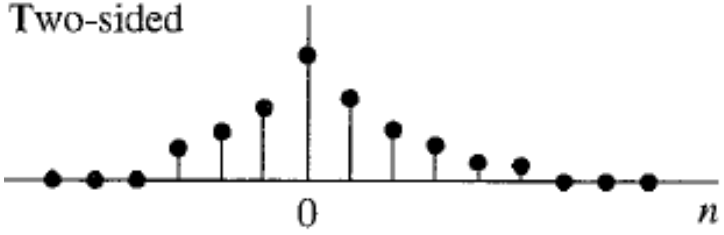
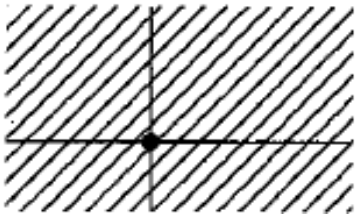
(d) $X_4(z) = 2z^2 + 4z + 5 + 7z^{-1} + z^{-3}$, ROC: entire z-plane except $z = 0$ and $z = \infty$

(e) $X_5(z) = 1$ [i.e., $\delta(n) \overset{z}{\longleftrightarrow} 1$], ROC: entire z-plane

(f) $X_6(z) = z^{-k}$ [i.e., $\delta(n-k) \overset{z}{\longleftrightarrow} z^{-k}$], $k > 0$, ROC: entire z-plane except $z = 0$

(g) $X_7(z) = z^k$ [i.e., $\delta(n+k) \overset{z}{\longleftrightarrow} z^k$], $k > 0$, ROC: entire z-plane except $z = \infty$

Characteristic Families of Signals with their corresponding ROCs

Signal	ROC
Finite-Duration Signals	
Causal 	 Entire z-plane except $z = 0$
Anticausal 	 Entire z-plane except $z = \infty$
Two-sided 	 Entire z-plane except $z = 0$ and $z = \infty$

Example 3.1.2 (Prokis)

Determine the z-transform of the signal

$$x(n) = (\frac{1}{2})^n u(n)$$

Solution:

x(n) can be expressed as $x(n) = \{1, (\frac{1}{2}), (\frac{1}{2})^2, (\frac{1}{2})^3, \cdots, (\frac{1}{2})^n, \cdots\}$

The z transform of x(n): $X(z) = 1 + \frac{1}{2}z^{-1} + (\frac{1}{2})^2 z^{-2} + (\frac{1}{2})^n z^{-n} + \cdots$

$$= \sum_{n=0}^{\infty} (\frac{1}{2})^n z^{-n} = \sum_{n=0}^{\infty} (\frac{1}{2}z^{-1})^n$$

Using the geometric series, $1 + A + A^2 + A^3 + \cdots = \frac{1}{1-A}$ if $|A| < 1$

We can write,

$$X(z) = \frac{1}{1 - \frac{1}{2}z^{-1}}, \qquad \text{ROC: } |\tfrac{1}{2}z^{-1}| < 1$$

$$|z| > \frac{1}{2}$$

ROC for Causal and Anticausal components of X(z)

$$z = re^{j\theta} \quad \text{where } r = |z| \text{ and } \theta = \measuredangle z.$$

$$X(z)|_{z=re^{j\theta}} = \sum_{n=-\infty}^{\infty} x(n)r^{-n}e^{-j\theta n}$$

In the ROC of $X(z)$, $|X(z)| < \infty$

$$|X(z)| = \left| \sum_{n=-\infty}^{\infty} x(n)r^{-n}e^{-j\theta n} \right|$$

$$\leq \sum_{n=-\infty}^{\infty} |x(n)r^{-n}e^{-j\theta n}| = \sum_{n=-\infty}^{\infty} |x(n)r^{-n}|$$

Here $|X(z)|$ is finite if the sequence $x(n)r^{-n}$ is absolutely summable.

The problem of finding the ROC for X(z) is equivalent to determining the range of values of r for which the sequence $x(n)r^{-n}$ is absolutely summable.

ROC for Causal and Anticausal components of X(z)- Cont.

$$|X(z)| \leq \sum_{n=-\infty}^{-1} |x(n)r^{-n}| + \sum_{n=0}^{\infty} \left|\frac{x(n)}{r^n}\right|$$

$$\leq \sum_{n=1}^{\infty} |x(-n)r^{n}| + \sum_{n=0}^{\infty} \left|\frac{x(n)}{r^n}\right|$$

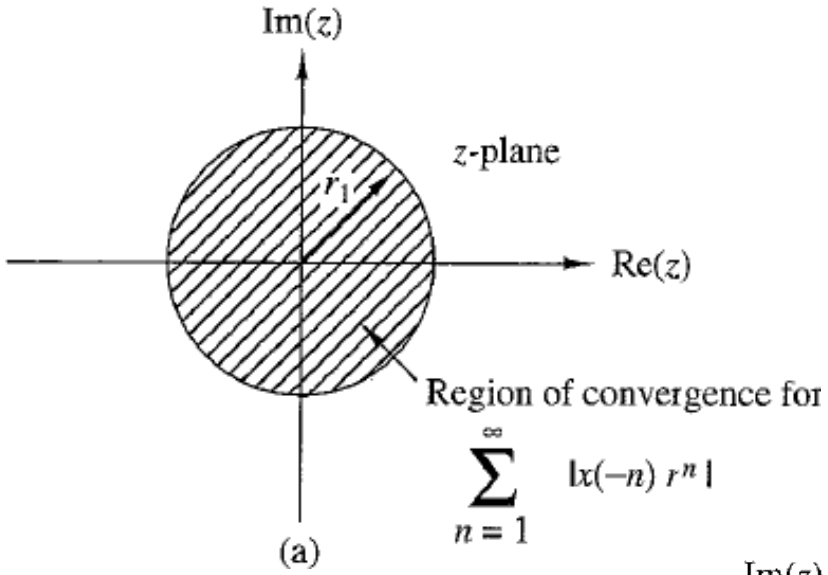
If X(z) converges in some region of complex plane, both summations in the above expression must be finite in that region.

If the first sum in expression converges, there must exist values of r small enough such that the product sequence $x(-n)r^n, 1 \leq n < \infty$, is absolutely summable. Therefore the ROC for the 1st some consists of all points in a circle of some radius r_1, where $r_1 \leq \infty$.

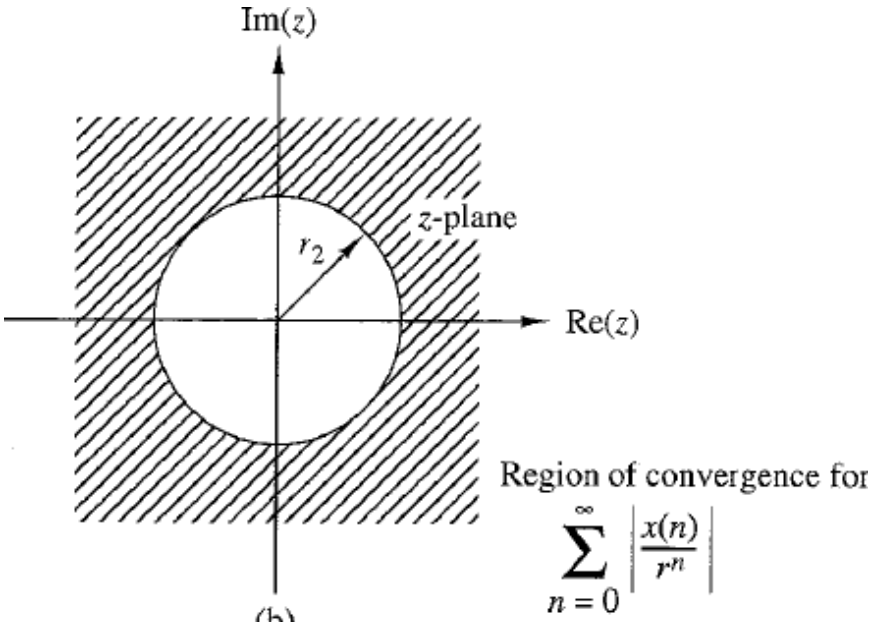
The second sum converges if there exist values of r large enough such that the product sequence, $\frac{x(n)}{r^n}, 0 \leq n \leq \infty$, is absolutely summable. The ROC for the second sum consists all points outside of circle of radius $r > r_2$.

ROC of X(z) is the common region in the z-plane ($r_1 < r < r_2$) where both sums are finite.

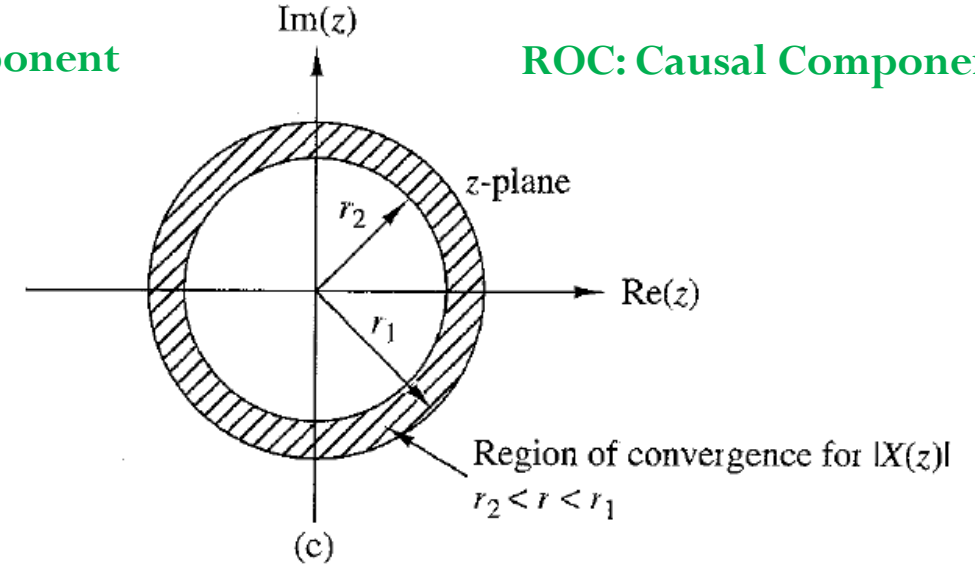
ROC for Causal and Anticausal components of X(z)- Cont.



ROC: Anticausal Component



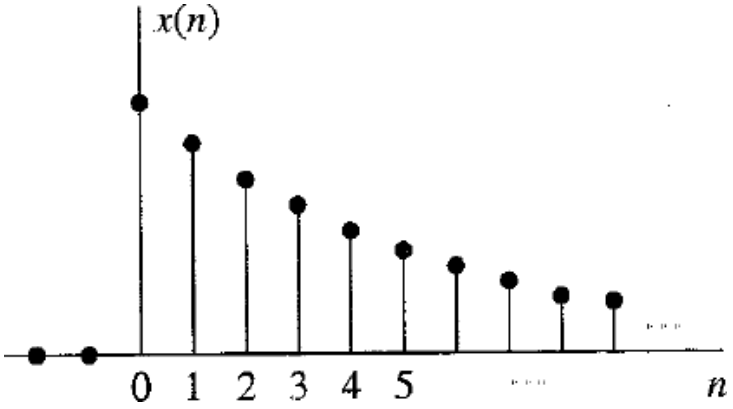
ROC: Causal Component



Example 3.1.3 (Prokis)

Determine the z-transform of the signal

$$x(n) = \alpha^n u(n) = \begin{cases} \alpha^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

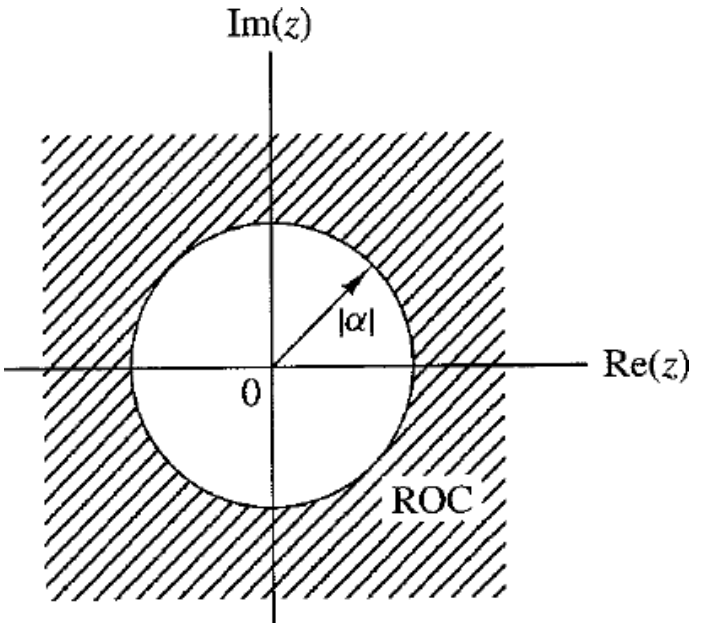


Solution:

$$X(z) = \sum_{n=0}^{\infty} \alpha^n z^{-n} = \sum_{n=0}^{\infty} (\alpha z^{-1})^n$$

If $|\propto z^{-1}| < 1$ or equivalently, $|z| > |\propto|$, this power series converges to $\frac{1}{1-\propto z^{-1}}$. Thus we have the z-transform pair

$$x(n) = \alpha^n u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{1 - \alpha z^{-1}}, \quad \text{ROC: } |z| > |\alpha|$$

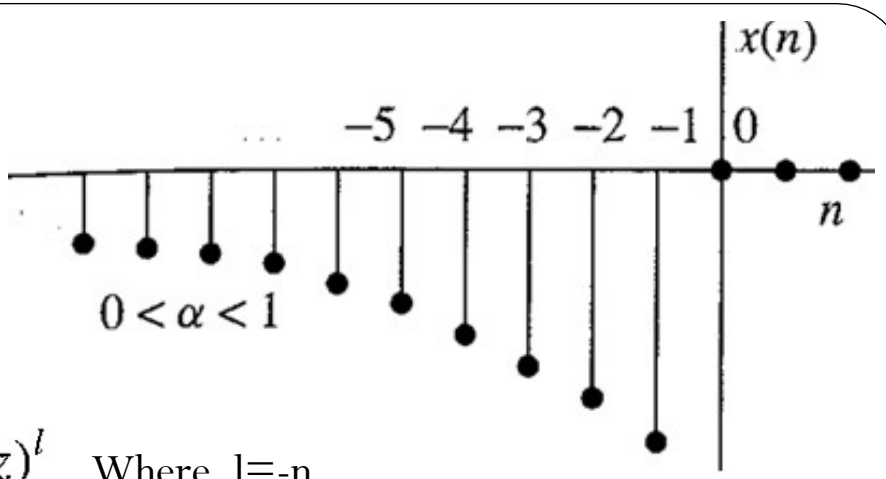


If $\propto = 1$, we obtained the z-transform of unit step signal

$$x(n) = u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{1 - z^{-1}}, \quad \text{ROC: } |z| > 1$$

Example 3.1.4 (Prokis)

Determine the z-transform of the signal



$$x(n) = -\alpha^n u(-n-1) = \begin{cases} 0, & n \geq 0 \\ -\alpha^n, & n \leq -1 \end{cases}$$

Solution:

$$X(z) = \sum_{n=-\infty}^{-1} (-\alpha^n) z^{-n} = -\sum_{l=1}^{\infty} (\alpha^{-1} z)^l$$ Where, l=-n

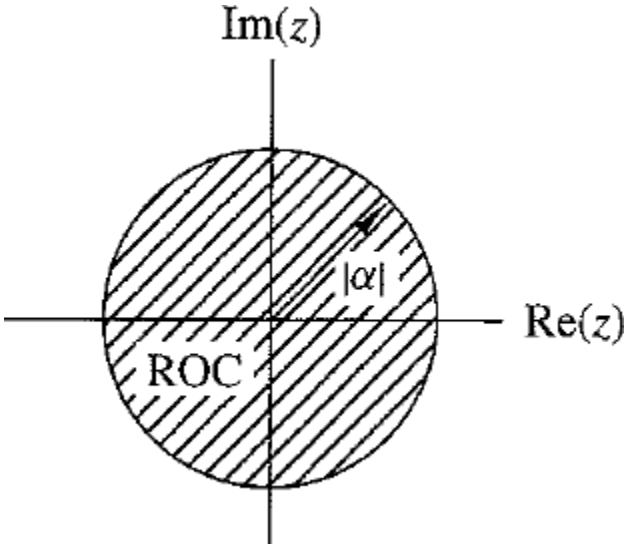
Using the formula,

$$A + A^2 + A^3 + \cdots = A(1 + A + A^2 + \cdots) = \frac{A}{1-A}$$

when $|A| < 1$ gives

$$X(z) = -\frac{\alpha^{-1} z}{1 - \alpha^{-1} z} = \frac{1}{1 - \alpha z^{-1}}$$

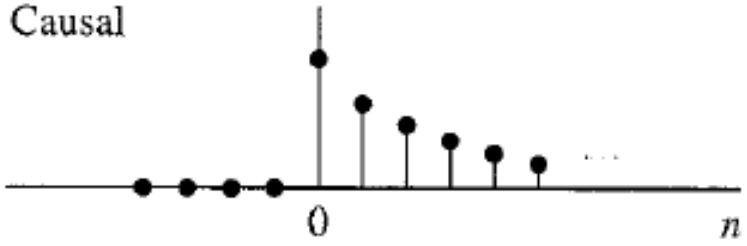
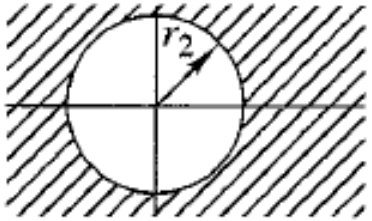
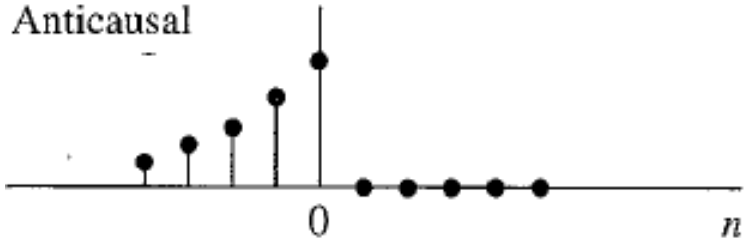
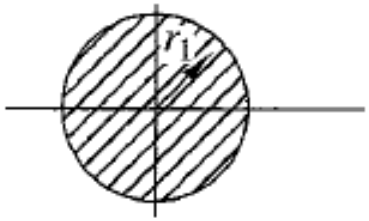
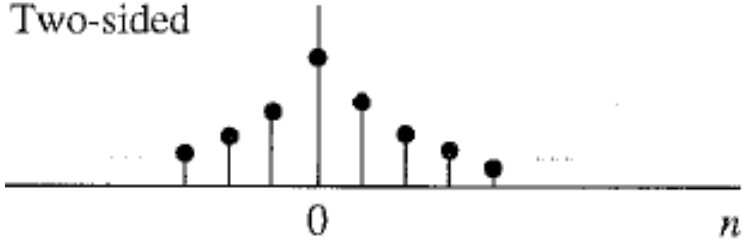
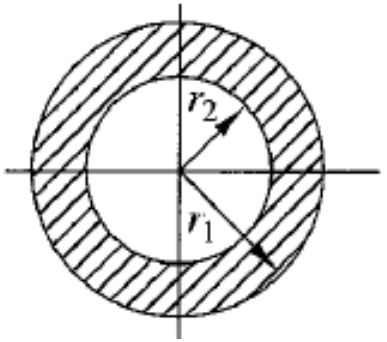
Provided that $|\alpha^{-1} z < 1|$ or, equivalently, $|z| < |\alpha|$. Thus



$$x(n) = -\alpha^n u(-n-1) \overset{z}{\longleftrightarrow} X(z) = -\frac{1}{1 - \alpha z^{-1}},$$

$$\text{ROC: } |z| < |\alpha|$$

Characteristic Families of Signals with their corresponding ROCs

Signal	ROC
Infinite-Duration Signals	
Causal 	 $\lvert z\rvert > r_2$
Anticausal 	 $\lvert z\rvert < r_1$
Two-sided 	 $r_2 < \lvert z\rvert < r_1$

Uniqueness of Z-Transform

- Z-transform does not uniquely specify the signal in the time domain without ROC.
- From the previous two examples, we see that the **causal signal $\alpha^n u(n)$** and **the anticausal signal $-\alpha^n u(-n-1)$** have identical closed-form expressions for the z-transform.

$$Z\{\alpha^n u(n)\} = Z\{-\alpha n u(-n-1)\} = \frac{1}{1-\alpha z^{-1}}$$

- A discrete-time signal x(n) is uniquely determined by not only its z-transform X(z), but also the region of convergence of X(z).
- The ROC of a causal signal is the exterior of a circle of some radius $r_2$while the ROC of an anticausal signal is the interior of a circle of some radius r_1

Example 3.1.5 (Prokis)

Determine the z-transform of the signal

$$x(n) = \alpha^n u(n) + b^n u(-n-1)$$

Solution:

$$X(z) = \sum_{n=0}^{\infty} \alpha^n z^{-n} + \sum_{n=-\infty}^{-1} b^n z^{-n} = \sum_{n=0}^{\infty} (\alpha z^{-1})^n + \sum_{l=1}^{\infty} (b^{-1} z)^l$$

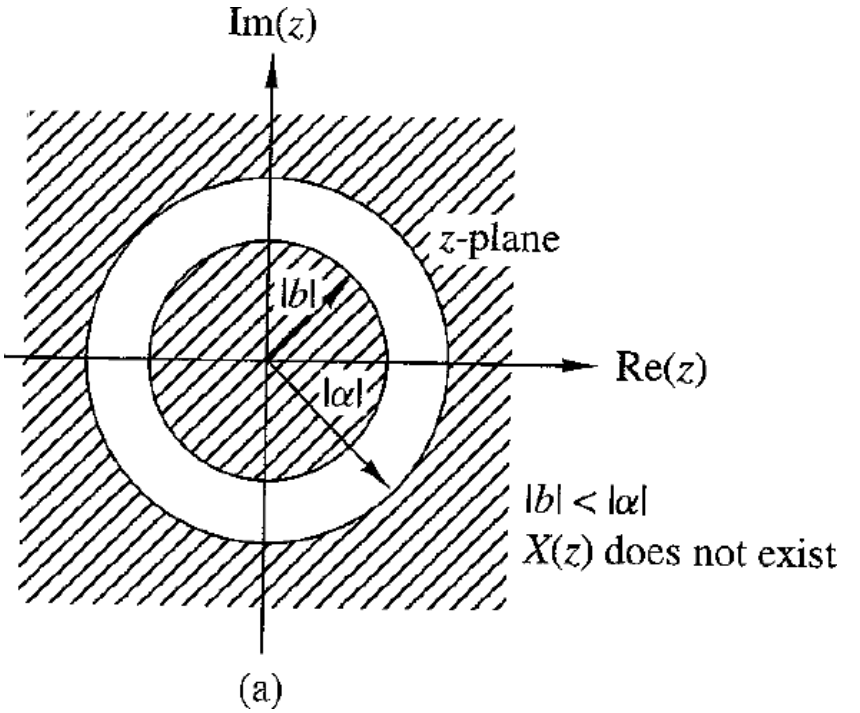
The first power series converges if $|\alpha z^{-1}| < 1$ or $|z| > |\alpha|$. The second power series converges if $|b^{-1} z| < 1$ or $|z| < |b|$.

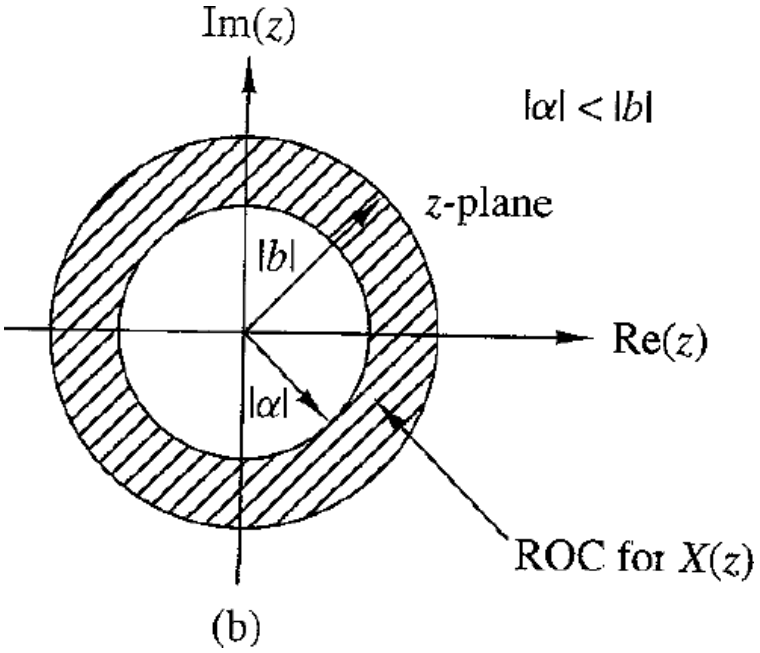
In determining the convergence of X(z), we consider two different cases:

Case 1 $|b| < |\alpha|$**:** In this case the two ROC above do not overlap, so X(z) does not exist.

Case 1 $|b| > |\alpha|$**:** In this case there is a ring in the z-plane where both power series converge simultaneously.

Example 3.1.5 (Prokis)-Cont.





$$X(z) = \frac{1}{1-\alpha z^{-1}} - \frac{1}{1-bz^{-1}}$$

$$= \frac{b-\alpha}{\alpha + b - z - \alpha b z^{-1}}$$

The ROC of $X(z)$ is $|\alpha| < |z| < |b|$.

Week 10
Slide 140-161

Properties of Z-Transform

Linearity

$$x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$$

$$x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$$

$$x(n) = a_1 x_1(n) + a_2 x_2(n) \overset{z}{\longleftrightarrow} X(z) = a_1 X_1(z) + a_2 X_2(z)$$

Examples 3.2.1 Determine the z-transform and the ROC of the signal

$$x(n) = [3(2^n) - 4(3^n)]u(n)$$

Solution:

If we define the signals: $x_1(n) = 2^n u(n)$ $x_2(n) = 3^n u(n)$

Then x(n) can be written as $x(n) = 3x_1(n) - 4x_2(n)$

According to linearity property, the z-transform is

$$X(z) = 3X_1(z) - 4X_2(z)$$

Properties of Z-Transform (Cont.)

We know from the Example 3.1.3,

$$\alpha^n u(n) \overset{z}{\longleftrightarrow} \frac{1}{1-\alpha z^{-1}}, \qquad \text{ROC: } |z| > |\alpha|$$

By setting $\alpha = 2$ and $\alpha = 3$, we obtain the z-transform of $x_1(n)$ and $x_2(n)$

$$x_1(n) = 2^n u(n) \overset{z}{\longleftrightarrow} X_1(z) = \frac{1}{1-2z^{-1}}, \qquad \text{ROC: } |z| > 2$$

$$x_2(n) = 3^n u(n) \overset{z}{\longleftrightarrow} X_2(z) = \frac{1}{1-3z^{-1}}, \qquad \text{ROC: } |z| > 3$$

The intersection of the ROC of $X_1(z)$ and $X_2(z)$ is $|z| > 3$. Thus the overall transform X(z) is

$$X(z) = \frac{3}{1-2z^{-1}} - \frac{4}{1-3z^{-1}}, \qquad \text{ROC: } |z| > 3$$

Examples 3.2.2 Determine the z-transform of the signals

(a) $x(n) = (\cos \omega_0 n)u(n)$

(b) $x(n) = (\sin \omega_0 n)u(n)$

Properties of Z-Transform (Cont.)

Solution (a): By using Eulers identity the signal x(n) can be expressed as

$$x(n) = (\cos \omega_0 n)u(n) = \frac{1}{2}e^{j\omega_0 n}u(n) + \frac{1}{2}e^{-j\omega_0 n}u(n)$$

Using linearity property of z-transform,

$$X(z) = \frac{1}{2}Z\{e^{j\omega_0 n}u(n)\} + \frac{1}{2}Z\{e^{-j\omega_0 n}u(n)\}$$

If we set $\alpha = e^{\pm j\omega_0}$ ($|\alpha| = |e^{\pm j\omega_0}| = 1$), we obtain

$$e^{j\omega_0 n}u(n) \overset{z}{\longleftrightarrow} \frac{1}{1 - e^{j\omega_0}z^{-1}}, \qquad \text{ROC: } |z| > 1$$

$$e^{-j\omega_0 n}u(n) \overset{z}{\longleftrightarrow} \frac{1}{1 - e^{-j\omega_0}z^{-1}}, \qquad \text{ROC: } |z| > 1$$

$$X(z) = \frac{1}{2}\frac{1}{1 - e^{j\omega_0}z^{-1}} + \frac{1}{2}\frac{1}{1 - e^{-j\omega_0}z^{-1}}, \qquad \text{ROC: } |z| > 1$$

After some algebraic manipulation-

$$(\cos \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{1 - z^{-1}\cos \omega_0}{1 - 2z^{-1}\cos \omega_0 + z^{-2}}, \qquad \text{ROC: } |z| > 1$$

(b) See the solution in book

Properties of Z-Transform (Cont.)

Time Shifting

If $x(n) \overset{z}{\longleftrightarrow} X(z)$

then $x(n-k) \overset{z}{\longleftrightarrow} z^{-k}X(z)$

The ROC of $z^{-k}X(z)$ is the same as that of X(z) except for z=0 if K>0, and Z=∞ if k<0.

See example 3.2.3 and 3.2.4

Scaling in z-domain

If $x(n) \overset{z}{\longleftrightarrow} X(z), \qquad \text{ROC: } r_1 < |z| < r_2$

then $a^n x(n) \overset{z}{\longleftrightarrow} X(a^{-1}z), \qquad \text{ROC: } |a|r_1 < |z| < |a|r_2$

For any constant 'a', real or complex.

Properties of Z-Transform (Cont.)

Determine the z-transforms of the signals

(a) $x(n) = a^n(\cos \omega_0 n)u(n)$

(b) $x(n) = a^n(\sin \omega_0 n)u(n)$

Solution (a): In example 3.2.2 (a) we have determined the z-transform of $(\cos \omega_0 n)\, u(n)$

$$(\cos \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{1 - z^{-1}\cos \omega_0}{1 - 2z^{-1}\cos \omega_0 + z^{-2}}, \qquad \text{ROC: } |z| > 1$$

Using time scaling property we get

$$a^n(\cos \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{1 - az^{-1}\cos \omega_0}{1 - 2az^{-1}\cos \omega_0 + a^2 z^{-2}}, \qquad |z| > |a|$$

Solution (b): In example 3.2.2 (b) we have determined the z-transform of $(\sin \omega_0 n)\, u(n)$

$$(\sin \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{z^{-1}\sin \omega_0}{1 - 2z^{-1}\cos \omega_0 + z^{-2}}, \qquad \text{ROC: } |z| > 1$$

Using time scaling property we get

$$a^n(\sin \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{az^{-1}\sin \omega_0}{1 - 2az^{-1}\cos \omega_0 + a^2 z^{-2}}, \qquad |z| > |a|$$

Properties of Z-Transform (Cont.)

Time Reversal

If $x(n) \overset{z}{\longleftrightarrow} X(z), \qquad \text{ROC: } r_1 < |z| < r_2$

then $x(-n) \overset{z}{\longleftrightarrow} X(z^{-1}), \qquad \text{ROC: } \frac{1}{r_2} < |z| < \frac{1}{r_1}$

Example 3.2.6: Determine the z-transform of the signal $x(n) = u(-n)$

In example 3.1.3 we have determined the z-transform of unit step signal u(n)

$$u(n) \overset{z}{\longleftrightarrow} \frac{1}{1 - z^{-1}}, \qquad \text{ROC: } |z| > 1$$

Using time reversal property we get

$$u(-n) \overset{z}{\longleftrightarrow} \frac{1}{1 - z}, \qquad \text{ROC: } |z| < 1$$

Properties of Z-Transform (Cont.)

Differentiation in z-domain

If $x(n) \overset{z}{\longleftrightarrow} X(z)$

then $nx(n) \overset{z}{\longleftrightarrow} -z\frac{dX(z)}{dz}$ **Both Transform have the same ROC**

Example 3.2.7: Determine the z-transform of the signal $x(n) = na^n u(n)$

The signal $x(n)$ can be expressed as $nx_1(n)$, where $x_1(n) = a^n u(n)$. In example 3.1.3, we have already determine z-transform of $a^n u(n)$.

$$x_1(n) = a^n u(n) \overset{z}{\longleftrightarrow} X_1(z) = \frac{1}{1 - az^{-1}}, \qquad \text{ROC: } |z| > |a|$$

Using differentiation in z-domain property we get

$$na^n u(n) \overset{z}{\longleftrightarrow} X(z) = -z\frac{dX_1(z)}{dz} = \frac{az^{-1}}{(1 - az^{-1})^2}, \qquad \text{ROC: } |z| > |a|$$

If we set a=1, we find the z transform of the unit ramp signal

$$nu(n) \overset{z}{\longleftrightarrow} \frac{z^{-1}}{(1 - z^{-1})^2}, \qquad \text{ROC: } |z| > 1$$

See Example 3.2.8 (Prokis)

Properties of Z-Transform (Cont.)

Convolution of two sequences

$$\text{If} \quad x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$$

$$x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$$

$$\text{then} \quad x(n) = x_1(n) * x_2(n) \overset{z}{\longleftrightarrow} X(z) = X_1(z)X_2(z)$$

The ROC of $X(z)$ is, at least, the intersection of that for $X_1(z)$ and $X_2(z)$.

Example 3.2.9: Compute the convolution $x(n)$ of the signals

$$x_1(n) = \{1, -2, 1\} \qquad x_2(n) = \begin{cases} 1, & 0 \le n \le 5 \\ 0, & \text{elsewhere} \end{cases}$$

The z-transform of $x_1(n)$ and $x_2(n)$

$$X_1(z) = 1 - 2z^{-1} + z^{-2}$$

$$X_2(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5}$$

Multiplication of $X_1(z)$ and $X_2(z)$

$$X(z) = X_1(z)X_2(z) = 1 - z^{-1} - z^{-6} + z^{-7} \qquad x(n) = \{\underset{\uparrow}{1}, -1, 0, 0, 0, 0, -1, 1\}$$

Some common z-transform Pair

TABLE 3.3 Some Common z-Transform Pairs

	Signal, $x(n)$	z-Transform, $X(z)$	ROC
1	$\delta(n)$	1	All z
2	$u(n)$	$\frac{1}{1-z^{-1}}$	$\lvert z\rvert > 1$
3	$a^n u(n)$	$\frac{1}{1-az^{-1}}$	$\lvert z\rvert > \lvert a\rvert$
4	$na^n u(n)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\lvert z\rvert > \lvert a\rvert$
5	$-a^n u(-n-1)$	$\frac{1}{1-az^{-1}}$	$\lvert z\rvert < \lvert a\rvert$
6	$-na^n u(-n-1)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\lvert z\rvert < \lvert a\rvert$
7	$(\cos \omega_0 n)u(n)$	$\frac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\lvert z\rvert > 1$
8	$(\sin \omega_0 n)u(n)$	$\frac{z^{-1}\sin\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\lvert z\rvert > 1$
9	$(a^n \cos \omega_0 n)u(n)$	$\frac{1-az^{-1}\cos\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\lvert z\rvert > \lvert a\rvert$
10	$(a^n \sin \omega_0 n)u(n)$	$\frac{az^{-1}\sin\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\lvert z\rvert > \lvert a\rvert$

Poles and Zeros

The zeros of a z-transform X(z) are the values of z for which X(z)=0.The poles of a z-transform are the values of z for which X(z)=∞. If X(z) is a rational function, then

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{a_0 + a_1 z^{-1} + \cdots + a_N z^{-N}} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}}$$

- ✓ We can express X(z) by poles –zeros plot in the complex plane, which shows the location of poles by cross (X) and location of zeros by circle (○)
- ✓ The multiplicity of multiple order poles or zeros is indicated by a number close to corresponding cross or circle.

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 z^{-M}}{a_0 z^{-N}} \frac{z^M + (b_1/b_0) z^{M-1} + \cdots + b_M/b_0}{z^N + (a_1/a_0) z^{N-1} + \cdots + a_N/a_0}$$

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0}{a_0} z^{-M+N} \frac{(z - z_1)(z - z_2) \cdots (z - z_M)}{(z - p_1)(z - p_2) \cdots (z - p_N)}$$

$$X(z) = G z^{N-M} \frac{\prod_{k=1}^{M} (z - z_k)}{\prod_{k=1}^{N} (z - p_k)}$$

- ✓ Obviously, by definition, the ROC of a z-transform should not contain any poles.

Pole-zero Plot

Example 3.3.1

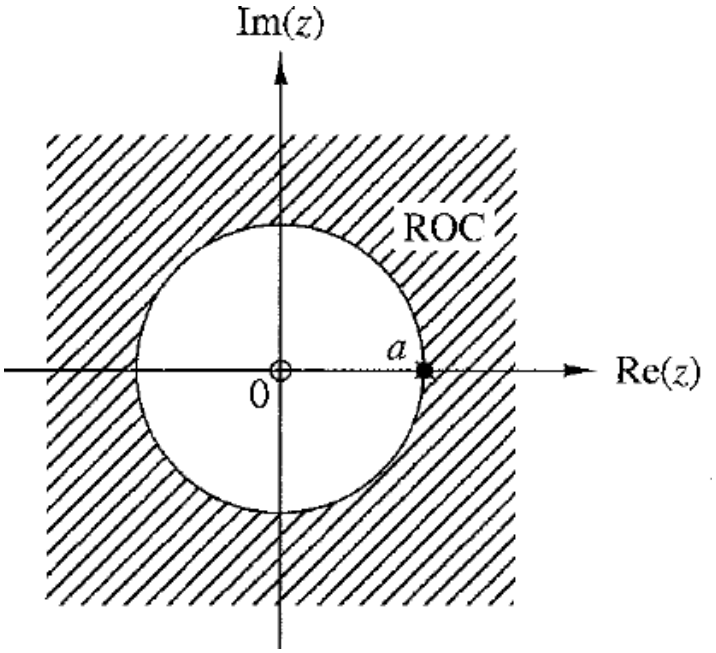
Determine the pole-zero plot for the signal

$$x(n) = a^n u(n), a > 0$$

The z-transform of the signal

$$X(z) = \frac{1}{1 - az^{-1}} = \frac{z}{z - a}, \qquad \text{ROC: } |z| > a$$

Thus X(z) has one zero at $z_1 = 0$ and one pole at $p_1 = a$. Note that the pole $p_1 = a$ is not included in the ROC since the z transform does not converge at a pole.



Pole-zero Plot

Example 3.3.2

Determine the pole-zero plot for the signal

$$x(n) = \begin{cases} a^n, & 0 \le n \le M-1 \\ 0, & elsewhere \end{cases}$$

Where, a>0

The z-transform of the signal

$$X(z) = \sum_{n=0}^{M-1} (az^{-1})^n = \frac{1-(az^{-1})^M}{1-az^{-1}} = \frac{z^M - a^M}{z^{M-1}(z-a)}$$

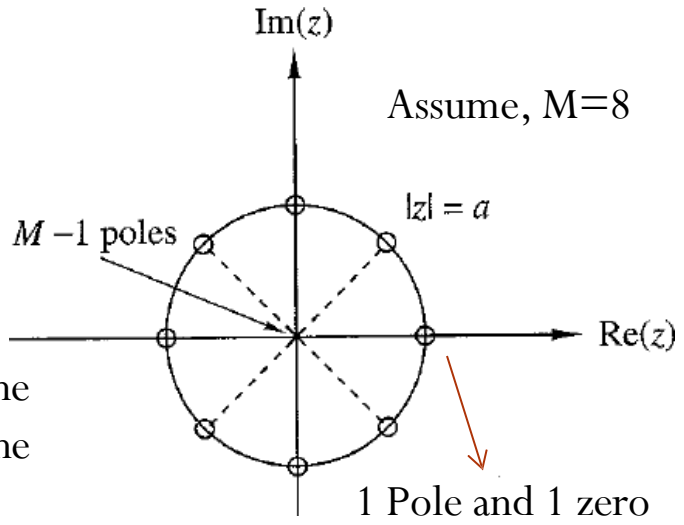
Since a>0, the equation $z^M = a^M$ has M roots at

$$z_k = ae^{j2\pi k/M} \qquad k = 0, 1, \ldots, M-1$$

The zero $z_0 = a$ cancels the pole at z=a.Thus

$$X(z) = \frac{(z-z_1)(z-z_2)\cdots(z-z_{M-1})}{z^{M-1}}$$

Which has M-1 poles and M-1 zeros. Note that the ROC is the enter z-plane except z=0 because of the M-1 poles located at the origin.



Pole location and Time-Domain Behavior for Causal Signals

A causal real signal with a single real pole in z-transform

$$x(n) = a^n u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{1 - az^{-1}}, \qquad \text{ROC: } |z| > |a|$$

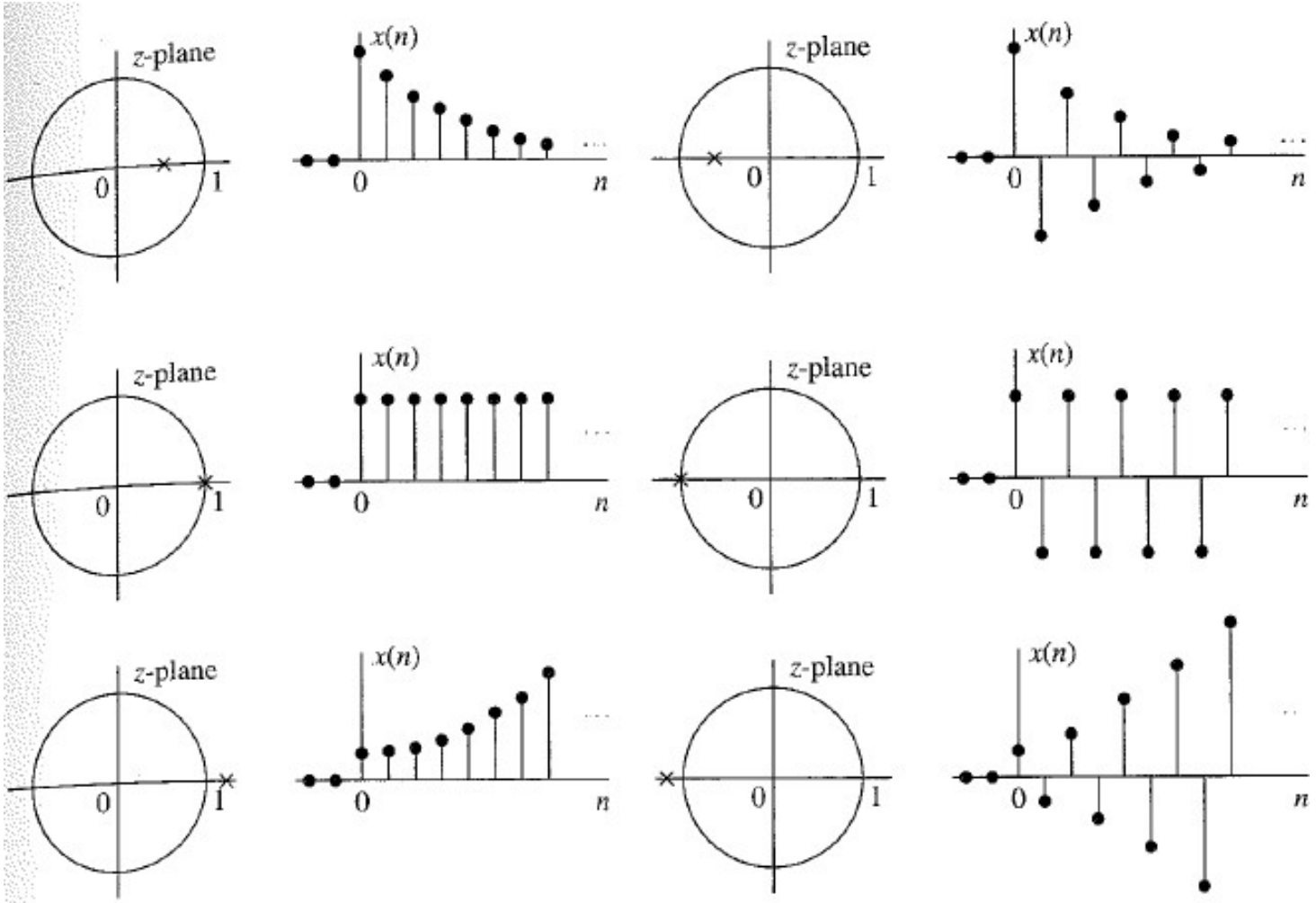


Figure 3.3.5 Time-domain behavior of a single-real-pole causal signal as a function of the location of the pole with respect to the unit circle.

Pole location and Time-Domain Behavior for Causal Signals (Cont.)

$$na^n u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{az^{-1}}{(1-az^{-1})^2}, \quad \text{ROC: } |z| > |a|$$

A causal real signal with double real pole in z-transform

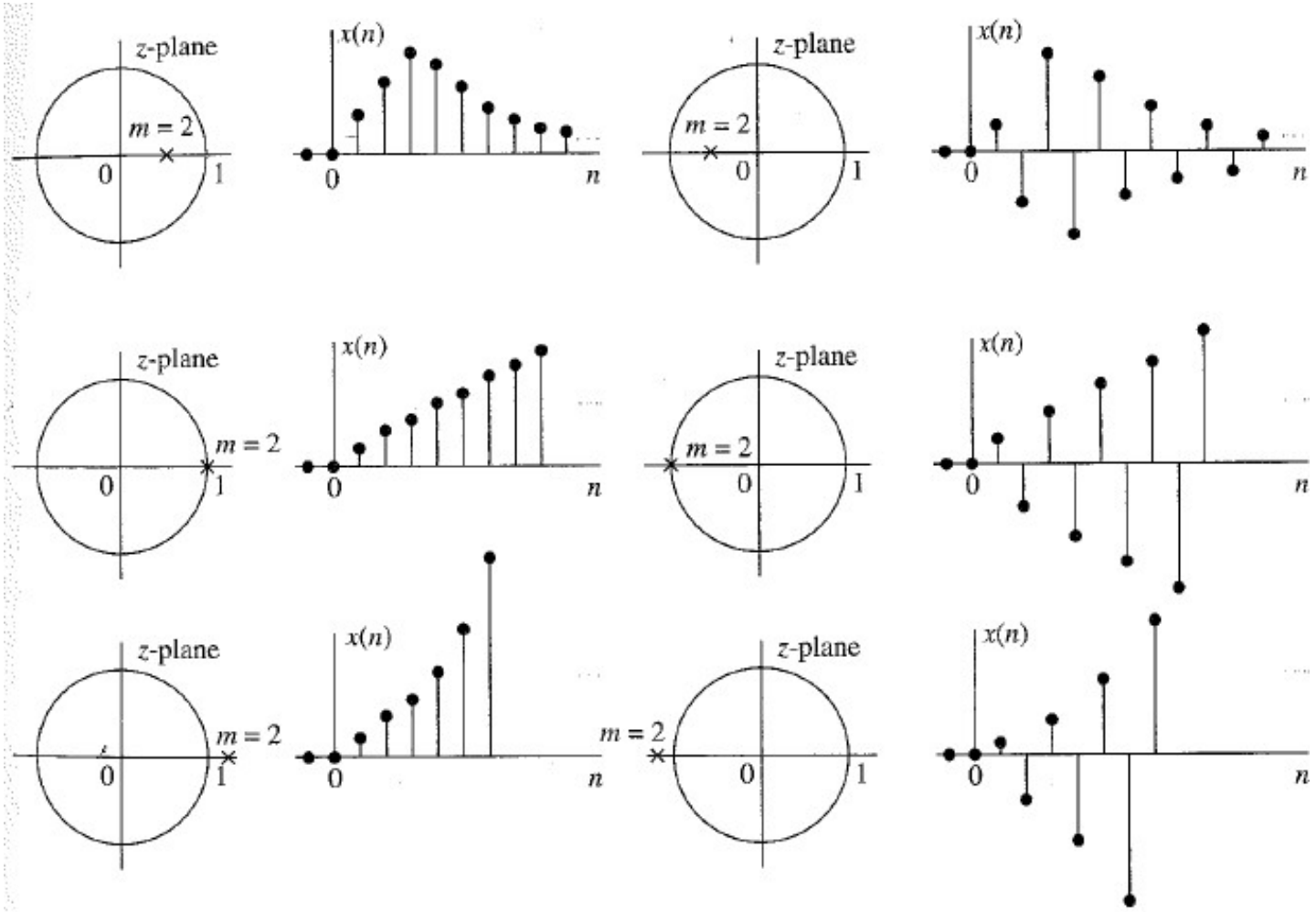


Figure 3.3.6 Time-domain behavior of causal signals corresponding to a double ($m = 2$) real pole, as a function of the pole location.

Pole location and Time-Domain Behavior for Causal Signals (Cont.)

A causal real signal with double complex-conjugate pole in z-transform

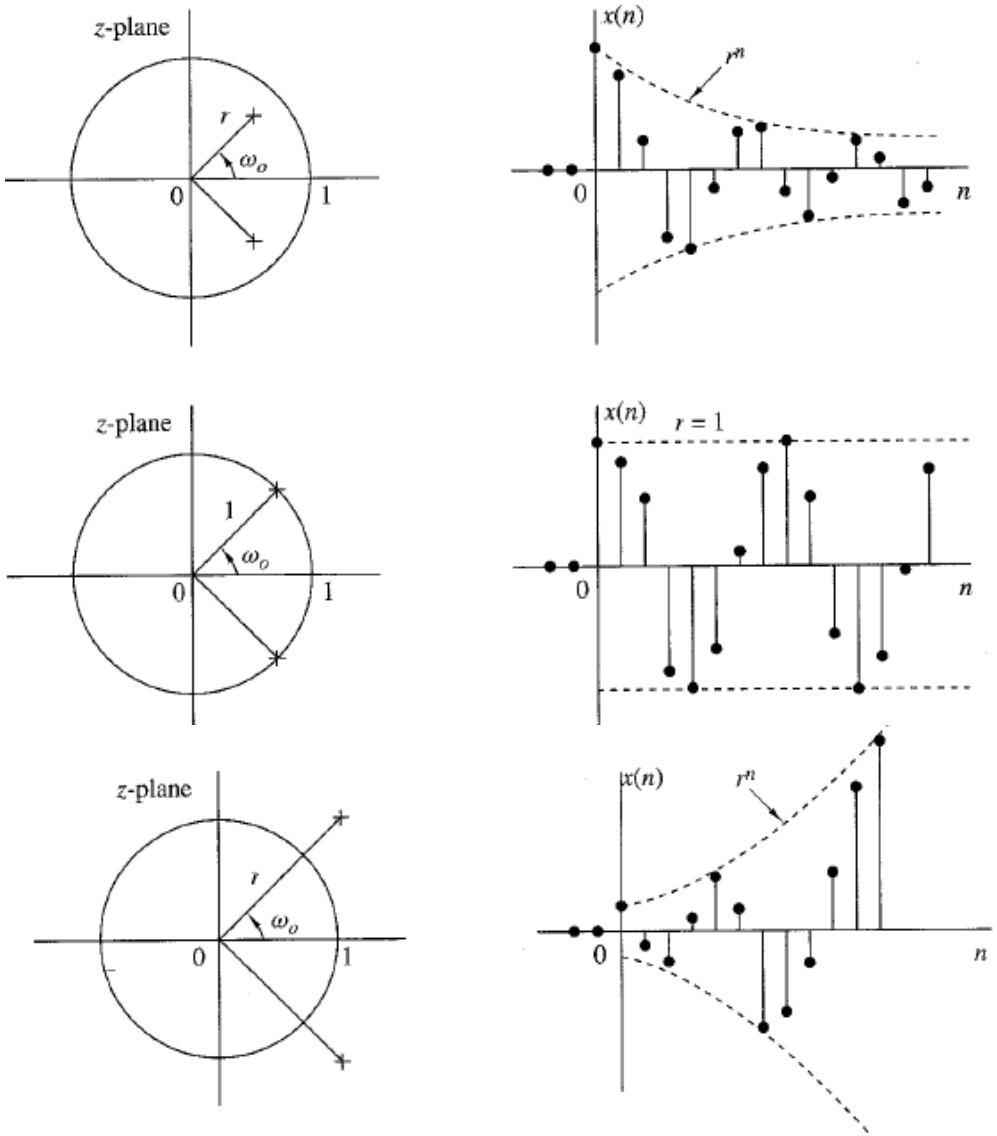


Figure 3.3.7 A pair of complex-conjugate poles corresponds to causal signals with oscillatory behavior.

Inversion of the z-Transform

There are three methods for evaluating the inverse z-Transform

1) Direct evaluation by contour integration.
2) Expansion into a series of terms, in the variables $z^{-1}\ and\ z$.
3) Partial fraction expansion and table lookup

Inverse z-Transform by Power Series Expansion

EXAMPLE 3.4.2

Determine the inverse z-transform of

$$X(z) = \frac{1}{1 - 1.5z^{-1} + 0.5z^{-2}}$$

when

(a) ROC: $|z| > 1$

(b) ROC: $|z| < 0.5$

Since the ROC is the exterior of a circle, we expect x(n) to be a causal signal. Thus we seek a power series expansion in negative power of z.

$$X(z) = \frac{1}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

$$= \frac{(1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}) + \frac{3}{2}z^{-1} - \frac{1}{2}z^{-2}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

$$= 1 + \frac{\frac{3}{2}z^{-1} - \frac{1}{2}z^{-2}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

Inverse z-Transform by Power Series Expansion (Cont.)

$$= 1 + \frac{\frac{3}{2}z^{-1}\left(1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}\right) - \frac{1}{2}z^{-2} + \frac{9}{4}z^{-2} - \frac{3}{4}z^{-3}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}} = 1 + \frac{3}{2}z^{-1} + \frac{\frac{7}{4}z^{-2} - \frac{3}{4}z^{-3}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

$$= 1 + \frac{3}{2}z^{-1} + \frac{\frac{7}{4}z^{-2}\left(1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}\right) - \frac{3}{4}z^{-3} + \frac{21}{8}z^{-3} - \frac{7}{8}z^{-4}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

$$= 1 + \frac{3}{2}z^{-1} + \frac{7}{4}z^{-2} + \frac{\frac{15}{8}z^{-3} - \frac{7}{8}z^{-4}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

$$= 1 + \frac{3}{2}z^{-1} + \frac{7}{4}z^{-2} + \frac{15}{8}z^{-3} + \frac{-\frac{7}{8}z^{-4} + \frac{45}{16}z^{-4} - \frac{15}{16}z^{-5}}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}}$$

$$= 1 + \frac{3}{2}z^{-1} + \frac{7}{4}z^{-2} + \frac{15}{8}z^{-3} + \frac{31}{16}z^{-4} + \ldots\ldots$$

$$x(n) = \{1, \underset{\uparrow}{\frac{3}{2}}, \frac{7}{4}, \frac{15}{8}, \frac{31}{16}, \ldots\}$$

Inverse z-Transform by Power Series Expansion (Cont.)

(b) In this case the ROC is the interior of circle. Consequently this signal x(n) is anticausal. To obtain a power series expansion in positive powers of z, we perform the long division in the following way

$$
\begin{array}{r}
2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \cdots \\
\hline
\tfrac{1}{2}z^{-2} - \tfrac{3}{2}z^{-1} + 1 \;\big)\, 1 \\
\underline{1 - 3z + 2z^2} \\
3z - 2z^2 \\
\underline{3z - 9z^2 + 6z^3} \\
7z^2 - 6z^3 \\
\underline{7z^2 - 21z^3 + 14z^4} \\
15z^3 - 14z^4 \\
\underline{15z^3 - 45z^4 + 30z^5} \\
31z^4 - 30z^5
\end{array}
$$

$$
X(z) = \frac{1}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}} = 2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \cdots
$$

$$
x(n) = \{\cdots 62, 30, 14, 6, 2, 0, 0\}
\qquad
\uparrow
$$

Inverse z-Transform by Partial-Fraction Expansion

Example 3.4.8: Determine the inverse z-transform of

$$X(z) = \frac{1}{1 - 1.5z^{-1} + 0.5z^{-2}}$$

if **(a)** ROC: $|z| > 1$
(b) ROC: $|z| < 0.5$
(c) ROC: $0.5 < |z| < 1$

Solution:

$$X(z) = \frac{z^2}{z^2 - 1.5z + 0.5}$$

$$\frac{X(z)}{z} = \frac{z}{(z-1)(z-0.5)} = \frac{A_1}{z-1} + \frac{A_2}{z-0.5}$$

$$z = (z - 0.5)A_1 + (z-1)A_2$$

Setting z=1, in the above equation, we get

$$1 = (1 - 0.5)A_1 \qquad \text{So, } A_1 = 2$$

Setting z=0.5, in the above equation, we get

$$0.5 = (0.5 - 1)A_2 \qquad \text{So, } A_2 = -1$$

Inverse z-Transform by Partial-Fraction Expansion (Cont.)

$$\frac{X(z)}{z} = \frac{2}{z-1} - \frac{1}{z-0.5} \qquad X(z) = \frac{2}{1-z^{-1}} - \frac{1}{1-0.5z^{-1}}$$

I.(a) ROC $|z| > 1$, the signal x(n) is causal, both term of above equation will be causal terms

$$x(n) = 2(1)^n u(n) - 0.5^n u(n) = (2 - 0.5^n)\,u(n)$$

(b) ROC $|z| < 0.5$, the signal x(n) is anticausal, both term of above equation will be anticausal terms:

$$x(n) = -2(1)^n u(-n-1) + 0.5^n u(-n-1) = (0.5^n - 2)u(-n-1)$$

(c) ROC $0.5 < |z| < 1$, which implies that the signal x(n) is two sided.

Thus one of the terms corresponds to a causal signal and the other to an anticausal signal.

Obviously, the ROC is the overlapping of the region $|z| > 0.5$ and $|z| < 1$. Hence the pole 0.5 provides the causal part and pole 1 anticausal part.

$$x(n) = -2(1)^n u(-n-1) - 0.5^n u(n)$$

Inverse z-Transform by Partial-Fraction Expansion (Cont.)

Practice Problem:

Example 3.4.9: Determine the causal signal x(n) whose z-transform is given by

$$x(z) = \frac{1 + z^{-1}}{1 - z^{-1} + 0.5z^{-2}}$$

Example 3.4.10: Determine the causal signal x(n) having the z-transform

$$x(z) = \frac{1}{(1 + z^{-1})(1 - z^{-1})^2}$$

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-5
Implementation of Discrete Time System

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ FIR System
- ✓ Structures for FIR System
- ✓ Direct form realization
- ✓ Cascade form realization
- ✓ Examples related to FIR system implementation
- ✓ IIR system
- ✓ Structures for FIR System
- ✓ Direct form structures of IIR system
- ✓ Cascade and Parallel form realization
- ✓ Examples related to IIR system implementation

Reference Book:

Digital Signal Processing (4th Edition), John G. Proakis, Dimitris K Manolakis

Chapter 9 (Implementation of Discrete Time System)

Week 11
Slide 164-192

Chapter Intended Learning Outcomes

(i)Ability to implement finite impulse response (FIR) and infinite impulse response (IIR) systems using different structures in terms of block diagram (or signal flow graph).

(ii)Ability to determine the system transfer function and difference equation given the corresponding block diagram (or signal flow graph) representation.

Finite Impulse Response (FIR) System

In signal processing, a finite impulse response (FIR) system is a system whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time.

$$h(n) = \begin{matrix} b_n, & 0 \le n \le M-1 \\ 0, & otherwise \end{matrix}$$

where M is some positive integer. This is called a finite impulse response (FIR) system because the non-zero part of the impulse response (b_k) is finite in extent. Because of that property, the convolution sum becomes a finite sum,

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k)$$

Structures for FIR System

In general, FIR system is described by the difference equation

$$y(n) = \sum_{k=0}^{M-1} b_k\, x(n-k)$$

Or, equivalently, by the system function

$$H(z) = \sum_{k=0}^{M-1} b_k\, z^{-k}$$

Where the coefficient $\{b_k\}$, is identical to the unit sample (impulse) response of the FIR system, that is

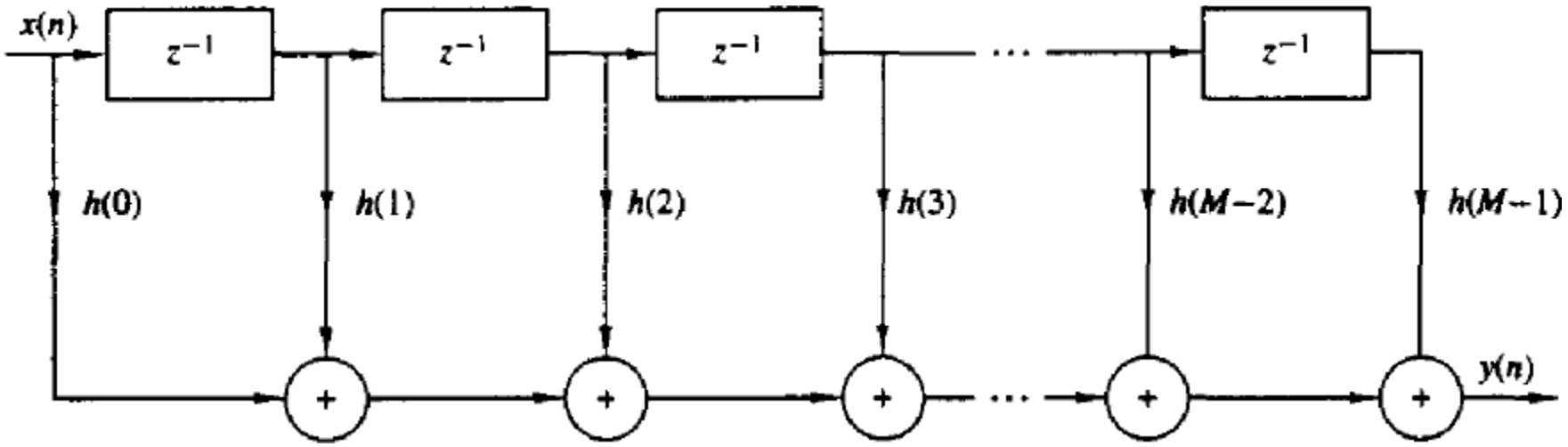
$$h(n) = \begin{cases} b_n, & 0 \le n \le M-1 \\ 0, & otherwise \end{cases}$$

Methods for Implementing FIR System

1) Direct form
2) Cascade form realization
3) Frequency sampling realization
4) Lattice Realization

Structures of FIR System: Direct form realization

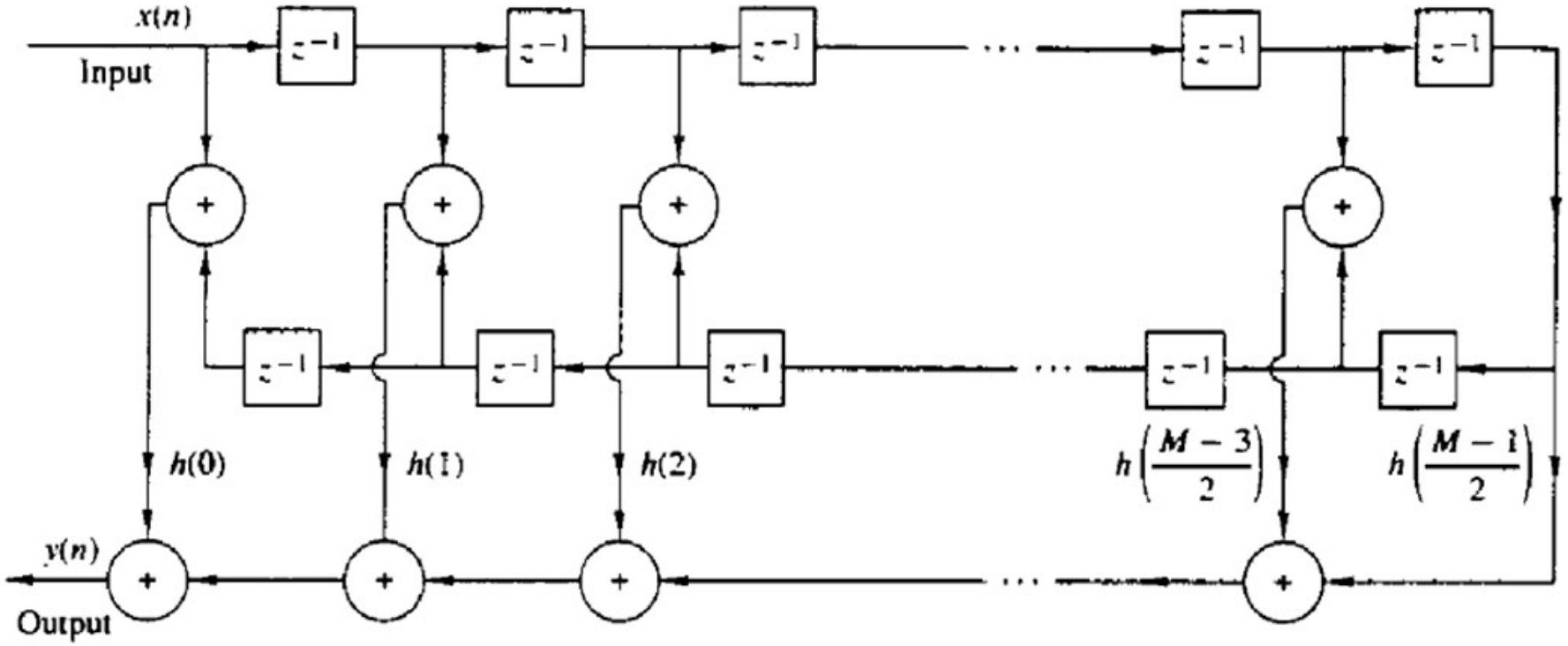
Direct form Realization



- ✓ This structure requires M-1 memory locations for storing the M-1 previous inputs and has a complexity of M multiplications and M-1 additions per output points.
- ✓ The direct-form realization is often called a transversal or tapped-delay-line filter.

Structures of FIR System: Direct form realization

Direct form realization of linear-phase FIR System



- ✓ When the FIR system has linear phase, the unit sample response of the system satisfies either the symmetry or asymmetry condition $h(n) = \pm h(M-1-n)$
- ✓ For such as system the number of multiplications is reduced from M to M/2 for M even and to (M-1)/2 for M odd.

Problems on Direct form realization of FIR System

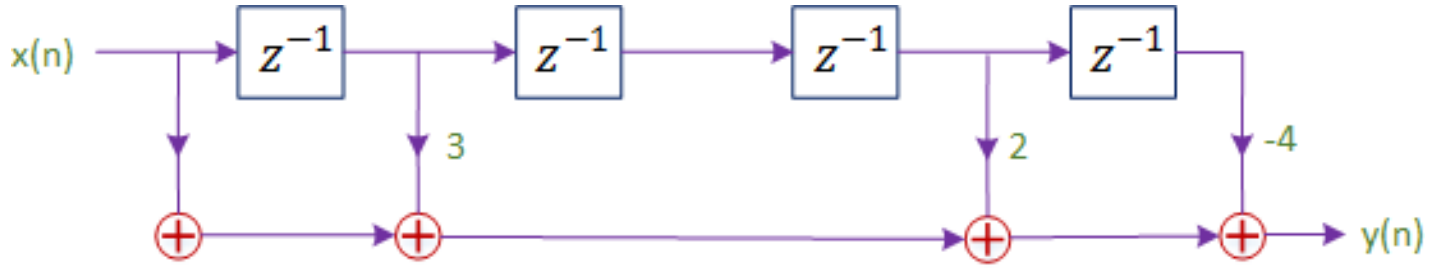
Problem: Consider an FIR system with system function

(a) $H(z) = 1 + 3z^{-1} + 2z^{-3} - 4z^{-4}$

(b) $H(z) = \frac{1}{4} + \frac{1}{2}z^{-1} + \frac{3}{4}z^{-2} + \frac{1}{2}z^{-3} + \frac{1}{4}z^{-4}$

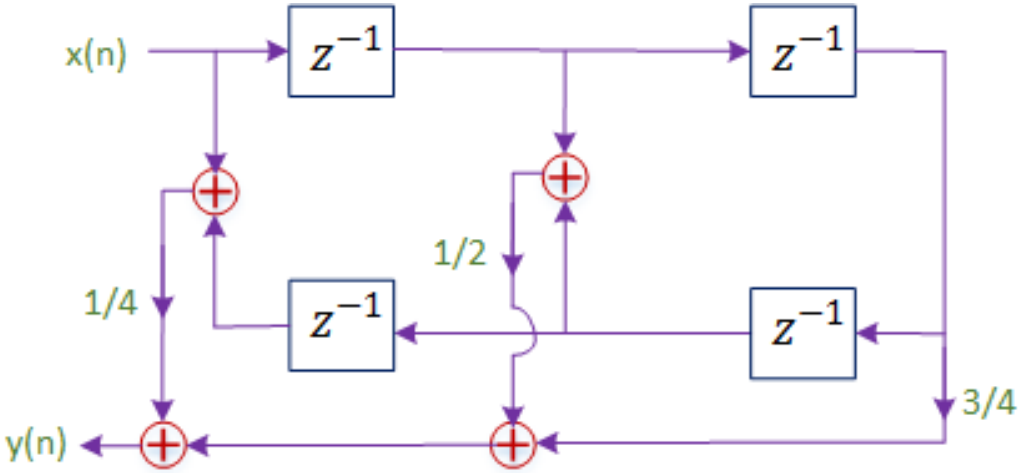
Sketch the direct form realization of the system.

(a)



(b) The system is a linear phase FIR system. It can be expressed as

$$H(z) = \frac{1}{4}(1 + z^{-4}) + \frac{1}{2}(z^{-1}+z^{-3}) + \frac{3}{4}z^{-2}$$



Problems on Direct form realization of FIR System

Problem:

(1) Determine a direct form realization for the following linear phase discrete time system

(a) $$h(n) = \underset{\uparrow}{\{1,2,3,4,3,2,1\}}$$

(b) $$h(n) = \underset{\uparrow}{\{1,2,3,3,2,1\}}$$

(2) Consider an FIR system with system function

$$H(z) = 1 + 2.88z^{-1} + 3.4048z^{-2} + 1.74z^{-3} + 0.4z^{-4}$$

Sketch the direct form realization of the system. How many additions and multiplications instructions are required per output point? Also determine the number of memory block.

Structures of FIR System: Cascade form realization

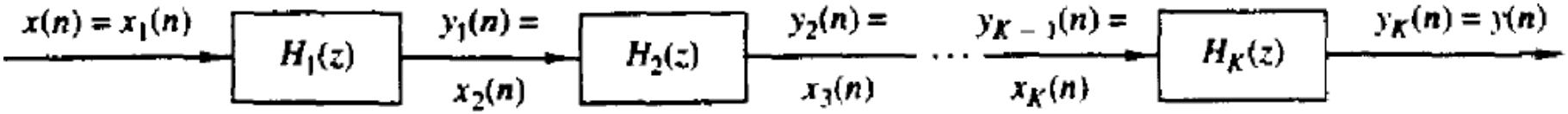
Cascade form structures of FIR System

In cascade realization of FIR system H(z) is factorized into second order FIR system so that

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} = \prod_{k=1}^{K} H_k(z)$$

where $H_k(z) = b_{k0} + b_{k1} z^{-1} + b_{k2} z^{-2} \qquad k = 1, 2, \ldots, K$

and k is the integer part of (M+1/2).



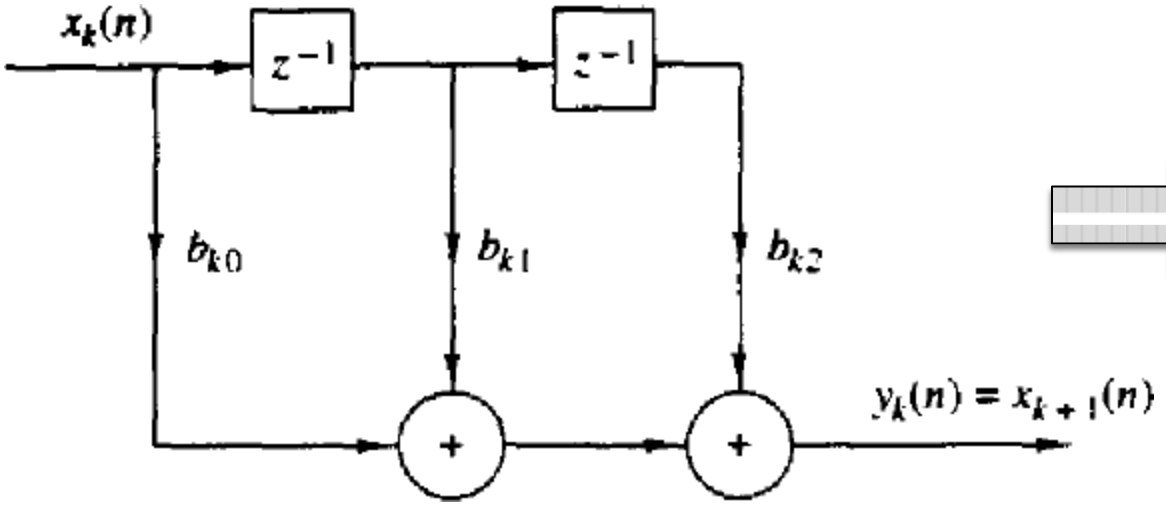
Structures of FIR System: Cascade form realization

Why second-order polynomial instead of first-order polynomial?

The zeros of H(z) are grouped in pairs to produce the second-order FIR systems of the form

$$H_k(z) = b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2} \qquad k = 1, 2, \ldots, K$$

It is always desirable to form pairs of complex-conjugate roots so that the coefficients $\{b_{ki}\}$ in the second order subsystems are real valued. By this way we can avoid the complex multiplications. On the other hand, real-valued roots can be paired in any arbitrary manner. The cascade-form realization along with the basic second-order section is shown below:



This is the basic building block to implement cascade form FIR structures

Example on Cascade form realization of FIR System

Example: Determine a cascade form realization for the following discrete time system with system function $H(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4}$

Solution: To factorize H(z), we use the MATLAB command roots([1 1 1 1 1]) to solve for the roots (or use calculator to find the roots of the polynomial H(z)):

```
 0.3090 + 0.9511i
 0.3090 - 0.9511i
-0.8090 + 0.5878i
-0.8090 - 0.5878i
```

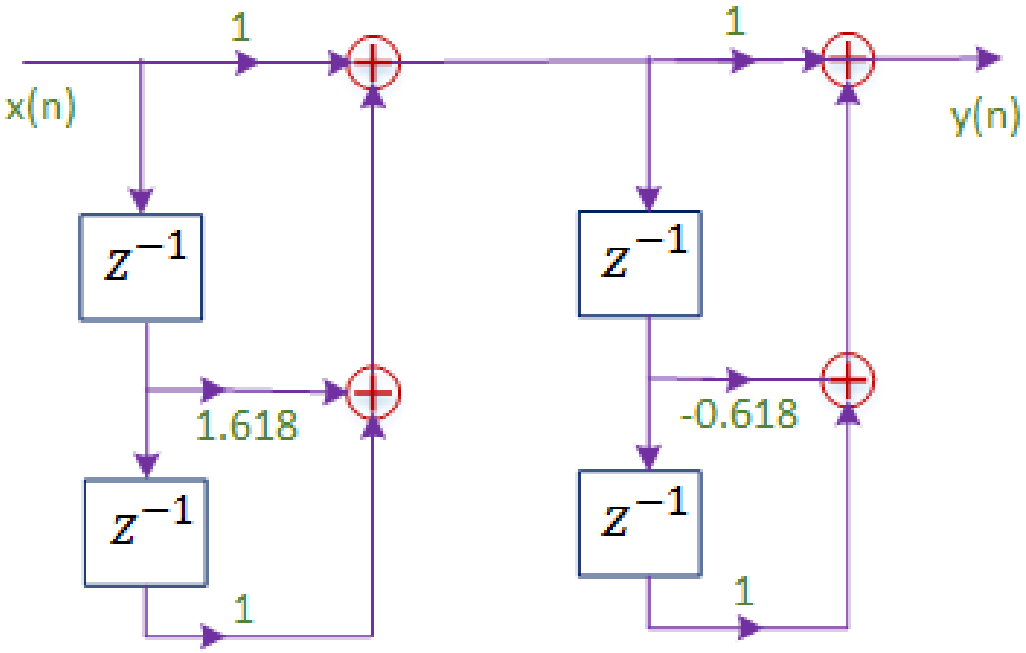
$$H(z) = \left(1 - [0.309 + j0.9511]z^{-1}\right)\left(1 - [0.309 - j0.9511]z^{-1}\right) \times \left(1 - [-0.809 + j0.5878]z^{-1}\right)\left(1 - [-0.809 - j0.5878]z^{-1}\right)$$

To get the second order subsystem with real-valued coefficients, we group the sections of complex conjugates together

$$\begin{aligned} H(z) &= \left(1 - 0.618z^{-1} + z^{-2}\right)\left(1 + 1.618z^{-1} + z^{-2}\right) \\ &= \left(1 + 1.618z^{-1} + z^{-2}\right)\left(1 - 0.618z^{-1} + z^{-2}\right) \end{aligned}$$

Example on Cascade form realization of FIR System (Cont.)

$$H(z) = \left(1 + 1.618z^{-1} + z^{-2}\right)\left(1 - 0.618z^{-1} + z^{-2}\right)$$



IIR System

If the impulse response of a system is infinite, the system is called IIR system.

The impulse response is "infinite" because there is feedback in the filter; if you put in an impulse (a single "1" sample followed by many "0" samples), an infinite number of non-zero values will come out (theoretically.)

In general, IIR system is described by the difference equation

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k)$$

Or, equivalently, by the system function

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

Methods for Implementing IIR System

1) Direct form (Direct form I and Direct form II)
2) Cascade form realization
3) Parallel form realization
4) Lattice structures
5) Lattice-ladder structures

Structures for IIR System: Direct form I

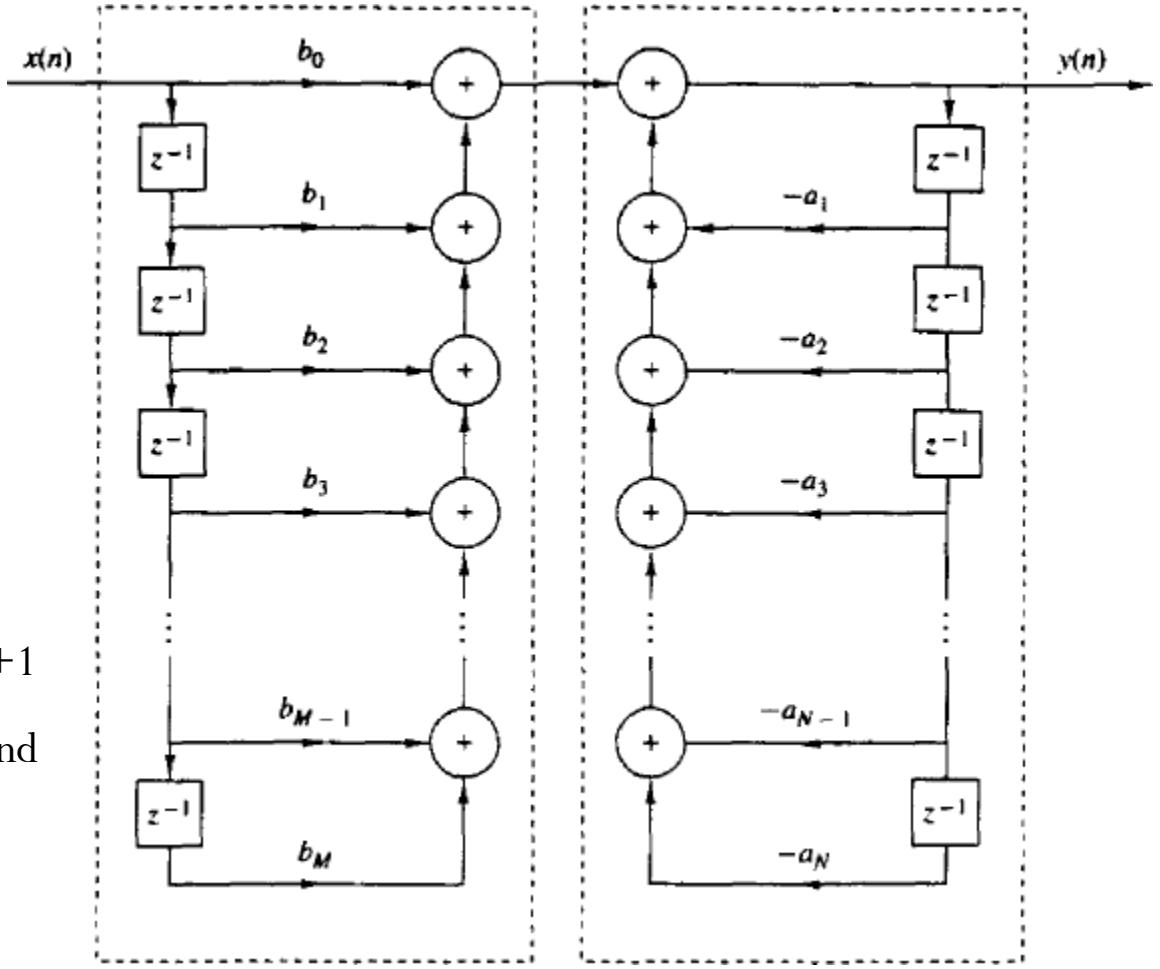
Direct form structures (Direct form I realization)

$$H(z) = H_1(z)H_2(z)$$

$$H_1(z) = \sum_{k=0}^{M} b_k z^{-k}$$

$$H_2(z) = \frac{1}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

This structure requires M+N+1 multiplications, M+N additions and M+N+1 memory locations.



Structures for IIR System: Direct form II

Direct form structures (Direct form II realization)/Canonic Form

Let us consider a first order system

$$y(n) = -a_1 y(n-1) + b_0 x(n) + b_1 x(n-1)$$

The nonrecursive part of the system

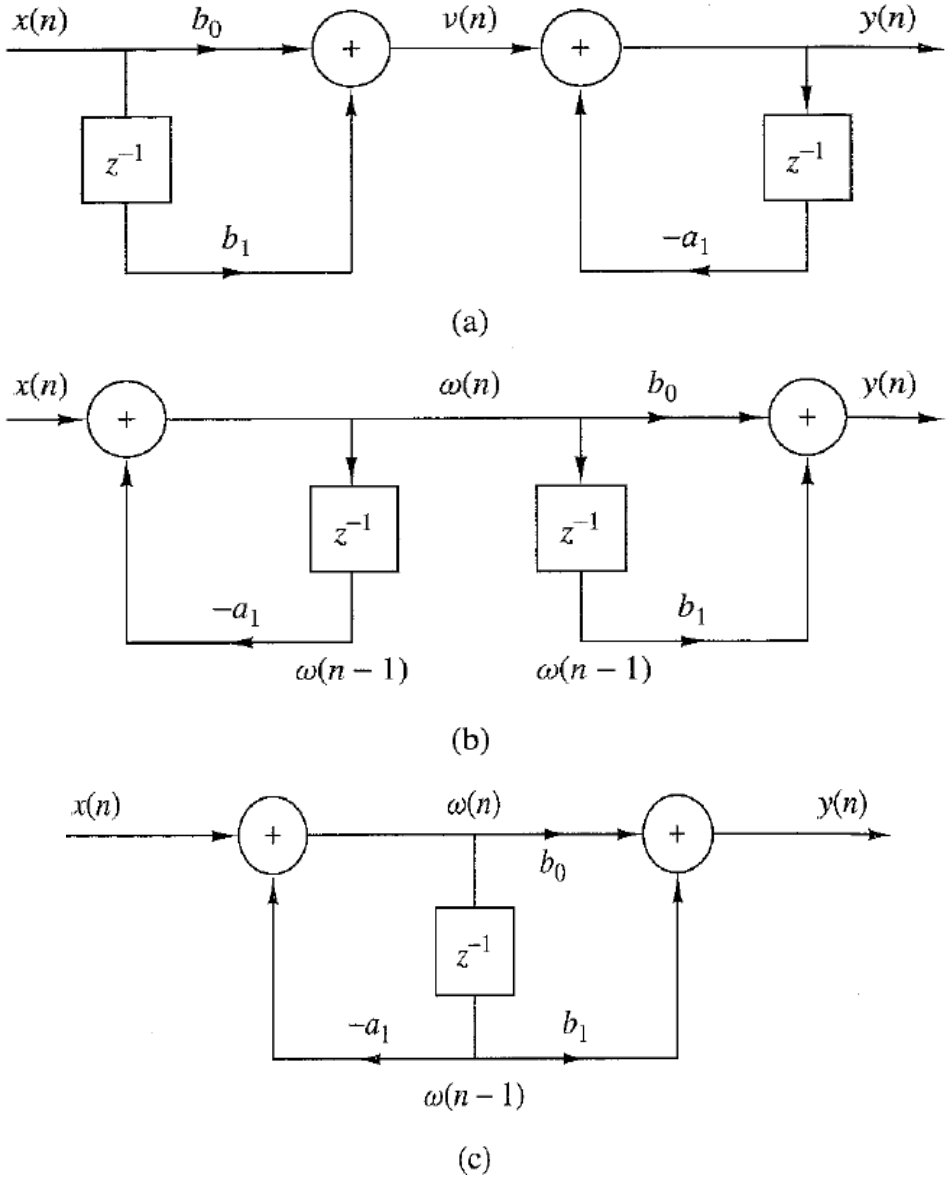
$$v(n) = b_0 x(n) + b_1 x(n-1)$$

The recursive part of the system

$$y(n) = -a_1 y(n-1) + v(n)$$

However, it is observed that if we interchange the order of the cascaded LTI system, the overall system response remain the same. Thus if we interchange the order of the recursive and nonrecursive system, we get

$$w(n) = -a_1 w(n-1) + x(n)$$
$$y(n) = b_0 w(n) + b_1 w(n-1)$$



Structures for IIR System: Direct form II

Direct form structures (Direct form II realization)/ Canonic Form

The difference equation to describe the general IIR system can be written as:

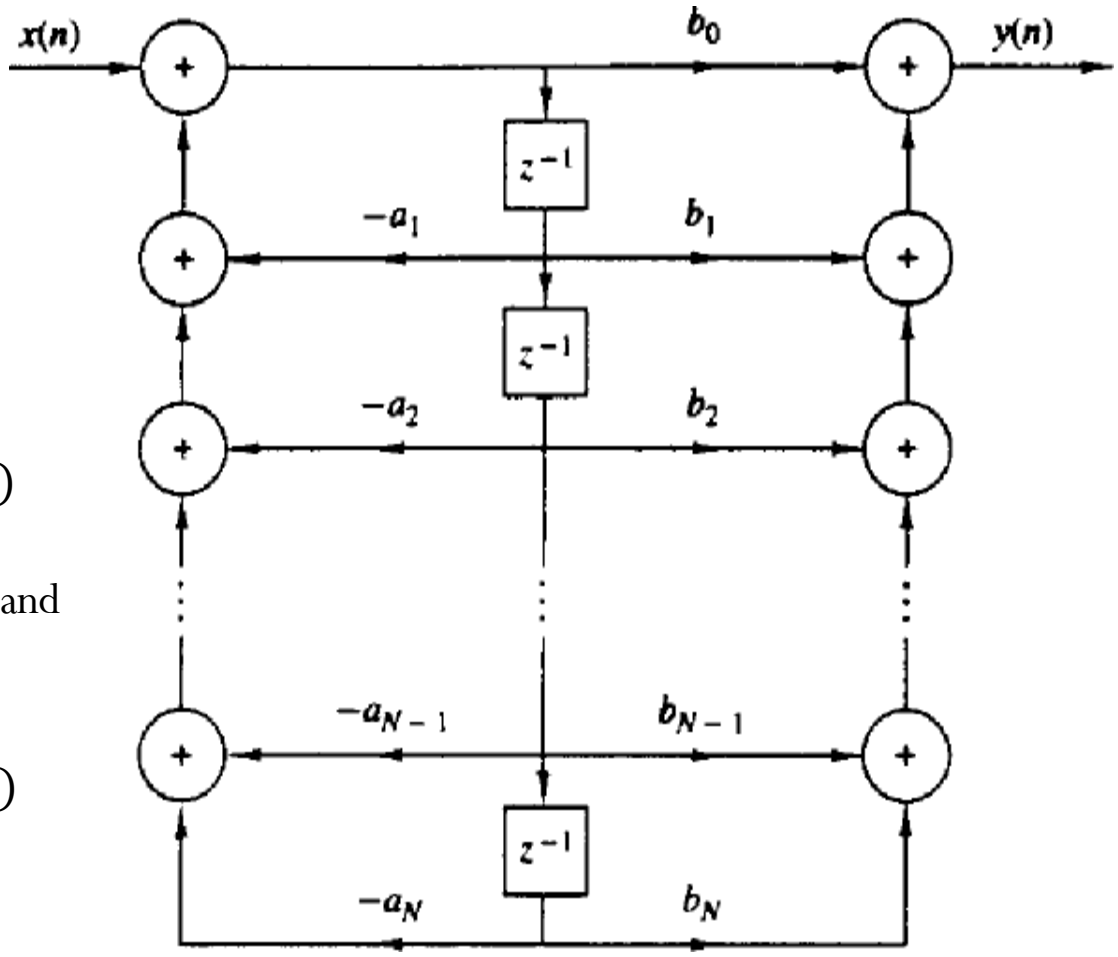
$$V(n) = \sum_{k=0}^{M} b_k x(n-k)$$

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + V(n)$$

Interchanging the order of recursive and nonrecursive part we can write

$$w(n) = -\sum_{k=1}^{N} a_k w(n-k) + x(n)$$

$$y(n) = \sum_{k=0}^{M} b_k w(n-k)$$



Direct form II (considering N=M)

Multiplications and additions are same as direct form I. But the required memory location is {M,N}

Structures for IIR System: Problem

Problem: For the following system

$$y(n) - \frac{3}{4}y(n-1) + \frac{1}{8}y(n-2) = x(n) + \frac{1}{3}x(n-1)$$

(a) Determine its system function.

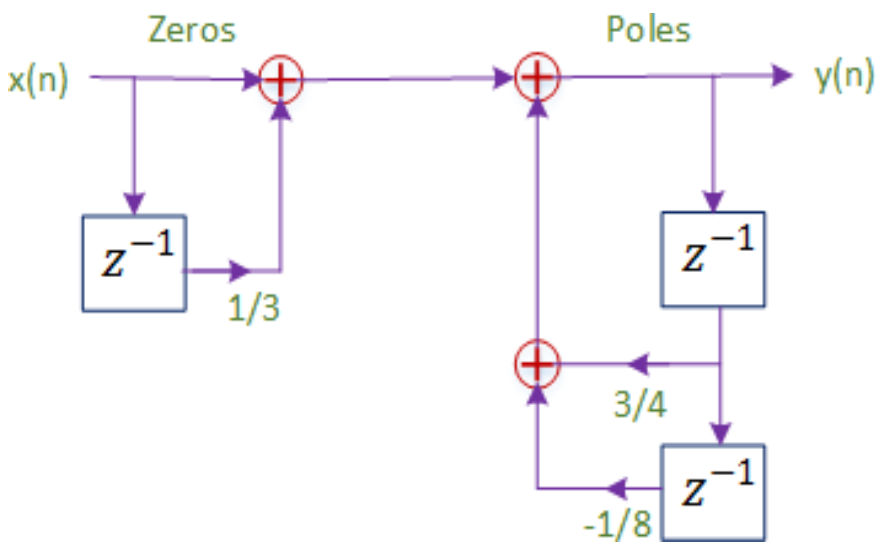
(b) Obtained the direct form I and direct form II structures.

Solution:

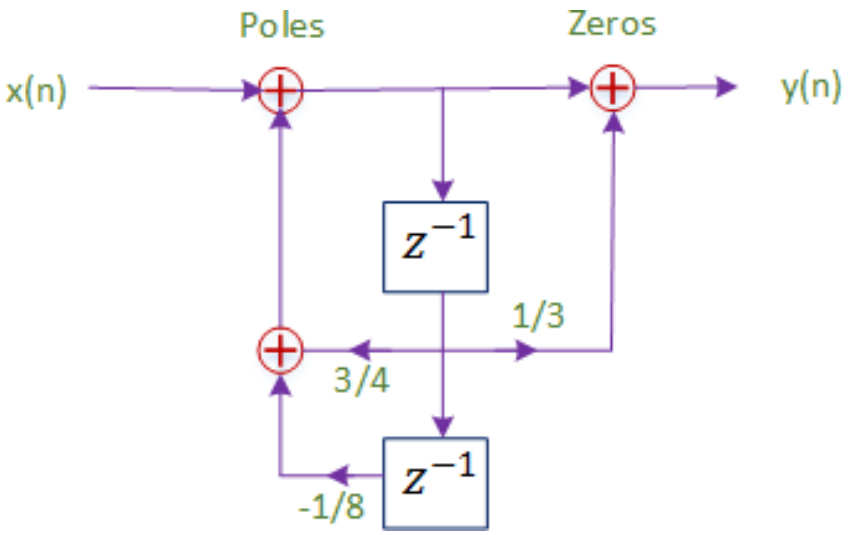
a)

$$H(z) = \frac{1 + \frac{1}{3}z^{-1}}{1 - \frac{3}{4}z^{-1} + \frac{1}{8}z^{-2}}$$

b)



Direct form I



Direct form II

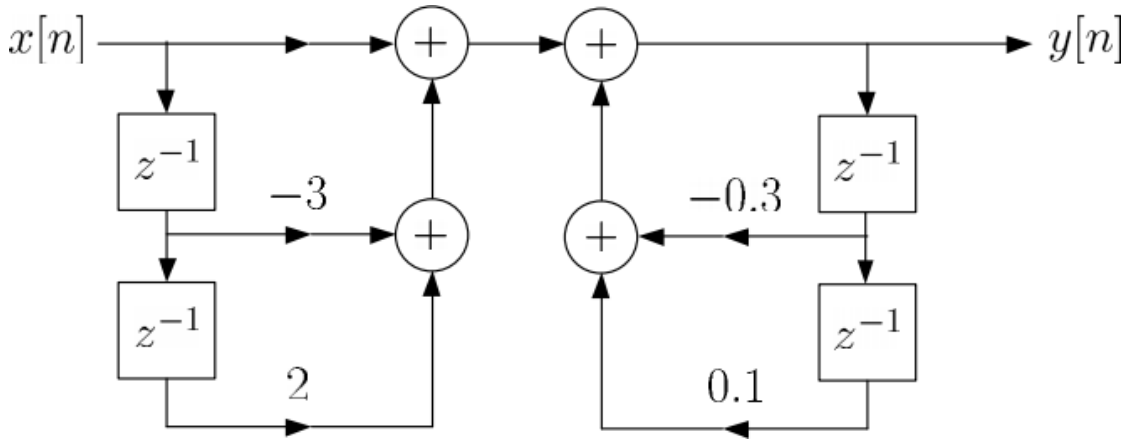
Structures for IIR System: Problem

Problem: Draw the block diagrams using the direct form I and canonic forms for the LTI system whose transfer function is:

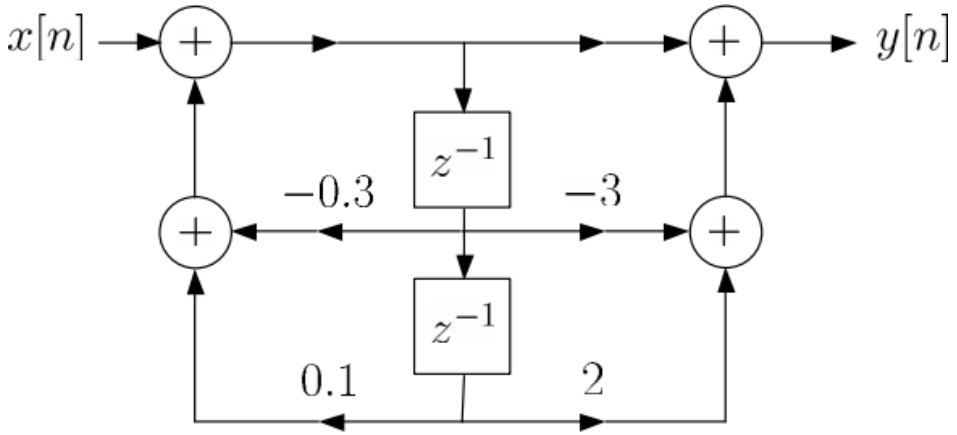
$$H(z) = \frac{1 - 3z^{-1} + 2z^{-2}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

Solution:

Direct form I



Canonic form/ Direct form II



Structures for IIR System: Cascade form structures

Cascade form structures

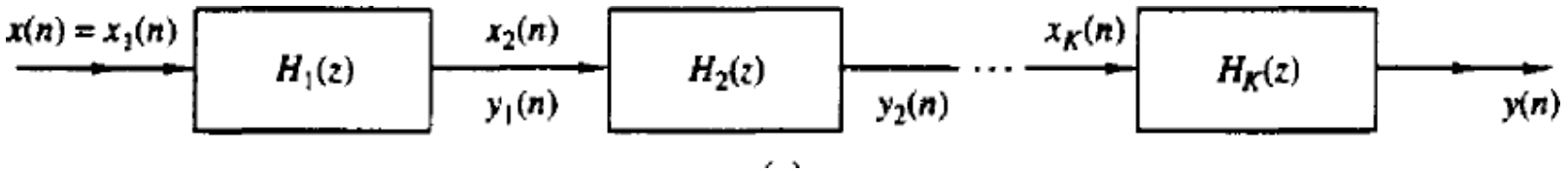
We can factorize the numerator and denominator polynomials of IIR system function in terms of second-order polynomial system functions as:

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} = \prod_{k=1}^{K} H_k(z)$$

Without the loss of generality we assume that $N \geq M$

Where k is the integer part of (N+1)/2. $H_k(z)$ has the general form

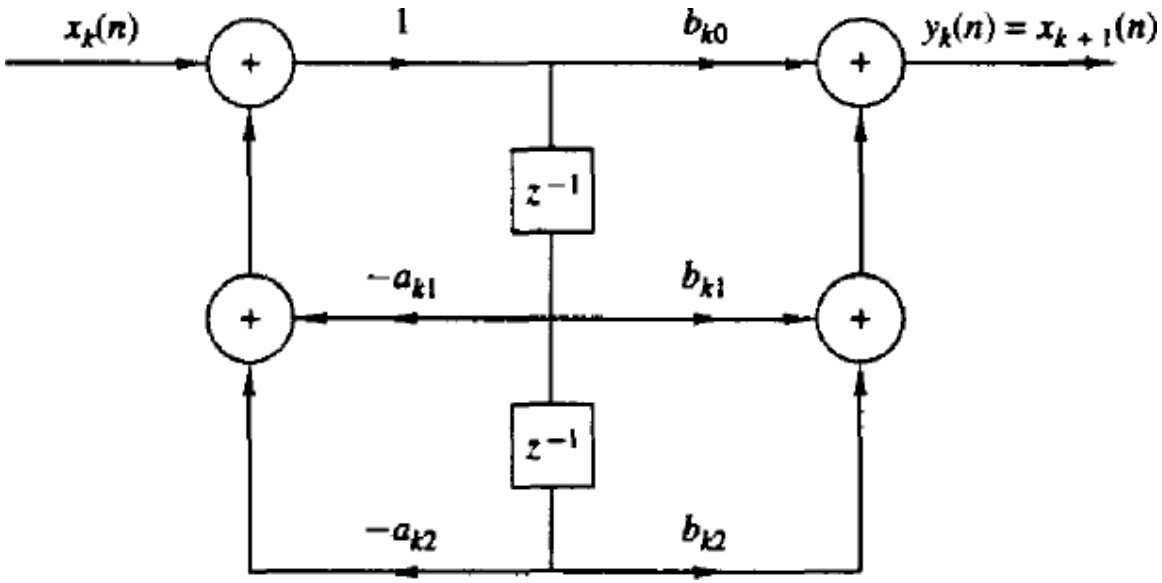
$$H_k(z) = \frac{b_{k0} + b_{k1} z^{-1} + b_{k2} z^{-2}}{1 + a_{k1} z^{-1} + a_{k2} z^{-2}}$$



Structures for IIR System: Cascade form structures

Cascade form structures

$$H_k(z) = \frac{b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2}}{1 + a_{k1}z^{-1} + a_{k2}z^{-2}}$$



Each of the second order subsystem can be realized in either the direct or canonic form. Nevertheless, the canonic form is preferred because it requires the minimum number of delay elements.

Structures for IIR System: Cascade form structures Problem

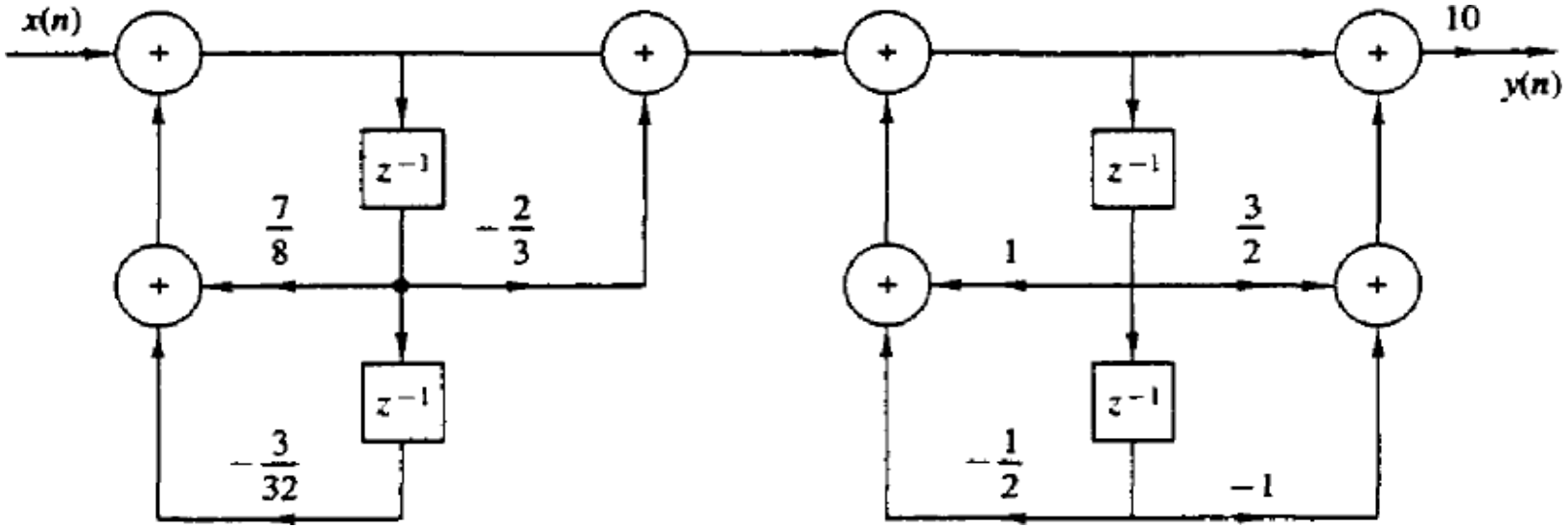
Example 7.3.1(Prokis): Determine the cascade realization of the system described by the system function

$$H(z) = \frac{10\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right)(1+2z^{-1})}{\left(1-\frac{3}{4}z^{-1}\right)(1-\frac{1}{8}z^{-1})\left\{1-(\frac{1}{2}+j\frac{1}{2})z^{-1}\right\}\left\{1-(\frac{1}{2}-j\frac{1}{2})z^{-1}\right\}}$$

Solution:

To form second order system we can make pair of 1st order system. One possible pairing of pole and zeros is given below where the pairs are formed in such way that the coefficients are real-

$$H_1(z) = \frac{1-\frac{2}{3}z^{-1}}{1-\frac{7}{8}z^{-1}+\frac{3}{32}z^{-2}} \qquad H_2(z) = \frac{1+\frac{3}{2}z^{-1}-z^{-2}}{1-z^{-1}+\frac{1}{2}z^{-2}} \qquad H(z) = 10H_1(z)H_2(z)$$

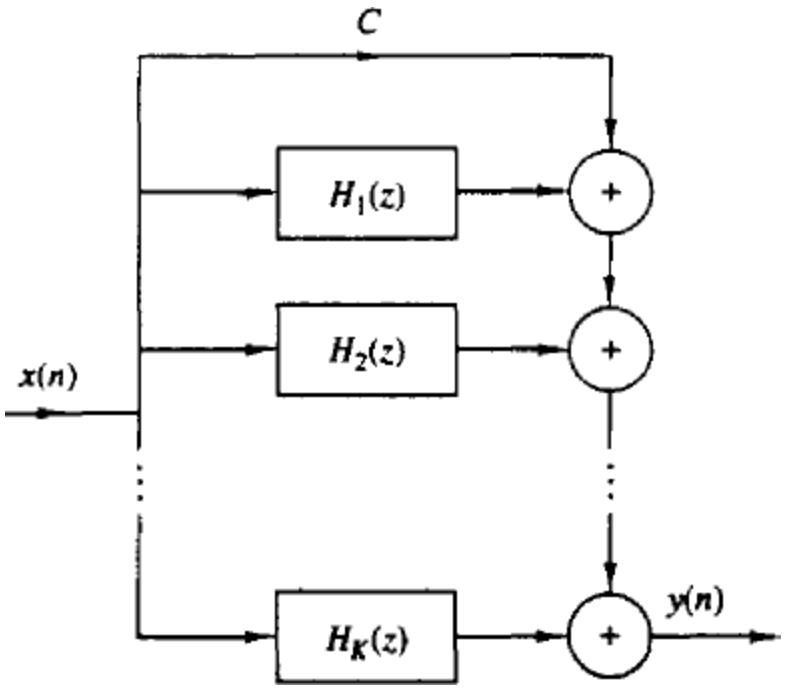


Structures for IIR System: Parallel form structures

Parallel form structures

A parallel-form realization of an IIR system can be obtained by performing a partial-fraction expansion of H(z) assuming that $N \geq M$. By performing partial-fraction expansion of H(z), we obtain the result

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} = C + \sum_{k=1}^{N} \frac{A_k}{1 - p_k z^{-1}}$$



Where $\{p_k\}$ are the poles, $\{A_k\}$ are the coefficients in the partial-fraction expansion, and the constant C is defined as $C = \frac{b_N}{a_N}$.

The parallel form structure consists of a parallel bank of single pole filters.

Structures for IIR System: Parallel form structures

Parallel form structures

In general, some of the poles of H(z) may be complex valued. In such a case, the corresponding coefficients A_k are also complex valued. To avoid multiplications by complex numbers, we can combine pairs of complex conjugate poles to form two-pole subsystems. Pairs of real-value poles are also combined arbitrary manner to form two-pole subsystems. Each of these subsystems has the form

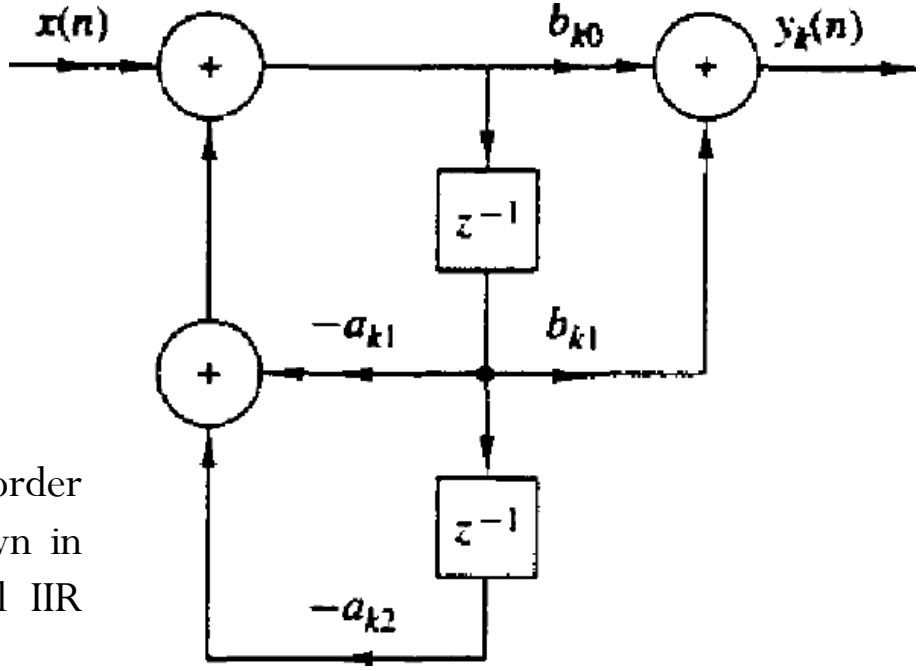
$$H_k(z) = \frac{b_{k0} + b_{k1}z^{-1}}{1 + a_{k1}z^{-1} + a_{k2}z^{-2}}$$

The overall function can be expressed as:

$$H(z) = C + \sum_{k=1}^{K} H_k(z)$$

Where K is the integer part of (N+1)/2

Direct form II (canonic) structure of the second-order section in a parallel IIR system realization is shown in Fig. which is the basic building block of parallel IIR system.



Structures for IIR System: Parallel form structures Problem

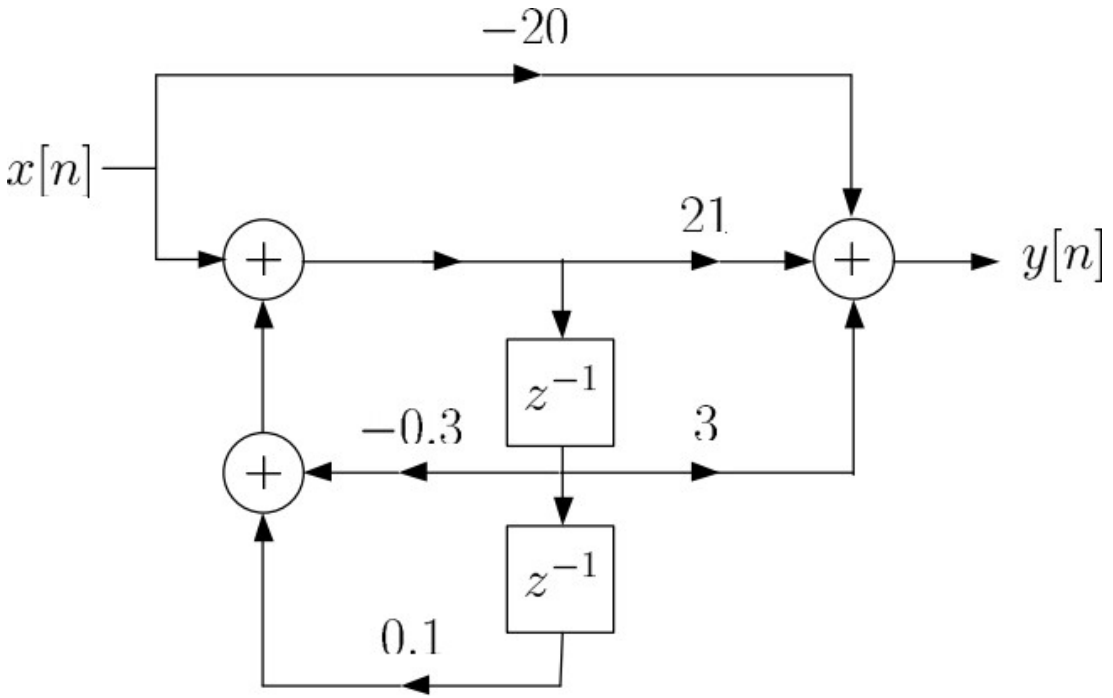
Problem: Draw the block diagram using parallel form with second order subsections for a LTI system whose transfer function is:

$$H(z) = \frac{1 - 3z^{-1} + 2z^{-2}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

Solution:

Following the long division we obtained:

$$H(z) = -20 + \frac{21 + 3z^{-1}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$



Structures for IIR System: Parallel form structures Problem

Problem: Draw the block diagram using parallel form with 1st order subsections for a LTI system whose transfer function is:

$$H(z) = \frac{1 - 3z^{-1} + 2z^{-2}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

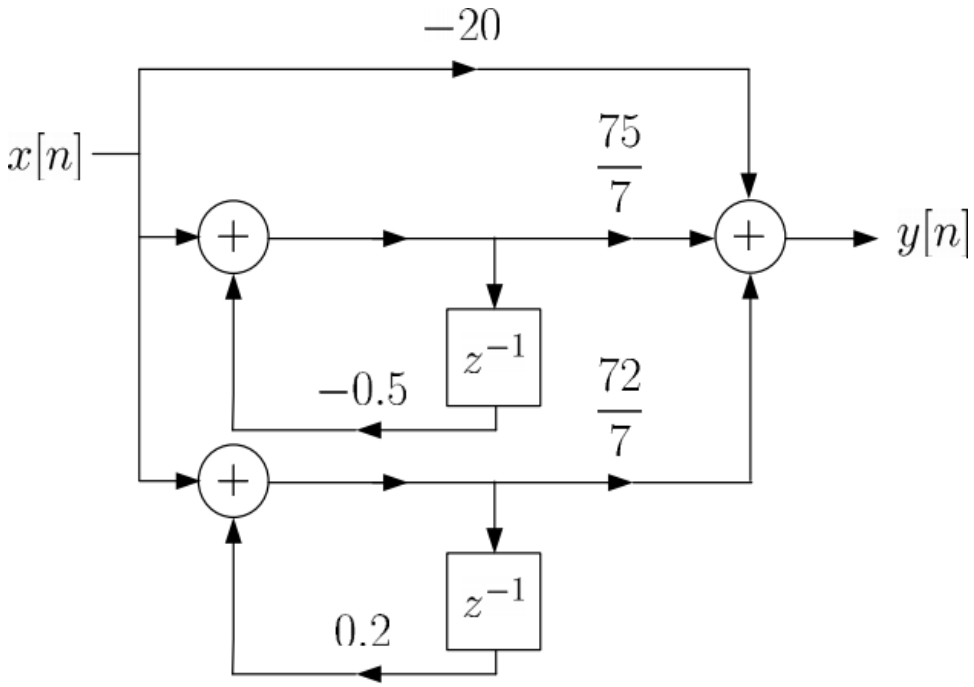
Solution:

Following the long division we obtained:

$$H(z) = -20 + \frac{21 + 3z^{-1}}{1 + 0.3z^{-1} - 0.1z^{-2}}$$

This expression is further expanded in partial fraction to get first-order sections as

$$H(z) = -20 + \frac{\frac{75}{7}}{1 + 0.5z^{-1}} + \frac{\frac{72}{7}}{1 - 0.2z^{-1}}$$



Structures for IIR System: Parallel form structures Problem

Example 7.3.1(Prokis): Determine the parallel realization of the system described by the system function

$$H(z) = \frac{10\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right)(1+2z^{-1})}{\left(1-\frac{3}{4}z^{-1}\right)(1-\frac{1}{8}z^{-1})\left\{1-(\frac{1}{2}+j\frac{1}{2})z^{-1}\right\}\left\{1-(\frac{1}{2}-j\frac{1}{2})z^{-1}\right\}}$$

Solution:

To obtain the parallel form realization, H(z) must be expanded in partial fractions.Thus we have

$$H(z) = \frac{A_1}{1-\frac{3}{4}z^{-1}} + \frac{A_2}{1-\frac{1}{8}z^{-1}} + \frac{A_3}{1-(\frac{1}{2}+j\frac{1}{2})z^{-1}} + \frac{A_3^*}{1-(\frac{1}{2}-j\frac{1}{2})z^{-1}}$$

The above expression can be written as:

$$10\left(1-\frac{1}{2}z^{-1}\right)\left(1-\frac{2}{3}z^{-1}\right)(1+2z^{-1}) = A_1\left[\left(1-\frac{1}{8}z^{-1}\right)\left\{1-(\frac{1}{2}+j\frac{1}{2})z^{-1}\right\}\left\{1-(\frac{1}{2}-j\frac{1}{2})z^{-1}\right\}\right]$$
$$+A_2\left[\left(1-\frac{3}{4}z^{-1}\right)\left\{1-(\frac{1}{2}+j\frac{1}{2})z^{-1}\right\}\left\{1-(\frac{1}{2}-j\frac{1}{2})z^{-1}\right\}\right]$$
$$+A_3\left[\left(1-\frac{3}{4}z^{-1}\right)(1-\frac{1}{8}z^{-1})\left\{1-(\frac{1}{2}-j\frac{1}{2})z^{-1}\right\}\right] + A_3^*\left[\left(1-\frac{3}{4}z^{-1}\right)(1-\frac{1}{8}z^{-1})\left\{1-(\frac{1}{2}+j\frac{1}{2})z^{-1}\right\}\right]$$

After some arithmetic we find that

$$A_1 = 2.93, \quad A_2 = -17.68, \quad A_3 = 12.25 - j14.57, \quad A_3^* = 12.25 + j14.57$$

Recombining pair of poles we get:

Structures for IIR System: Parallel form structures Problem

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-6
Discrete Fourier Transform (DFT) & Fast Fourier Transform (FFT)

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ Fourier Analysis
- ✓ Discrete Fourier Transform (DFT)
- ✓ Difference between DTFT and DFT.
- ✓ The DFT as linear Transformation.
- ✓ Problems related to DFT and IDFT
- ✓ Circular convolution
- ✓ Fast Fourier Transform
- ✓ Decimation in Time FFT Algorithm

Reference Book:

Digital Signal Processing (4th Edition), John G. Proakis, Dimitris K Manolakis

Chapter 7 (Discrete Fourier Transform) and
Chapter 8(Efficient Computation of the DFT)

Digital Signal Processing, Barrie Jervis

Chapter-2 (Discrete Transforms)

Week 12
Slide 194-211

Fourier Analysis

Fourier analysis convert a time domain signal into frequency domain signal. Can be divided into 4 types:

a) Aperiodic continuous.

b) Periodic continuous (Fourier Series).

c) Aperiodic Discrete (DTFT).

d) Periodic Discrete (DFT)

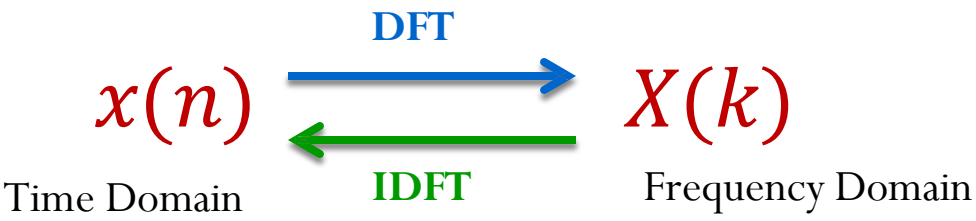
Discrete Fourier Transform

The discrete Fourier transform of a discrete-time signal x(n) is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi nk}{N}} \qquad \text{K=0, 1, ..., N-1}$$

The Inverse Discrete Fourier Transform (IDFT) is defined as

$$x(n) = \frac{1}{N} \sum_{n=0}^{N-1} X(k) e^{j\frac{2\pi nk}{N}} , \qquad n = 0,1, \dots \dots \dots N-1$$



Difference between DTFT and DFT

$$X(\omega) = \sum_{n=-\infty}^{+\infty} x(n)e^{-j\omega n}$$

In DTFT frequency domain (ω) is continuous.

ω changes from 0 to 2π, but it is continuous.

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi nk}{N}}$$

In DFT frequency domain ($\omega = \frac{2\pi k}{N}$) is discrete.

ω changes from 0 to 2π taking 0 to (N-1) number of samples.

The DFT as a Linear Transformation

The N-Point DFT is defined as $$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi nk}{N}}$$

The above expression also written as $$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}$$

Where, $W_N^{nk} = e^{-j\frac{2\pi nk}{N}}$ Is called the twiddle factor

If N=4 we say it is 4-Point DFT. For N=4, the above expression can be written as

$$X(k) = \sum_{n=0}^{3} x(n) W_4^{nk}$$

$$X(0) = x(0)W_4^0 + x(1)W_4^0 + x(2)W_4^0 + x(3)W_4^0$$

$$X(1) = x(0)W_4^0 + x(1)W_4^1 + x(2)W_4^2 + x(3)W_4^3$$

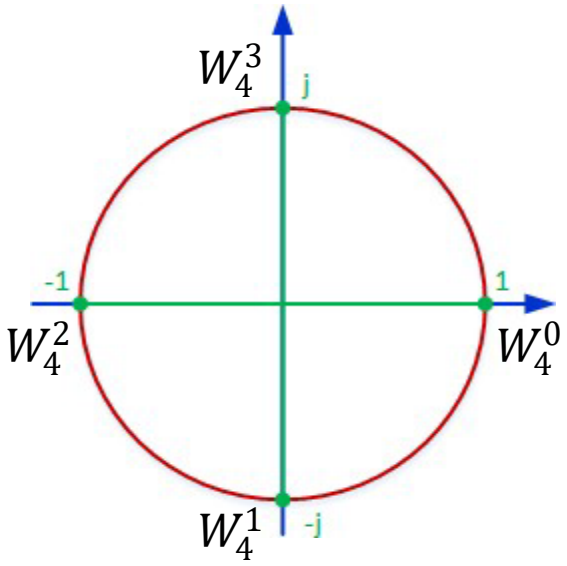
$$X(2) = x(0)W_4^0 + x(1)W_4^2 + x(2)W_4^4 + x(3)W_4^6$$

$$X(3) = x(0)W_4^0 + x(1)W_4^3 + x(2)W_4^6 + x(3)W_4^9$$

The DFT as a Linear Transformation (cont.)

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix}$$

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix}$$



Periodicity property of W

$$W_N^{k+\frac{N}{2}} = -W_N^k \gg W_4^{k+2} = -W_4^k$$

Note that for 4 point DFT, the computation of each point of DFT can be accomplished by 4 complex multiplication and (4-1) complex additions.

Total computation for 4-point DFT, Addition= (4-1)X3 and Multiplication= 4X4

Hence the N-point DFT values can be computed in a total of N^2 multiplications and N(N-1) complex additions.

The DFT as a Linear Transformation (cont.)

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ . \\ . \\ X(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & . & . & 1 \\ 1 & W_N^1 & W_N^2 & . & . & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & . & . & W_N^{2(N-1)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & . & . & W_N^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ . \\ . \\ x(N-1) \end{bmatrix}$$

The N-point DFT may be expressed in matrix form as: $\mathbf{X}_N = \mathbf{W}_N \mathbf{x}_N$

Where W_N is the matrix of the linear transformation. If we assume that the inverse of W_N exists, we obtain $\mathbf{x}_N = \mathbf{W}_N^{-1} \mathbf{X}_N$

The expression of inverse DTFT can be expressed as

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn},$$

$$n = 0, 1, \ldots, N-1$$

$$\mathbf{x}_N = \frac{1}{N} \mathbf{W}_N^* \mathbf{X}_N$$

$$\mathbf{W}_N^{-1} = \frac{1}{N} \mathbf{W}_N^*$$

$$\mathbf{W}_N \mathbf{W}_N^* = N \mathbf{I}_N$$

Problem on DFT

Example 7.1.3: Compute the DFT of the 4-point sequence
x(n)={0,1,2,3}

Solution:

The matrix of the linear transformation for 4-point DFT can be expressed as

$$W_4 = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

Now, Using the matrix expression $X_N = W_N x_N$, we can calculate DFT of the above sequence as

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 0+1+2+3 \\ 0-j-2+3j \\ 0-1+2-3 \\ 0+j-2-3j \end{bmatrix} = \begin{bmatrix} 6 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix}$$

Problem on IDFT

Problem:The 4-point DFT of a discrete time sequence x(n) is given below. Determine x(n).

Solution: **X(k)={6,-2+2j,-2,-2-2j}**

The matrix of the linear transformation for 4-point DFT can be expressed as

$$W_4 = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \quad \text{So, } W_4^* = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}$$

Now, Using the matrix expression $x_N = \frac{1}{N} W_N^* X_N$, we can calculate IDFT of the above sequence as

$$\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 6 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 0 \\ 4 \\ 8 \\ 12 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix}$$

Problem: Compute the convolution of the sequences $x_1(n) = \{1, 2, 3, 1\}$ and $x_2(n) = \{4, 3, 2, 2\}$ using DFT and IDFT

Multiplication of Two DFTs and Circular Convolution

Suppose that we have two finite-duration sequences of length N, $x_1(n)$ and $x_2(n)$. Their respective N-point DFTs are

$$X_1(k) = \sum_{n=0}^{N-1} x_1(n) e^{-j2\pi nk/N}, \qquad k = 0, 1, \ldots, N-1$$

$$X_2(k) = \sum_{n=0}^{N-1} x_2(n) e^{-j2\pi nk/N}, \qquad k = 0, 1, \ldots, N-1$$

If we multiply the two DFTs together, the result is a DFT say $X_3(k)$, of a sequence $x_3(n)$ of length N.

Let us determine the relationship between $x_3(n)$ and the sequences $x_1(n)$ and $x_2(n)$.

$$X_3(k) = X_1(k) X_2(k), \qquad k = 0, 1, \ldots, N-1$$

The IDFT of $X_3(k)$ is:

$$x_3(m) = \frac{1}{N} \sum_{k=0}^{N-1} X_3(k) e^{j2\pi km/N}$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} X_1(k) X_2(k) e^{j2\pi km/N}$$

Multiplication of Two DFTs and Circular Convolution (Cont.)

Putting the value $X_1(k)$ and $X_2(k)$

$$x_3(m) = \frac{1}{N}\sum_{k=0}^{N-1}\left[\sum_{n=0}^{N-1} x_1(n)e^{-j2\pi kn/N}\right]\left[\sum_{l=0}^{N-1} x_2(l)e^{-j2\pi kl/N}\right]e^{j2\pi km/N}$$

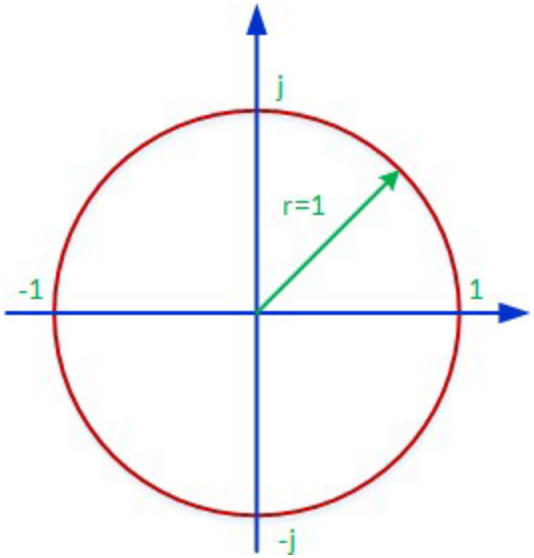
$$= \frac{1}{N}\sum_{n=0}^{N-1} x_1(n)\sum_{l=0}^{N-1} x_2(l)\left[\sum_{k=0}^{N-1} e^{j2\pi k(m-n-l)/N}\right]$$

We can write

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & a = 1 \\ \dfrac{1-a^N}{1-a}, & a \neq 1 \end{cases}$$

Where,

$$a = e^{j2\pi(m-n-l)/N}$$



if (m-n-l) is a multiple of N, i.e.

$$m - n - l = pN, p \text{ is an integer}$$

a becomes $a = e^{j2\pi p} = 1$

Multiplication of Two DFTs and Circular Convolution (Cont.)

$a^N = 1$, for any value $a \neq 0$

$$a^N = e^{j2\pi(m-n-l)} = \cos\big(2\pi(m-n-l)\big) + j\sin\big(2\pi(m-n-l)\big) = 1$$

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & l = m-n+pN = ((m-n))_N, \quad p \text{ an integer} \\ 0, & \text{otherwise} \end{cases}$$

Finally we can write,

$$x_3(m) = \sum_{n=0}^{N-1} x_1(n)x_2((m-n))_N, \qquad m = 0, 1, \ldots, N-1$$

The above expression has the form of a convolution sum. However it is not the ordinary linear convolution which relates the output sequence y(n) of a linear system to the input sequence x(n) and impulse response h(n). **The above expression involves the index $(m-n)_N$ and is called circular convolution. Multiplication of DFTs of two sequences is equivalent to the circular convolution of the two sequences in the time domain.**

Problem on Circular Convolution

Problem: **Compute the circular convolution of following two sequences**

$$x_1(n) = \{\underset{\uparrow}{2}, 1, 2, 1\}$$

$$x_2(n) = \{\underset{\uparrow}{1}, 2, 3, 4\}$$

Solution:

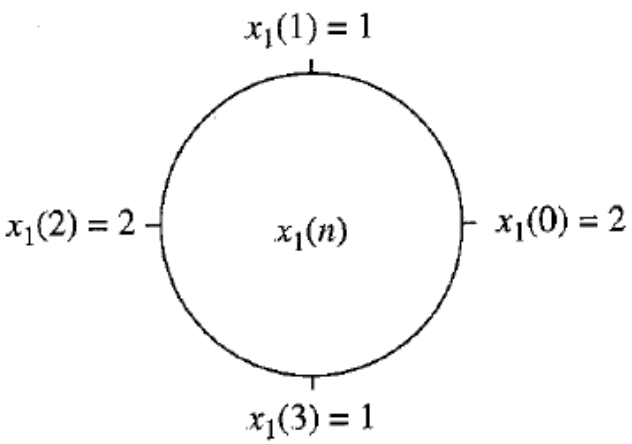
$$x_3(m) = \sum_{n=0}^{N-1} x_1(n)x_2((m-n))_N, \qquad m = 0, 1, \ldots, N-1$$

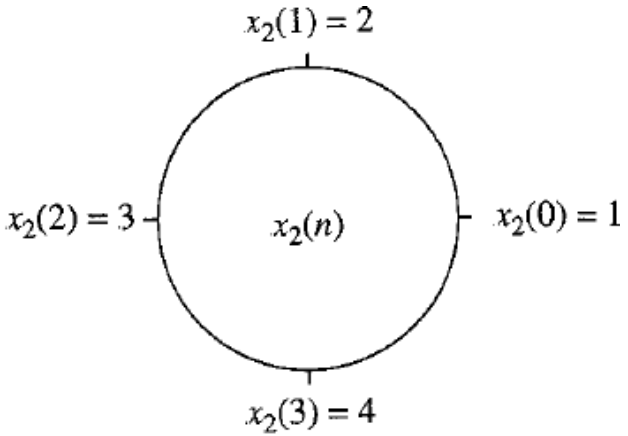
$$x_3(0) = \sum_{n=0}^{3} x_1(n)x_2((-n))N \qquad x_3(1) = \sum_{n=0}^{3} x_1(n)x_2((1-n))_4$$

$$x_3(2) = \sum_{n=0}^{3} x_1(n)x_2((2-n))_4 \qquad x_3(3) = \sum_{n=0}^{3} x_1(n)x_2((3-n))_4$$

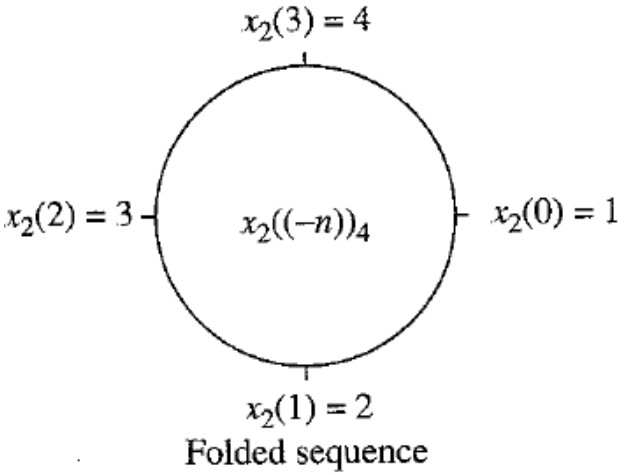
We can represent the sequences in graph where the samples are placed in counterclockwise direction in a circle

Problem on Circular Convolution (Cont.)

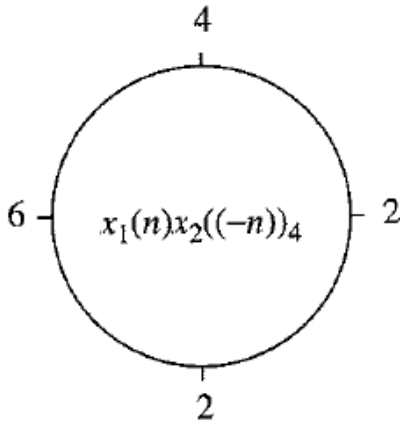




(a)



Folded sequence

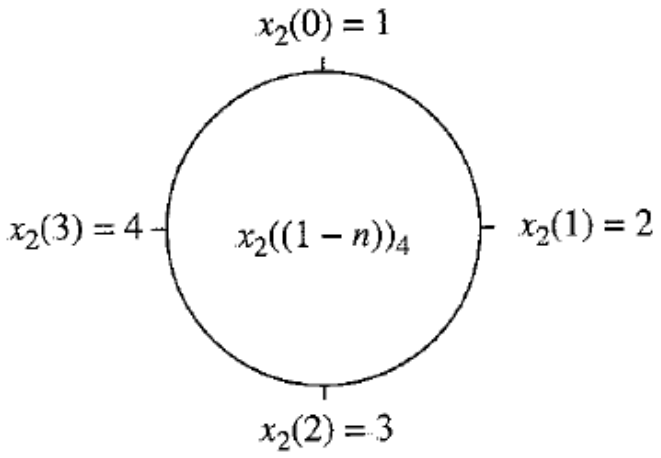


Product sequence

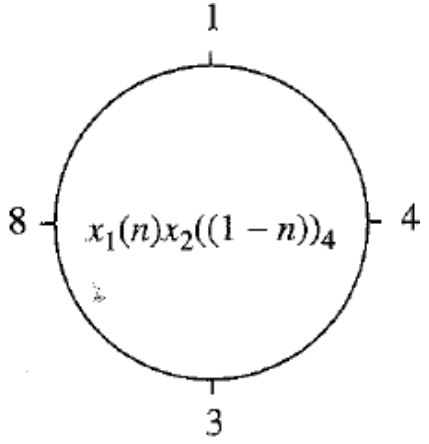
(b)

$$x_3(0) = 2 + 4 + 6 + 2 = 14$$

Problem on Circular Convolution (Cont.)



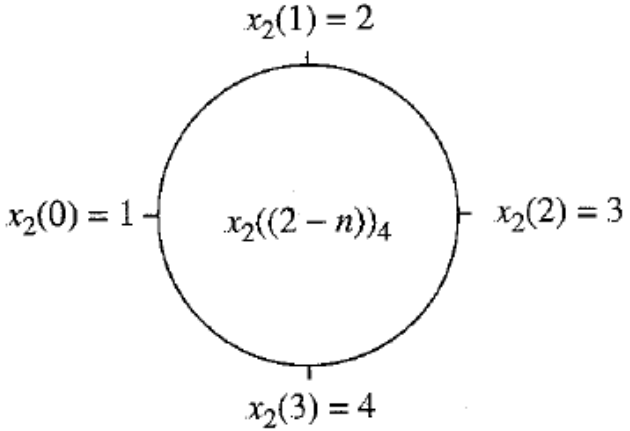
Folded sequence rotated by one unit in time



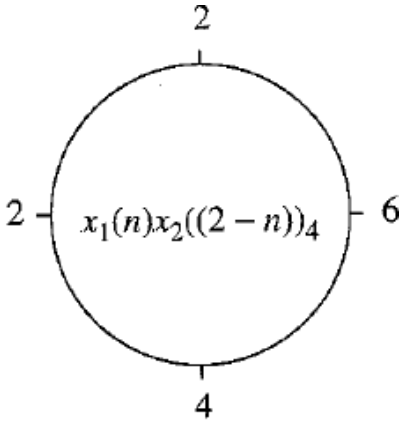
Product sequence

(c)

$$x_3(1) = 4 + 1 + 8 + 3 = 16$$



Folded sequence rotated by two units in time

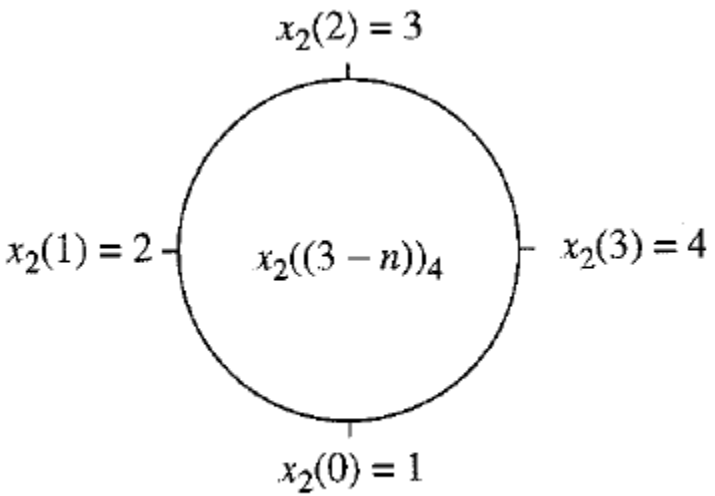


Product sequence

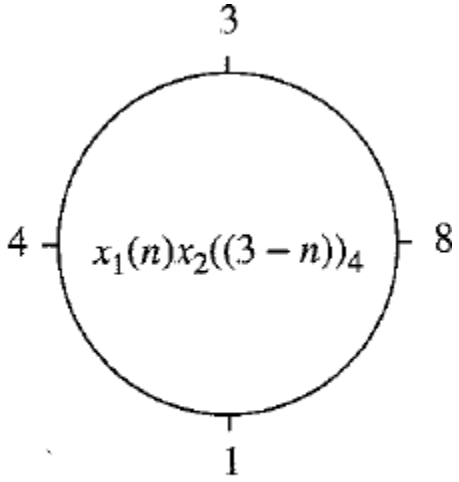
(d)

$$x_3(2) = 6 + 2 + 2 + 4 = 14$$

Problem on Circular Convolution (Cont.)



Folded sequence rotated by three units in time



Product sequence

(e)

$$x_3(3) = 8 + 3 + 4 + 1 = 16$$

$$x_3(n) = \{14, 16, 14, 16\}$$
$$\uparrow$$

Alternative method of computing Circular Convolution

Problem: Compute the circular convolution of following two sequences

$$x_1(n) = \{\underset{\uparrow}{2}, 1, 2, 1\}$$

$$x_2(n) = \{\underset{\uparrow}{1}, 2, 3, 4\}$$

Solution:

$$\begin{bmatrix} x_3(0) \\ x_3(1) \\ x_3(2) \\ x_3(3) \end{bmatrix} = \begin{bmatrix} 2 & 1 & 2 & 1 \\ 1 & 2 & 1 & 2 \\ 2 & 1 & 2 & 1 \\ 1 & 2 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 2+2+6+4 \\ 1+4+3+8 \\ 2+2+6+4 \\ 1+4+3+8 \end{bmatrix} = \begin{bmatrix} 14 \\ 16 \\ 14 \\ 16 \end{bmatrix}$$

$$x_3(n) = \{\underset{\uparrow}{14}, 16, 14, 16\}$$

Zero padding is a simple concept, it simply refers to adding zeros to end of a time domain signals to increase its length.

Zero padding

Computation of linear convolution by circular convolution

Compute the linear convolution of the following two sequences by circular convolution

$$x(n) = \{\underset{\uparrow}{1}, 2, 3, 1\} \qquad h(n) = \{1, \underset{\uparrow}{2}, 1, -1\}$$

Solution:

Length of convolution sequence = 4+41=7

Start of sequence= min (min(x(n)) min(h(n))) =-1

After zero padding:

$$x\,(n) = \underset{\uparrow}{\{1,2,3,1,0,0,0\}} \qquad h\,(n) = \underset{\uparrow}{\{1,2,1,-1,0,0,0\}}$$

$$\begin{bmatrix} y(-1) \\ y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 3 & 2 \\ 2 & 1 & 0 & 0 & 0 & 1 & 3 \\ 3 & 2 & 1 & 0 & 0 & 0 & 1 \\ 1 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 3 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 & 2 & 1 & 0 \\ 0 & 0 & 0 & 1 & 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 1 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 8 \\ 8 \\ 3 \\ -2 \\ -1 \end{bmatrix}$$

$$y(n) = \{\ldots, 0, 0, \underset{\uparrow}{1}, 4, 8, 8, 3, -2, -1, 0, 0, \ldots\}$$

Class Test Next Week

Syllabus: Slide 121-213

Week 13
Slide 213-224

Fast Fourier Transform (FFT)

A Fast Fourier Transform (FFT) is an efficient algorithm to compute the Discrete Fourier Transform (DFT) and inverse of DFT. FFT requires a smaller number of arithmetic operations such as multiplications and addition than DFT (i.e. FFT requires lesser computation time than DFT).

No of computations in direct DFT	Computations in FFT
Multiplications: N^2 Additions: N(N-1)	Multiplications: $\frac{N}{2} log_2(N)$ Additions: $N\ log_2(N)$

For, $N = 10^6$

Total mathematical operations required to find DFT

Direct DFT: $10^{12} + 10^6(10^6 - 1) \approx 2 \times 10^{12}$

FFT: $(5 \times 10^5 \times log_2(10^6) + 10^6 \times log_2(10^6) \cong 24 \times 10^6$

If each mathematical operation needs 1 ns to compute by digital computer,

Direct DFT needs $2 \times 10^{12} ns = 2 \times 10^3 s = 2000 s$

Whether FFT algorithm needs $24 \times 10^6 \text{ns} = 24 \times 10^{-3} s = 0.024\ s$

Fast Fourier Transform (FFT) Algorithm

- ✓ Direct computation of the DFT is less efficient because it does not exploit the properties of symmetry and periodicity of the phase factor W_N^{nk}

Symmetry property: $W_N^{k+N/2} = -W_N^k$

Periodicity property: $W_N^{k+N} = W_N^k$

- ✓ FFT algorithms exploit the above properties of phase factor to reduce the number of mathematical calculations to compute DFT. There are many FFT algorithm which involves a wide range of mathematics.
- ✓ On the basis of decimation (decimation means decomposition into decimal parts) process FFT

algorithms are two types.

- ✓ **Decimation-in-Time FFT algorithm:** The sequence x(n) will be broken up into odd numbered and even numbered subsequences. This algorithm was first proposed by Cooley and Tukey in 1965.

The decimation in time FFT Algorithm

Cooley-Tukey Algorithm

$$X(k) = \sum_{n=0}^{N-1} x_n e^{-j\frac{2\pi nk}{N}}, \quad k = 0,1,\dots,N-1$$

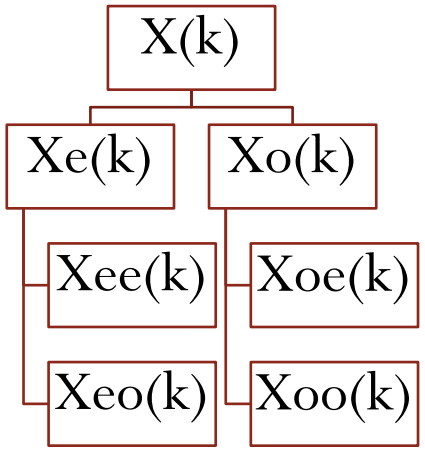
$$= \sum_{n=0}^{N-1} x_n W_N^{nk} \qquad \text{Where, } W = e^{-j\frac{2\pi}{N}}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} W_N^{2nk} + \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} W_N^{(2n+1)k}$$

Even sequence x_0, x_2, x_4, x_6

Odd sequence x_1, x_3, x_5, x_7

$$= \sum_{n=0}^{\frac{N}{2}-1} x_{2n} W_{N/2}^{2nk} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} x_{2n+1} W_{N/2}^{nk} = X_e(k) + W_N^k X_o(k), \quad k = 0,1,\dots,N-1$$



Some relationship involving W_N

$$W_N^2 = (e^{-j\frac{2\pi}{N}})^2 = e^{-j\frac{2\pi}{N/2}} = W_{N/2}$$

Symmetry: $W_N^{(k+\frac{N}{2})} = W_N^k W_N^{N/2}$

$$= W_N^k e^{-j\pi} = -W_N^k$$

Periodicity: $W_N^{(k+N)} = -W_N^k$

The decimation in time FFT Algorithm (Cont.)

Stage-1

$$X_e(k) = X_{ee}(k) + W_{\frac{N}{2}}^{k} X_{eo}(k), \qquad k = 0 \text{ to } \frac{N}{2} - 1$$

Even sequence x_0, x_4 — Odd sequence x_2, x_6

$$X_{ee}(k) = x_o + W_{\frac{N}{4}}^{k} x_4 \qquad X_{eo}(k) = x_2 + W_{\frac{N}{4}}^{k} x_6$$

$$X_{ee}(0) = x_o + x_4 \qquad X_{eo}(0) = x_2 + x_6$$

$$X_{ee}(1) = x_o - x_4 \qquad X_{eo}(1) = x_2 - x_6$$

$$W_{\frac{N}{4}}^{1} = W_2^1 = e^{-j\frac{2\pi}{2}.1} = -1$$

$$X_o(k) = X_{oe}(k) + W_{\frac{N}{2}}^{k} X_{oo}(k), \qquad k = 0 \text{ to } \frac{N}{2} - 1$$

Even sequence x_1, x_5 — Odd sequence x_3, x_7

$$X_{oe}(0) = x_1 + x_5 \qquad X_{oo}(0) = x_3 + x_7$$

$$X_{oe}(1) = x_1 - x_5 \qquad X_{oo}(1) = x_3 - x_7$$

The decimation in time FFT Algorithm (Cont.)

Stage-2

$$X_e(k) = X_{ee}(k) + W_{\frac{N}{2}}^{k} X_{eo}(k), \qquad k = 0 \text{ to } \frac{N}{2} - 1$$

$$X_e(0) = X_{ee}(0) + W_{8/2}^{0} X_{eo}(0) = X_{ee}(0) + W_8^0 X_{eo}(0)$$

$$X_e(1) = X_{ee}(1) + W_{\frac{8}{2}}^{1} X_{eo}(1) = X_{ee}(1) + W_8^2 X_{eo}(1)$$

$$X_e(2) = X_{ee}(2) + W_{\frac{8}{2}}^{2} X_{eo}(2) = X_{ee}(0) - W_8^0 X_{eo}(0)$$

$$X_e(3) = X_{ee}(3) + W_{\frac{8}{2}}^{3} X_{eo}(3) = X_{ee}(1) - W_8^2 X_{eo}(1)$$

$X_{ee}(2) = x_o + W_{\frac{N}{4}}^{2} x_4$

$= x_o + W_2^2 x_4 = x_o + x_4$

$= X_{ee}(0)$

$X_{eo}(2) = X_{eo}(0)$

$W_{\frac{8}{2}}^{2} = e^{-j\frac{2\pi 4}{8}} = -1$

$W_N^{(k+\frac{N}{2})} = -W_N^k$

Similarly,

$$X_o(0) = X_{oe}(0) + W_{8/2}^{0} X_{oo}(0) = X_{oe}(0) + W_8^0 X_{oo}(0)$$

$$X_o(1) = X_{oe}(1) + W_{\frac{8}{2}}^{1} X_{oo}(1) = X_{oe}(1) + W_8^2 X_{oo}(1)$$

$$X_o(2) = X_{oe}(2) + W_{\frac{8}{2}}^{2} X_{oo}(2) = X_{oe}(0) - W_8^0 X_{oo}(0)$$

$$X_o(3) = X_{oe}(3) + W_{\frac{8}{2}}^{3} X_{oo}(3) = X_{oe}(1) - W_8^2 X_{oo}(1)$$

The decimation in time FFT Algorithm (Cont.)

$$X(k) = X_e(k) + W_N^k X_o(k),$$

Stage-3

$$X(0) = X_e(0) + W_8^0 X_o(0)$$

$$X(1) = X_e(1) + W_8^1 X_o(1)$$

$$X(2) = X_e(2) + W_8^2 X_o(2)$$

$$X(3) = X_e(3) + W_8^3 X_o(3)$$

$$X(4) = X_e(4) + W_8^4 X_o(4) = X_e(0) - W_8^0 X_o(0)$$

$$X(5) = X_e(5) + W_8^5 X_o(5) = X_e(1) - W_8^1 X_o(1)$$

$$X(6) = X_e(6) + W_8^6 X_o(6) = X_e(2) - W_8^2 X_o(2)$$

$$X(7) = X_e(7) + W_8^7 X_o(7) = X_e(3) - W_8^3 X_o(3)$$

$$X_{ee}(4) = x_o + W_2^4 x_4$$

$$= x_o + x_4 = X_{ee}(0)$$

Similarly, $X_{eo} = X_{eo}(0)$

$$X_e(4) = X_{ee}(4) + W_{\frac{8}{2}}^4 X_{eo}(4)$$

$$= X_{ee}(0) + X_{eo}(0)$$

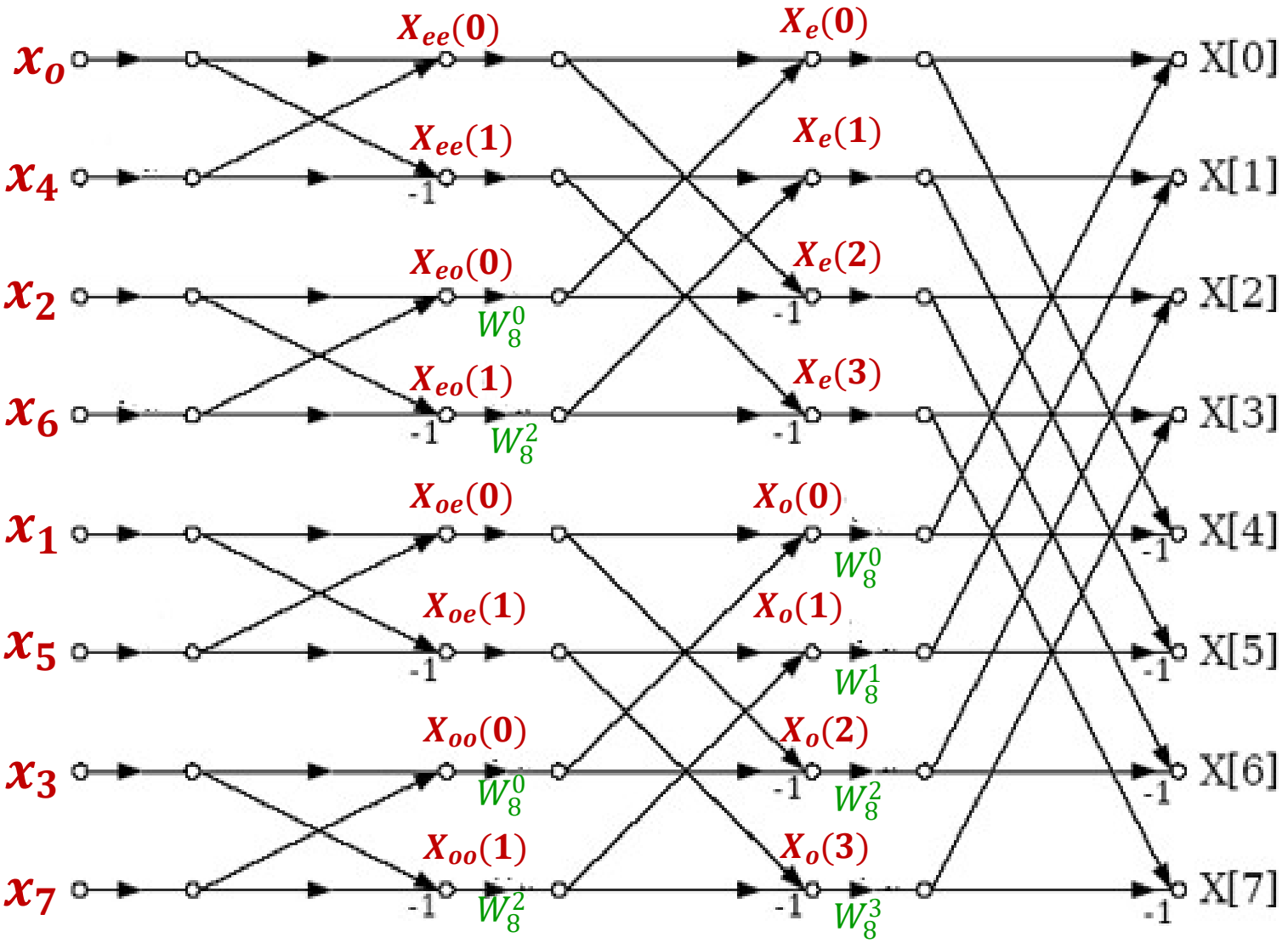
$$= X_e(0)$$

Similarly, $X_o(4) = X_0(0)$

$$W_8^{(0+\frac{8}{2})} = -\boldsymbol{W_8^0}$$

The decimation in time FFT Algorithm (Cont.)

8-point decimation in time FFT Algorithm



The decimation in time FFT Algorithm (Cont.)

Reduction of computational complexity

The basic computation performed at every stage (previous fig: 8-point DIT FFT algorithm) is to take two complex numbers, say the pair (a,b), multiply b by W_N^r, and then add and subtract the product from a to form two new complex numbers (A,B). This basic computation is called a **butterfly** because the flow graph resembles a butterfly.

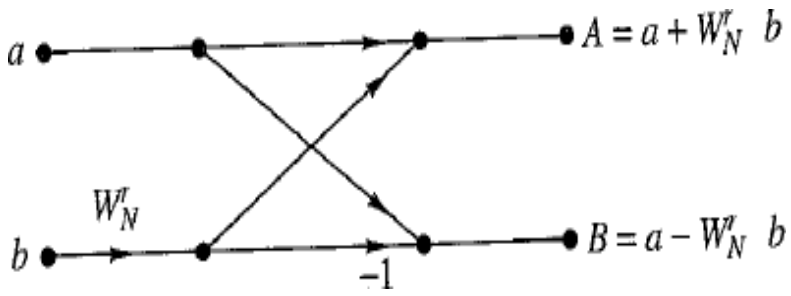


Fig: Basic butterfly computation in the decimation-in-time FFT algorithm

In general, each butterfly involves one complex multiplication and two complex additions. For N point FFT, there are N/2 butterflies per stage of the computation process and $log_2(N)$ stages.

Therefore,

Total number of complex multiplications is $\frac{N}{2} log_2(N)$

And complex addition is $N\ log_2(N)$

Comparison of Computational Complexity for the Direct Computation of the DFT Versus the FFT Algorithm

Number of Points, N	Complex Multiplications in Direct Computation, N^2	Complex Multiplications in FFT Algorithm, $(N/2)\log_2 N$	Speed Improvement Factor
4	16	4	4.0
8	64	12	5.3
16	256	32	8.0
32	1,024	80	12.8
64	4,096	192	21.3
128	16,384	448	36.6
256	65,536	1,024	64.0
512	262,144	2,304	113.8
1,024	1,048,576	5,120	204.8

The decimation in time FFT Algorithm (Cont.)

Problem on FFT

Problem: For the discrete time sequence x(n)={1,1,1,0,0,0,0,0}, find the 8-point DFT using DIT-FFT algorithm.

Solution:

Even Part	x_0	x_4	x_2	x_6
	1	0	1	0
Odd Part	x_1	x_5	x_3	x_7
	1	0	0	0

Stage-1:

$X_{ee}(0) = x_o + x_4 = 1$

$X_{ee}(1) = x_o - x_4 = 1$

$X_{eo}(0) = x_2 + x_6 = 1$

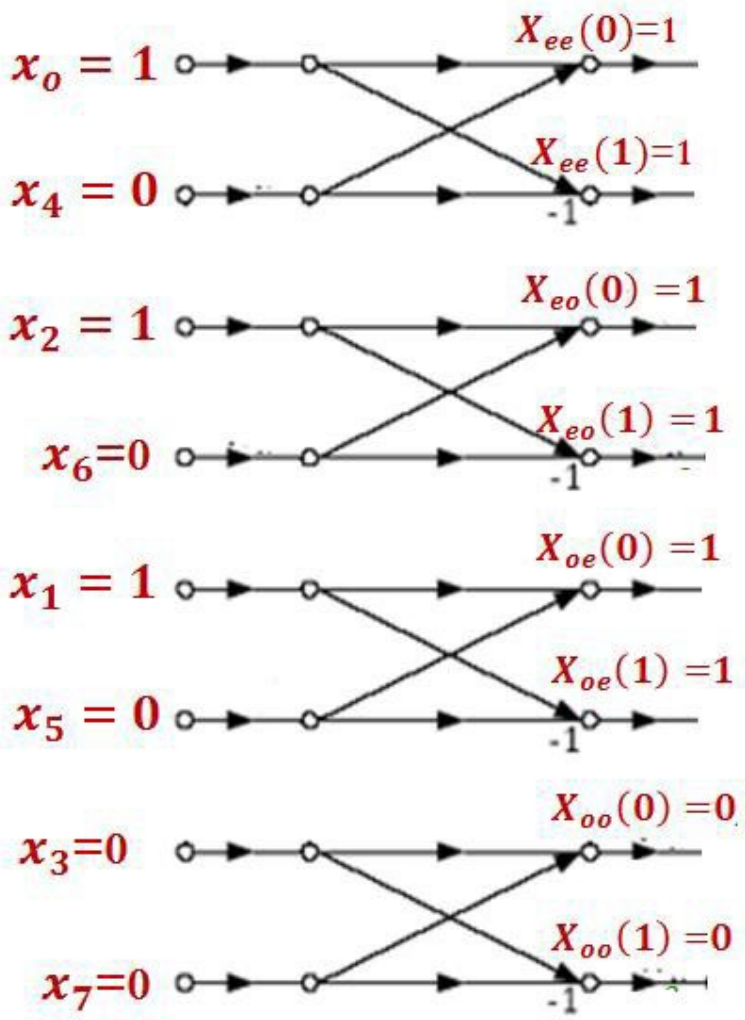
$X_{eo}(1) = x_2 - x_6 = 1$

$X_{oe}(0) = x_1 + x_5 = 1$

$X_{oe}(1) = x_1 - x_5 = 1$

$X_{oo}(0) = x_3 + x_7 = 0$

$X_{oo}(1) = x_3 - x_7 = 0$



Problem on FFT (Cont.)

Stage-2

$$W_8^2 = e^{-j\frac{\pi}{2}} = -j$$

$$X_e(0) = X_{ee}(0) + W_8^0 X_{eo}(0) = 2$$

$$X_e(1) = X_{ee}(1) + W_8^2 X_{eo}(1) = 1 - j$$

$$X_e(2) = X_{ee}(0) - W_8^0 X_{eo}(0) = 0$$

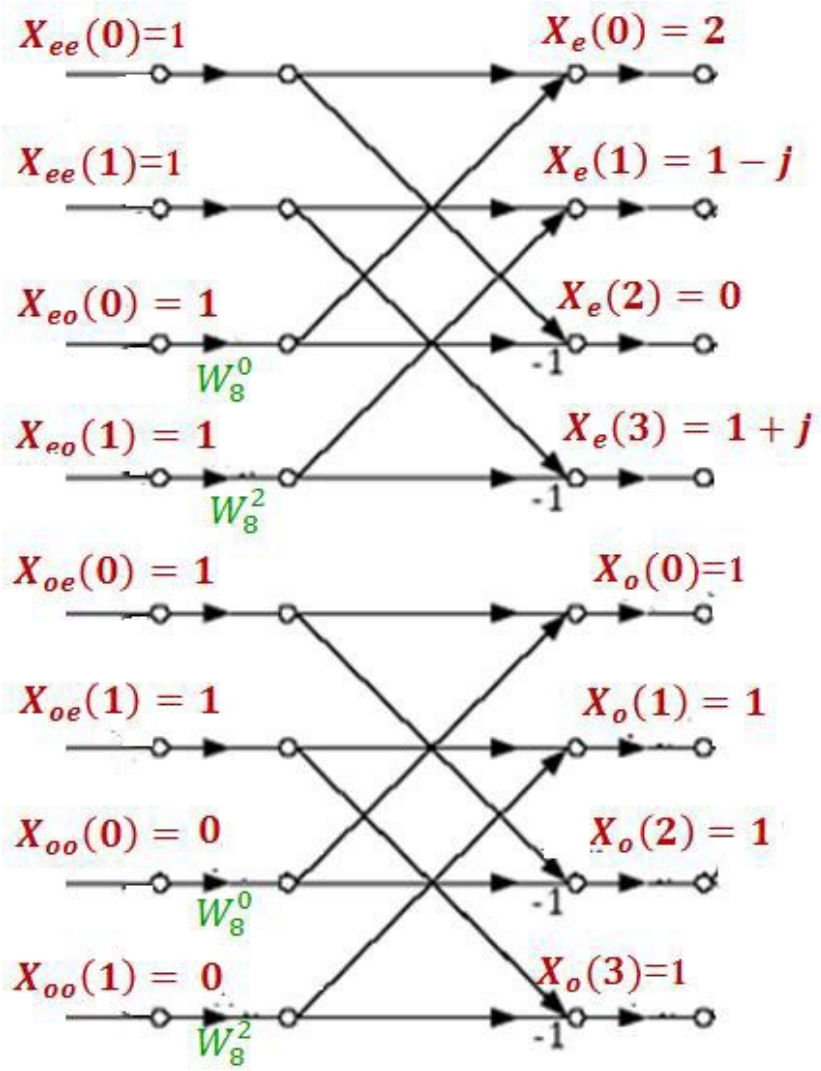
$$X_e(3) = X_{ee}(1) - W_8^2 X_{eo}(1) = 1 + j$$

$$X_o(0) = X_{oe}(0) + W_8^0 X_{oo}(0) = 1$$

$$X_o(1) = X_{oe}(1) + W_8^2 X_{oo}(1) = 1$$

$$X_o(2) = X_{oe}(0) - W_8^0 X_{oo}(0) = 1$$

$$X_o(3) = X_{oe}(1) - W_8^2 X_{oo}(1) = 1$$



Problem on FFT (Cont.)

Stage-3

$$W_8^1 = e^{-j\frac{\pi}{4}} = \frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}} \qquad W_8^3 = e^{-j\frac{3\pi}{4}} = -\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$$

$$X(0) = X_e(0) + W_8^0 X_o(0) = 3$$

$$X(1) = X_e(1) + W_8^1 X_o(1) = 1 - j + \left(\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}\right)$$

$$X(2) = X_e(2) + W_8^2 X_o(2) = -j$$

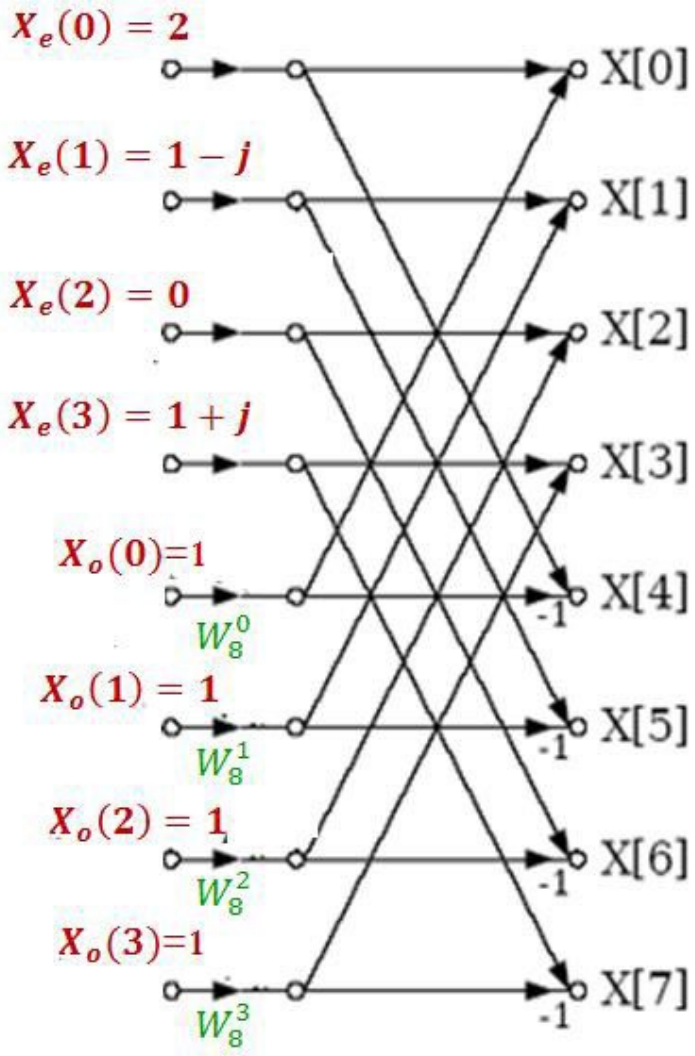
$$X(3) = X_e(3) + W_8^3 X_o(3) = 1 + j + \left(-\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}\right)$$

$$X(4) = X_e(0) - W_8^0 X_o(0) = 2 - 1 = 1$$

$$X(5) = X_e(1) - W_8^1 X_o(1) = 1 - j - \left(\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}\right)$$

$$X(6) = X_e(2) - W_8^2 X_o(2) = j$$

$$X(7) = X_e(3) - W_8^3 X_o(3) = 1 + j - \left(-\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}\right)$$



University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-7
Introduction to digital Filter

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ Function of digital Filter
- ✓ Comparison of Analog and Digital Filter
- ✓ Filter kernel
- ✓ Types of filter
- ✓ Time domain and frequency domain parameters of filter.
- ✓ Design of all frequency selective filter from low pass filter kernel.

Reference Book:

The Scientist and Engineer's Guide to Digital Signal Processing, By Steven W. Smith (2nd Edition)

Chapter-14 (Introduction to Digital Filters)

Digital Signal Processing: A practical approach, By Emmanuel C Ifeachor, Barrie W Jervis

Chapter-5 (A framework for digital filter design)

Week 14
Slide 228-239

Functions of Digital Filter

- ✓ Filters have two uses: signal separation and signal restoration.
- ✓ **Signal separation** is needed when a signal has been contaminated with interference, noise, or other signals. For example, imagine a device for measuring the electrical activity of a baby's heart (EKG) while still in the womb. The raw signal will likely be corrupted by the breathing and heartbeat of the mother. A filter might be used to separate these signals so that they can be individually analyzed.
- ✓ **Signal restoration** is used when a signal has been distorted in some way. For example, an audio recording made with poor equipment may be filtered to better represent the sound as it actually occurred. Another example is the deblurring of an image acquired with an improperly focused lens, or a shaky camera.

Comparison of Analog and Digital Filter

Advantages

- ✓ Digital filters can have characteristics which are not possible with analogue filters, such as a truly linear phase response.
- ✓ Unlike analogue filters, the performance of digital filters does not vary with environment changes, for example thermal variations. This eliminates the need to calibrate periodically.
- ✓ The frequency response of digital filter can be automatically adjusted if it is implemented using a programmable processor, that is why they are widely used in adaptive filters.
- ✓ Several input signals or channels can be filtered by one digital filter without the need to replicate the hardware.
- ✓ Both filtered and unfiltered data can be saved for further use.
- ✓ Advantage can be readily taken of the tremendous advantages in VLSI technology to fabricate digital filters and to make them small in size, to consume low power, and to keep the cost down.

Comparison of Analog and Digital Filter

- ✓ In practice, the precision achieved with analog filters is restricted; for example, typically a maximum of only about 60 to 70 dB stop band attenuation is possible with active filters designed with off-the-shelf components. With digital filters the precision is limited only by the word length used.
- ✓ Digital filters, in comparison, are vastly superior in the sharp frequency response that can be achieved. For example, a low-pass digital filter has a gain of 1 +/- 0.0002 from DC to 1000 hertz, and a gain of less than 0.0002 for frequencies above 1001 hertz. The entire transition occurs within only 1 hertz. But we can't except this from an analog filter, due to limitations of the electronics, such as the accuracy and stability of the resistors and capacitors. Digital filters can achieve thousands of times better performance than analog filters.
- ✓ Digital filter can be used at very low frequencies, found in many biomedical applications for example, where the use of analog filters are impractical. Also, digital filters can be made to work over a wide range of frequencies by a mere change to the sampling frequency.

Comparison of Analog and Digital Filter

Disadvantages

Speed Limitation: The maximum bandwidth of signals that digital filters can handle, in real time, is much lower than analogue filters. In real time situations, the analog-digital-analog conversion processes introduce a speed constraint on the digital filter performance. The conversion time of the ADC and the settling time of the DAC limit the highest frequency that can be processed. Further, the speed of operation of a digital filter depends on the speed of digital processor used and on the number of arithmetic operations that must be performed for the filtering algorithm, which increases as the filter response is made tighter.

Finite word length effects: Digital filters are subject to ADC noise resulting from quantizing a continuous signal, and to roundoff noise incurred during computation. With higher order recursive filters, the accumulation of roundoff noise could lead to instability.

Long Design and Development Times: The design and development times for digital filters, especially hardware development, can be much longer than analog filters.

Filter Response and Filter Kernel

- ✓ Every linear filter has an impulse response, a step response and a frequency response. Each of these responses contains complete information about the filter, but in a different form. If one of the three is specified, the other two are fixed and can be directly calculated. All three of these representations are important, because they describe how the filter will react under different circumstances.
- ✓ The most straightforward way to implement a digital filter is by convolving the input signal with the digital filter's impulse response. All possible linear filters can be made in this manner. When the impulse response is used in this way, filter designers give it a special name: **the filter kernel**.

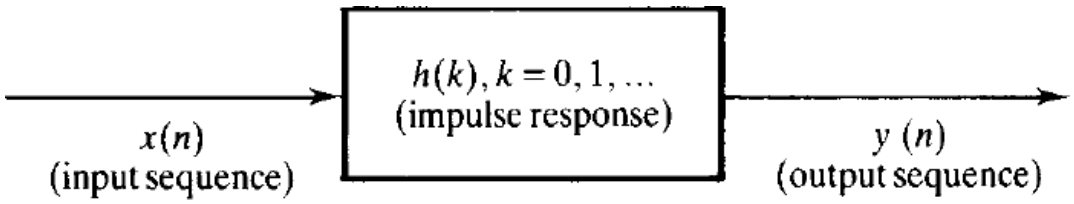


Figure: A conceptual representation of a digital filter.

The step response, (b), can be found by discrete integration of the impulse response, (a). The frequency response can be found from the impulse response by using the Fast Fourier Transform (FFT), and can be displayed either on a linear scale, (c), or in decibels, (d).

Types of Filter

Digital filters are broadly divided into two classes, namely **infinite impulse response (IIR)** and **finite impulse response (FIR)** filters. Either type of filter in its basic form, can be represented by its impulse response sequence h(k). The input and output signals to the filter are related by the convolution sum, which are given by

IIR: $$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k)$$

FIR: $$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k)$$

It is evident from these equations that, for IIR filters, the impulse response is of infinite duration whereas for FIR it is of finite duration, since h(k) for the FIR has only N values. In practice it is not feasible to compute the output of the IIR filter using above equation because the length of its impulse response is too long (infinite in theory). Instead the IIR filtering equation is expressed in a recursive form (that's why IIIR filter is also called recursive filter):

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^{N} a_k x(n-k) - \sum_{k=1}^{M} b_k y(n-k)$$

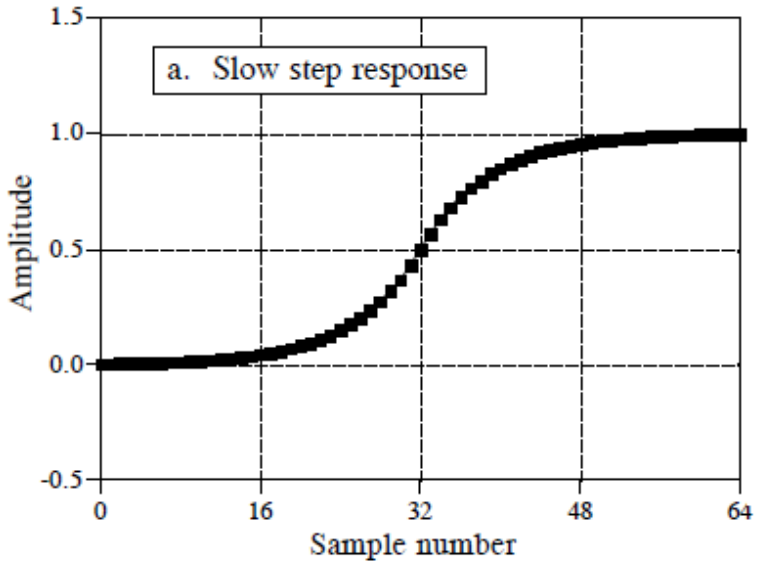
Time Domain Parameters of Filter

The step response is used to measure how well a filter performs in the time domain. Three parameters are important: (1) transition speed (risetime), shown in (a) and (b),

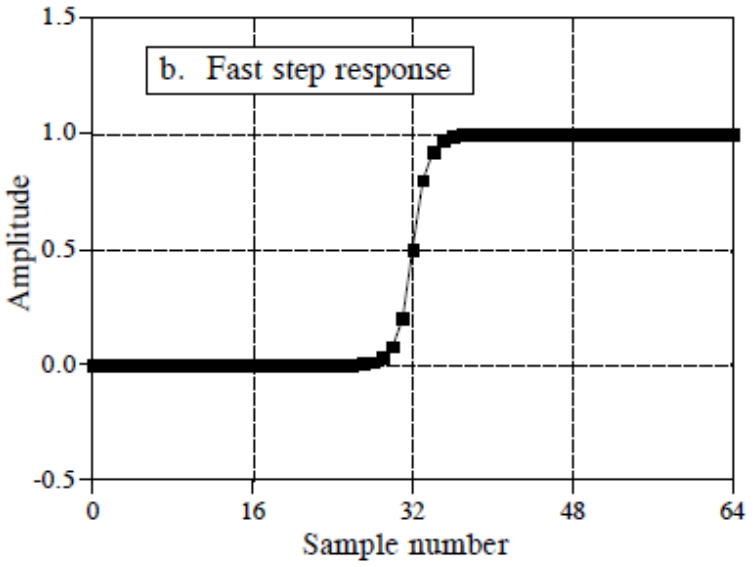
(2) overshoot, shown in (c) and (d), and

(3) phase linearity (symmetry between the top and bottom halves of the step), shown in (e) and (f).

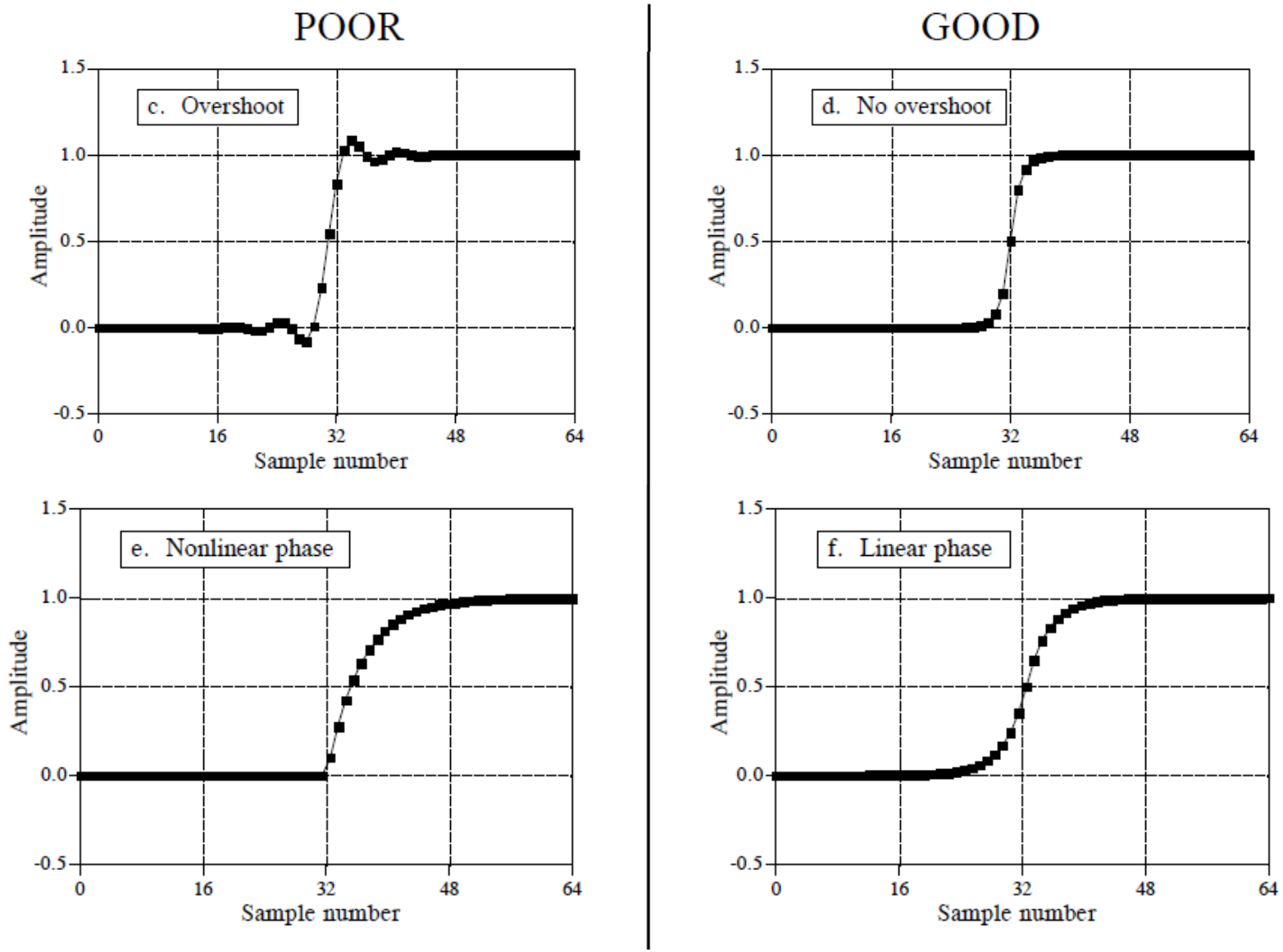
POOR



GOOD



Time Domain Parameters of Filter (Cont.)

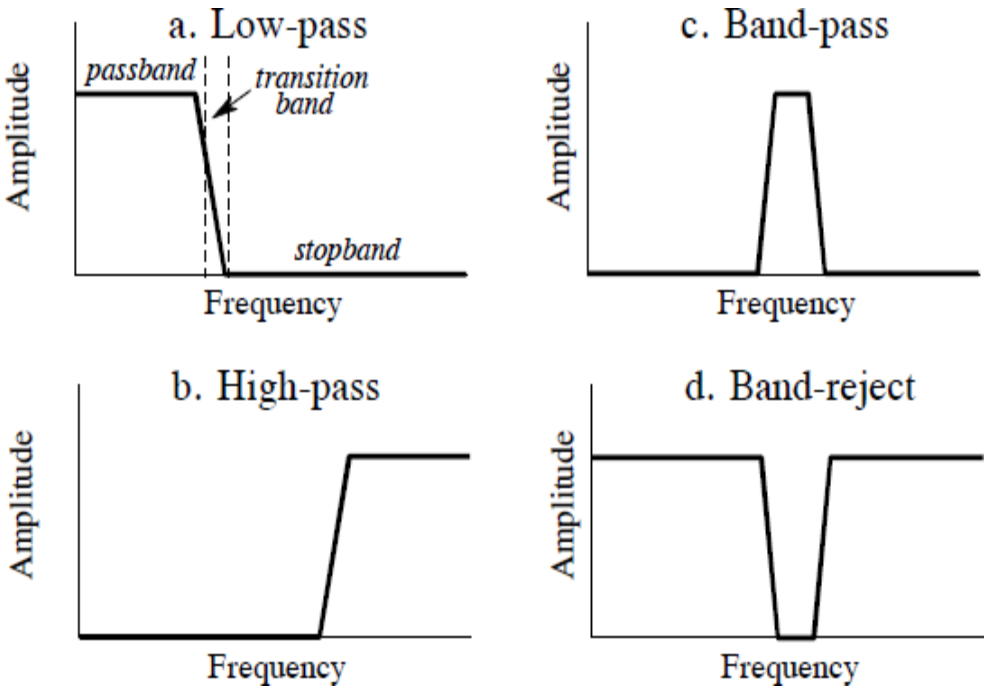


Frequency Domain Parameters of Filter

Figure shows the four basic frequency responses. The purpose of these filters is to allow some frequencies to pass unaltered, while completely blocking other frequencies.

Pass band and Stop Band: The passband refers to those frequencies that are passed, while the stopband contains those frequencies that are blocked. The transition band is between. A fast roll-off means that the transition band is very narrow.

Cutoff Frequency: The division between the passband and transition band is called the cutoff frequency. In analog filter design, the cutoff frequency is usually defined to be where the amplitude is reduced to 0.707 (i.e., -3dB). Digital filters are less standardized, and it is common to see 99%, 90%, 70.7%, and 50% amplitude levels defined to be the cutoff frequency.



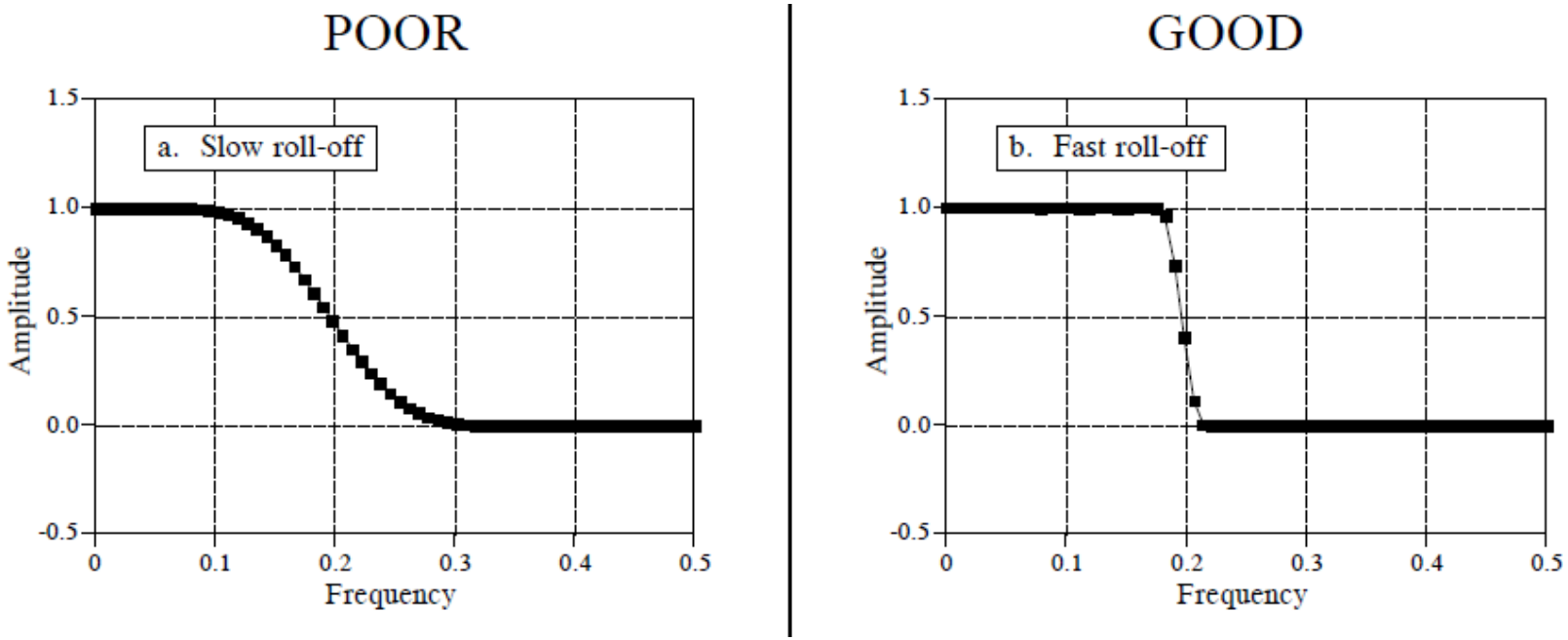
Frequency Domain Parameters of Filter (Cont.)

Parameters for evaluating *frequency domain* performance. The frequency responses shown are for low-pass filters. Three parameters are important:

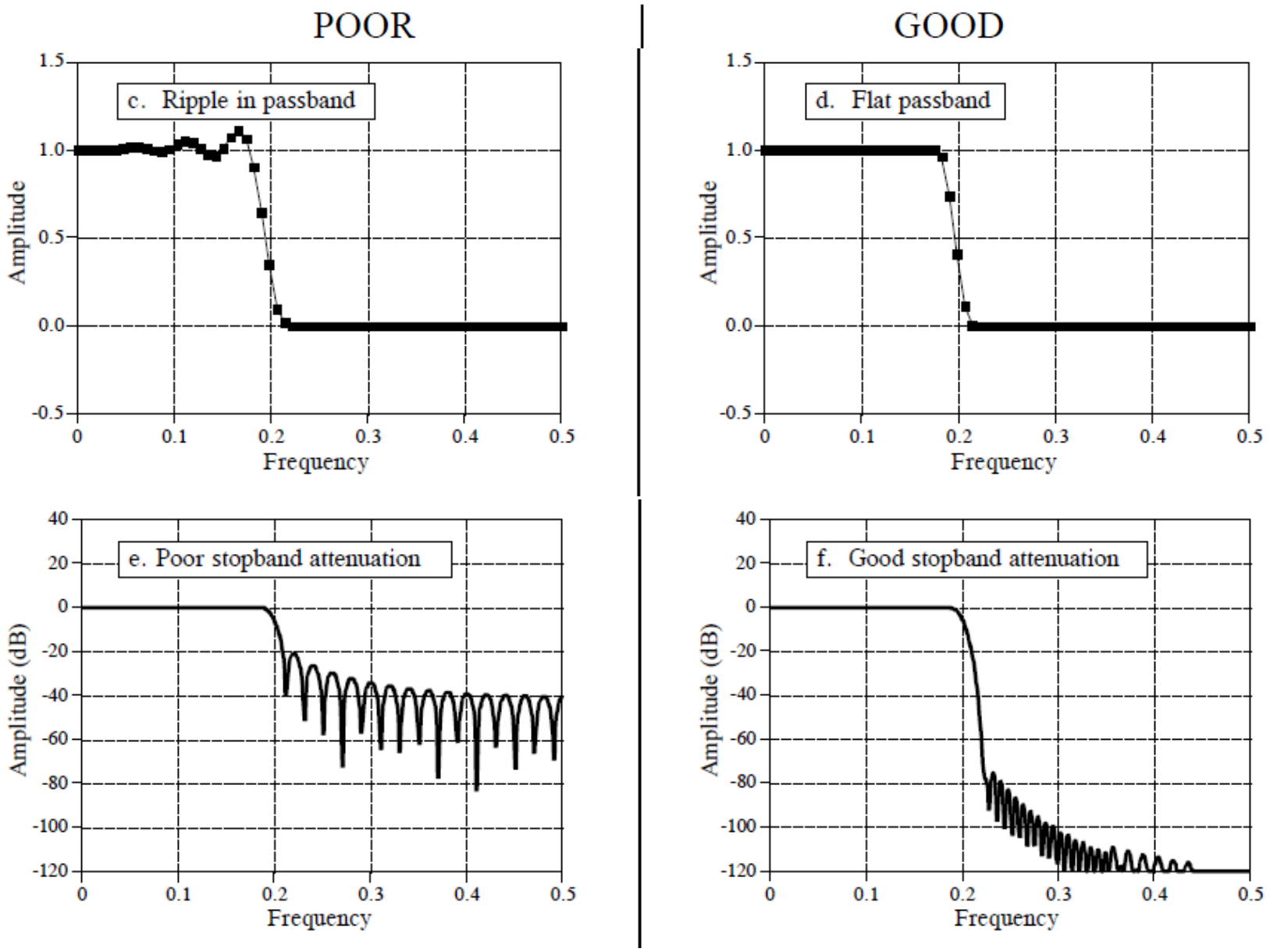
(1) roll-off sharpness, shown in (a) and (b),

(2) passband ripple, shown in (c) and (d), and

(3) stopband attenuation, shown in (e) and (f).



Frequency Domain Parameters of Filter (Cont.)



Week 15
Slide 241-253

Other filter kernel from low pass filter kernel

High-pass, band-pass and band-reject filters are designed by starting with a low-pass filter, and then converting it into the desired response. For this reason, most discussions on filter design only give examples of low-pass filters.

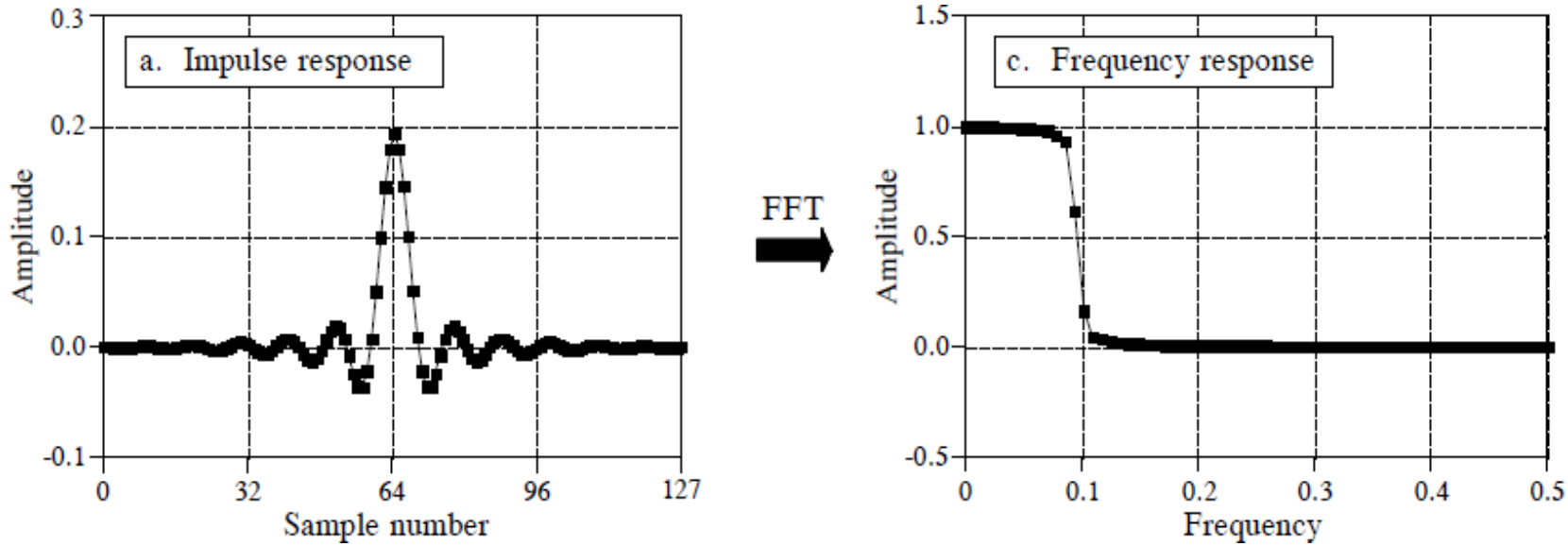
There are two methods for the low-pass to high-pass conversion:

a) spectral inversion and **b) spectral reversal**.

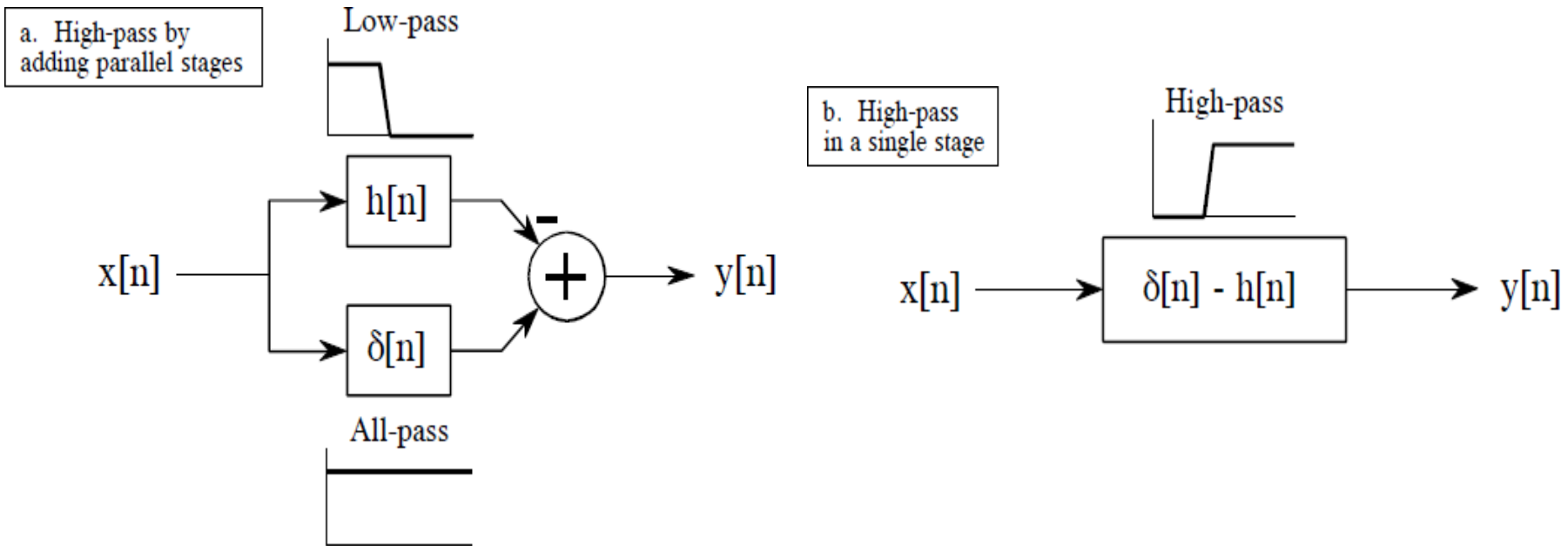
Both are equally useful.

Bandpass and bandstop filters can be obtained from lowapass and highpass filter.

Low pass filter kernel



Low pass to high pass conversion: spectral inversion



In (a), the input signal, x[n] , is applied to two systems in parallel. One of these systems is a low-pass filter, with an impulse response given by h[n] . The other system does nothing to the signal (pass all frequency), and therefore has an impulse response that is a delta function, δ[n] . The overall output, y[n] , is equal to the output of the all-pass system minus the output of the low-pass system. Since the low frequency components are subtracted from the original signal, only the high frequency components appear in the output.Thus, a high-pass filter is formed.

Low pass to high pass conversion: spectral inversion (cont.)

From the discussion of previous slide we can say-

Two things must be done to change the low-pass filter kernel into a high-pass filter kernel.

First, change the sign of each sample in the filter kernel.

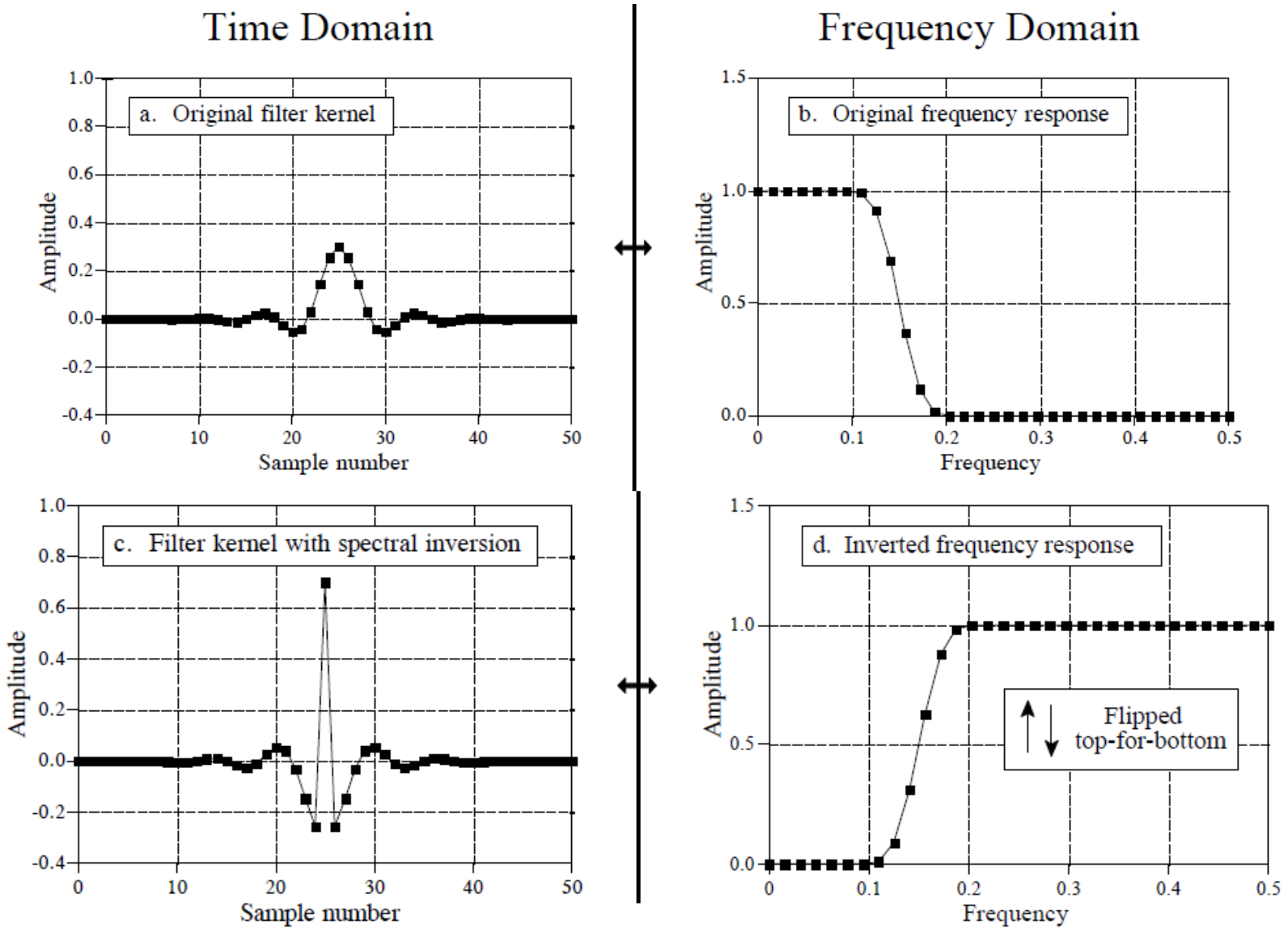
Second, add *one* to the sample at the center of symmetry. $\delta(n) - h(n)$

This results in the high-pass filter kernel shown in (c), with the frequency response shown in (d) [**see the fig in next slide**].

Spectral inversion *flips* the frequency response *top-for-bottom*, changing the passbands into stopbands, and the stopbands into passbands. In other words, it changes a filter from low-pass to high-pass, high-pass to low-pass, band-pass to band-reject, or band-reject to band-pass.

For this technique to work, the low-frequency components exiting the low-pass filter must have the same phase as the low-frequency components exiting the all-pass system. This places two restrictions on the method: (1) the original filter kernel must have left-right symmetry (i.e., a zero or linear phase), and (2) the impulse must be added at the center of symmetry.

Low pass to high pass conversion: spectral inversion (Cont.)



Low pass to high pass conversion: spectral reversal

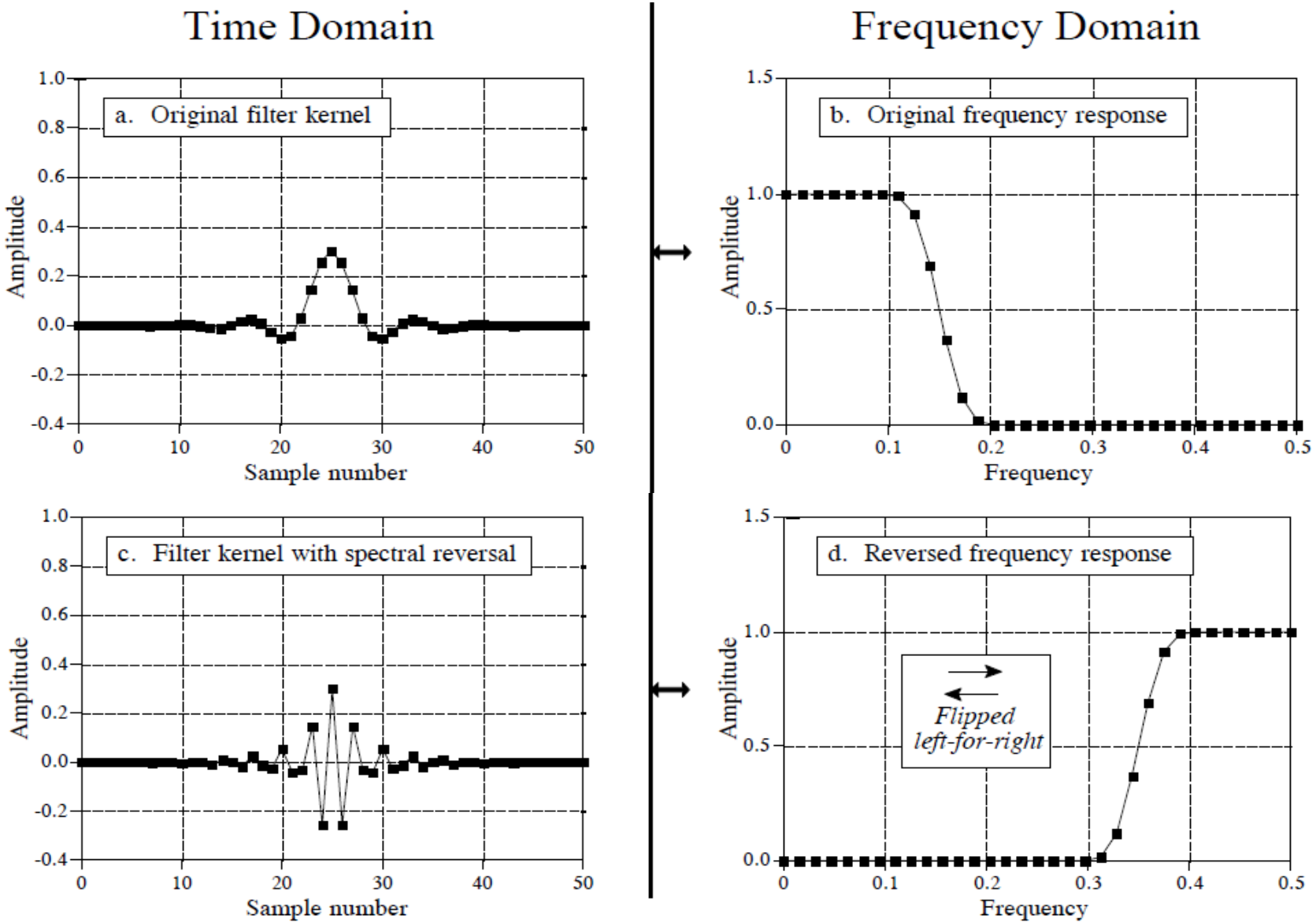
The second method for low-pass to high-pass conversion, spectral reversal, is illustrated in **Fig. (next slide)**.

The high-pass filter kernel, is formed by changing the sign of every other sample in low pass filter kernel.

Just as before, the low-pass filter kernel in (a) corresponds to the frequency response in (b). The high-pass filter kernel, (c), is formed by changing the sign of every other sample in (a). As shown in (d), this flips the frequency domain left-for-right: 0 becomes 0.5 and 0.5 becomes 0. The cutoff frequency of the example low-pass filter is 0.15, resulting in the cutoff frequency of the high-pass filter being 0.35.

Changing the sign of every other sample is equivalent to multiplying the filter kernel by a sinusoid with a frequency of 0.5. This has the effect of shifting the frequency domain by 0.5. Look at (b) and imagine the negative frequencies between -0.5 and 0 that are of mirror image of the frequencies between 0 and 0.5. The frequencies that appear in (d) are the negative frequencies from (b) shifted by 0.5.

Low pass to high pass conversion: spectral reversal (Cont.)



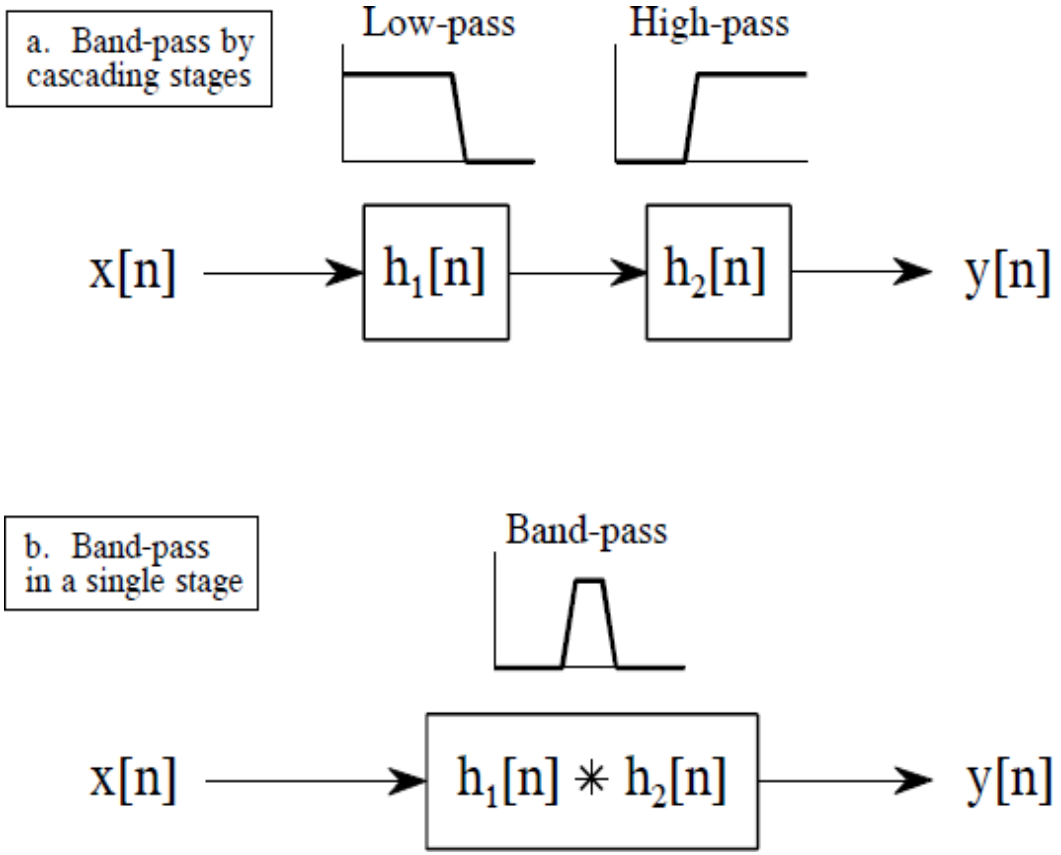
Band-pass filter from low and high pass filter kernel

Method 1: cascading of low and high pass filter kernel

As shown in (a), a band-pass filter can be formed by cascading a low-pass filter and a high-pass filter.

Method 2: Convolution of low and high pass filter kernel

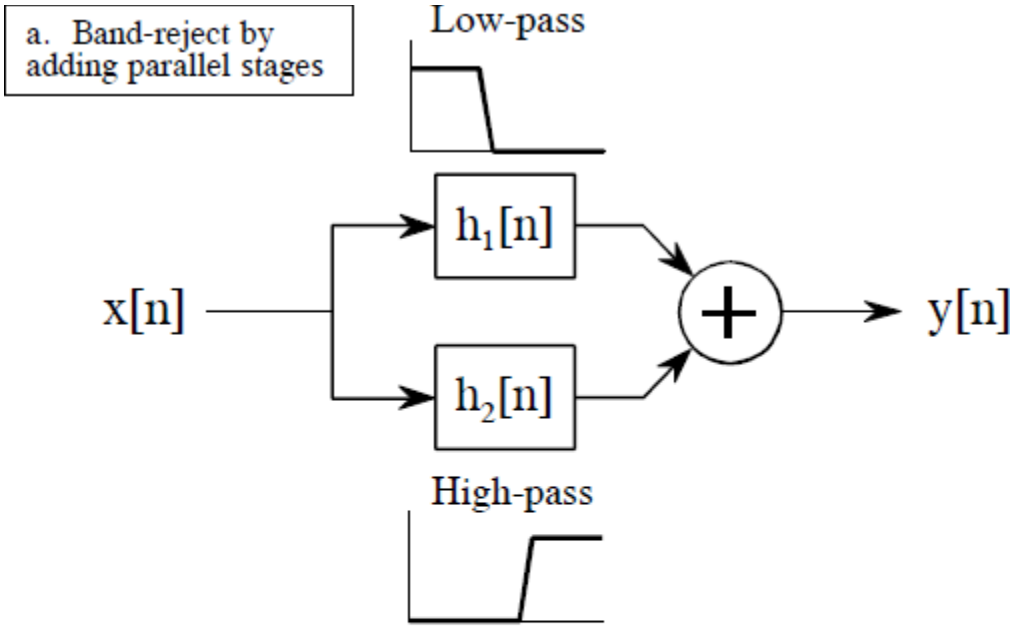
This can be reduced to a single stage, shown in (b). The filter kernel of the single stage is equal to the convolution of the low-pass and high pass filter kernels.



Band-reject filter from low and high pass filter kernel

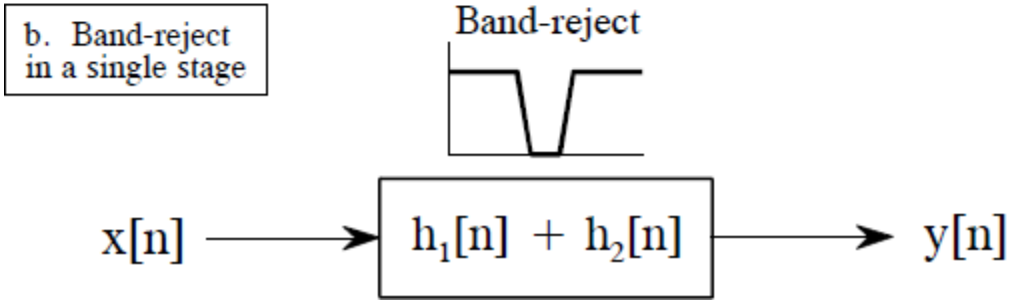
Method 1: Parallel Combination of low and high pass filter kernel

As shown in (a), a band-reject filter is formed by the parallel combination of a low-pass filter and a high-pass filter with their outputs added.



Method 2: Addition of low and high pass filter kernel

Figure (b) shows this reduced to a single stage, with the filter kernel found by adding the low-pass and high-pass filter kernels.



University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-8A
FIR Filter Design (Windowing Technique)

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ Impulse response of Ideal Filters
- ✓ Modification of ideal filter for implementation
- ✓ Different window functions
- ✓ Design equations of low-pass, high-pass, band-pass and band-stop filter.
- ✓ Problems on FIR filter design

Reference Book:

The Scientist and Engineer's Guide to Digital Signal Processing, By Steven W. Smith (2nd Edition)

Chapter-14 (Introduction to Digital Filters)

Digital Signal Processing: Fundamentals and Applications, By Li Tan, Jean Jiang

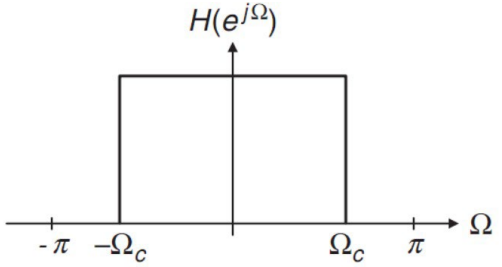
Chapter-7 (Finite Impulse Response Filter Design)

Week 16
Slide 255-282

Impulse response of Ideal low pass filter

Suppose that we want to design a lowpass filter with a cutoff frequency of ω_c, i.e. the desired frequency response will be:

$$H(\omega) = \begin{cases} 1 & |\omega| < \omega_c \\ 0 & else \end{cases}$$



Frequency response of an ideal lowpass filter.

To find the equivalent time-domain representation, we calculate the inverse discrete-time Fourier transform:

$$h[n] = \frac{1}{2\pi}\int_{-\pi}^{+\pi} H_d(\omega)e^{j\omega n}d\omega$$

$$h[n] = \frac{1}{2\pi}\int_{-\omega_c}^{+\omega_c} e^{j\omega n}d\omega = \frac{sin(n\omega_c)}{n\pi}$$

We are unable to calculate h[0] by above equation. h[0] is calculated by following formula-

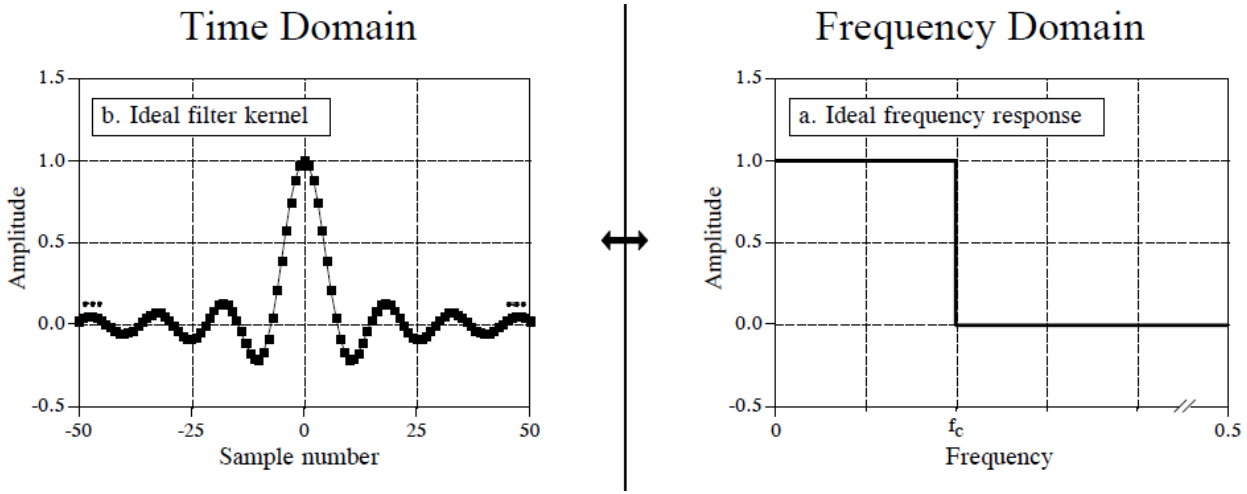
$$h[0] = \frac{1}{2\pi}\int_{-\omega_c}^{+\omega_c} e^{j\omega\times 0}\, d\omega = \frac{\omega_c}{\pi}$$

$$h(n) = \begin{cases} \dfrac{\sin(n\omega_c)}{n\pi}, & n \neq 0 \\ \dfrac{\omega_c}{\pi}, & n = 0 \end{cases}$$

Ideal low pass filter, Is it physically realizable?

In ideal lowpass filter, all frequencies below the cutoff frequency, f_c , are passed with unity amplitude, while all higher f_c frequencies are blocked. The pass band is perfectly flat, the attenuation in the stop band is infinite, and the transition between the two is infinitesimally small. Taking the Inverse Fourier Transform of this ideal frequency response produces the ideal filter kernel (impulse response) , called the sinc function, given by:

$$h(n) = \begin{cases} \dfrac{\sin(n\omega_c)}{n\pi}, & n \neq 0 \\ \dfrac{\omega_c}{\pi}, & n = 0 \end{cases}$$



Ideal low pass filter, Is it physically realizable? (Cont.)

Convolving an input signal with this filter kernel provides a perfect low-pass filter. But the problems are,

- **Noncausal**: The filter kernel (impulse response) of ideal filter is noncausal (has both positive and negative index) and hence it cannot be realized in practice.
- **Infinite Length**: It continues to both negative and positive infinity without dropping to zero amplitude. While this infinite length is not a problem for mathematics, it is a show stopper for computers.

Due to the above problems the implementation of ideal filter is not possible, i.e. it cannot be realized in practice

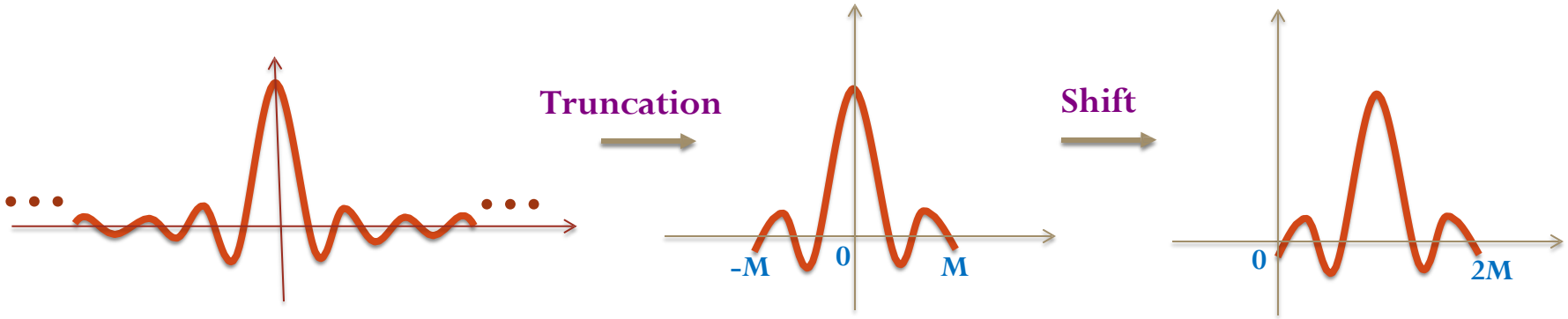
Modification in Ideal low pass filter for practical realization

To get around this problem, we will make two modifications to the ideal lowpass filter kernel:

First, it is truncated to 2M+1 points, symmetrically chosen around the main lobe, where M is an even number. All samples outside these points (-M to M) are set to zero, or simply ignored.

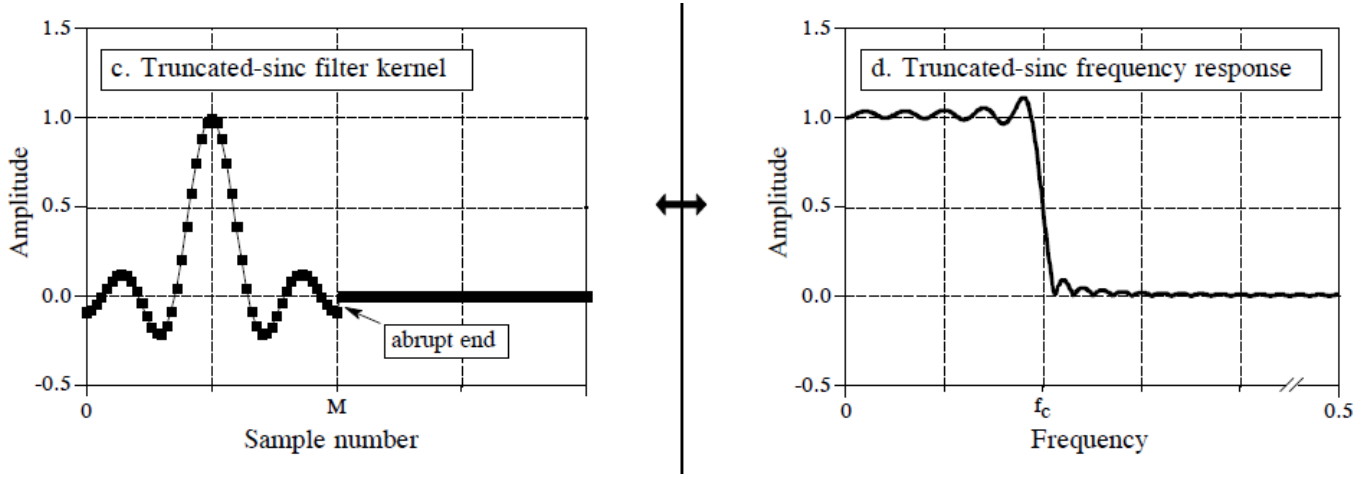
Second, the entire sequence is shifted to the right so that it runs from 0 to 2M. This allows the filter kernel to be represented using only positive indexes.

Since the modified filter kernel is only an approximation to the ideal filter kernel, it will not have an ideal frequency response.



Effect of modification in Ideal low pass filter: Gibbs Phenomenon

The abrupt discontinuity at the ends of the truncated sinc function excessive ripple in the passband and poor attenuation in the stopband. Increasing the length of the filter kernel does not reduce these problems; the discontinuity is significant no matter how long M is made. The oscillatory behavior near the band edge of the filter is called the Gibbs Phenomenon.



Window function to reduce Gibbs phenomena

To alleviate the presence of large oscillations in both the passband and the stopband, we should multiply a function with the filter kernel that contains a taper and decays towards zero gradually, instead of abruptly.

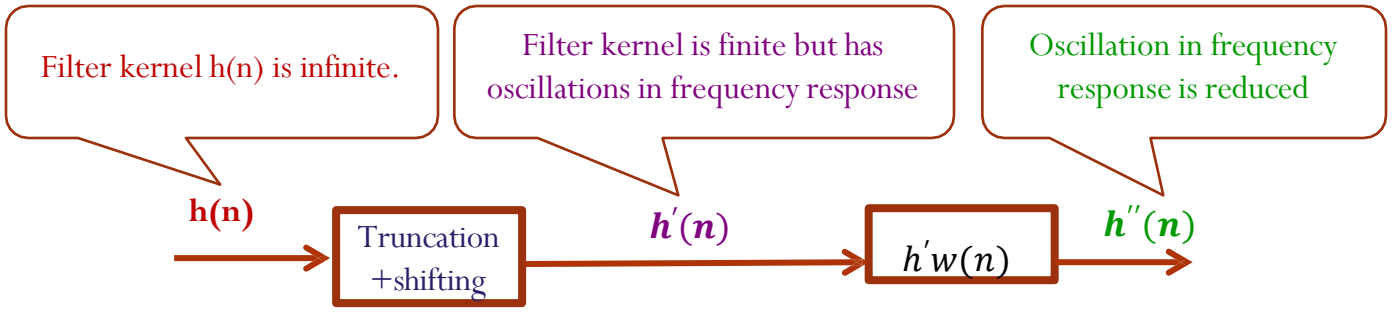
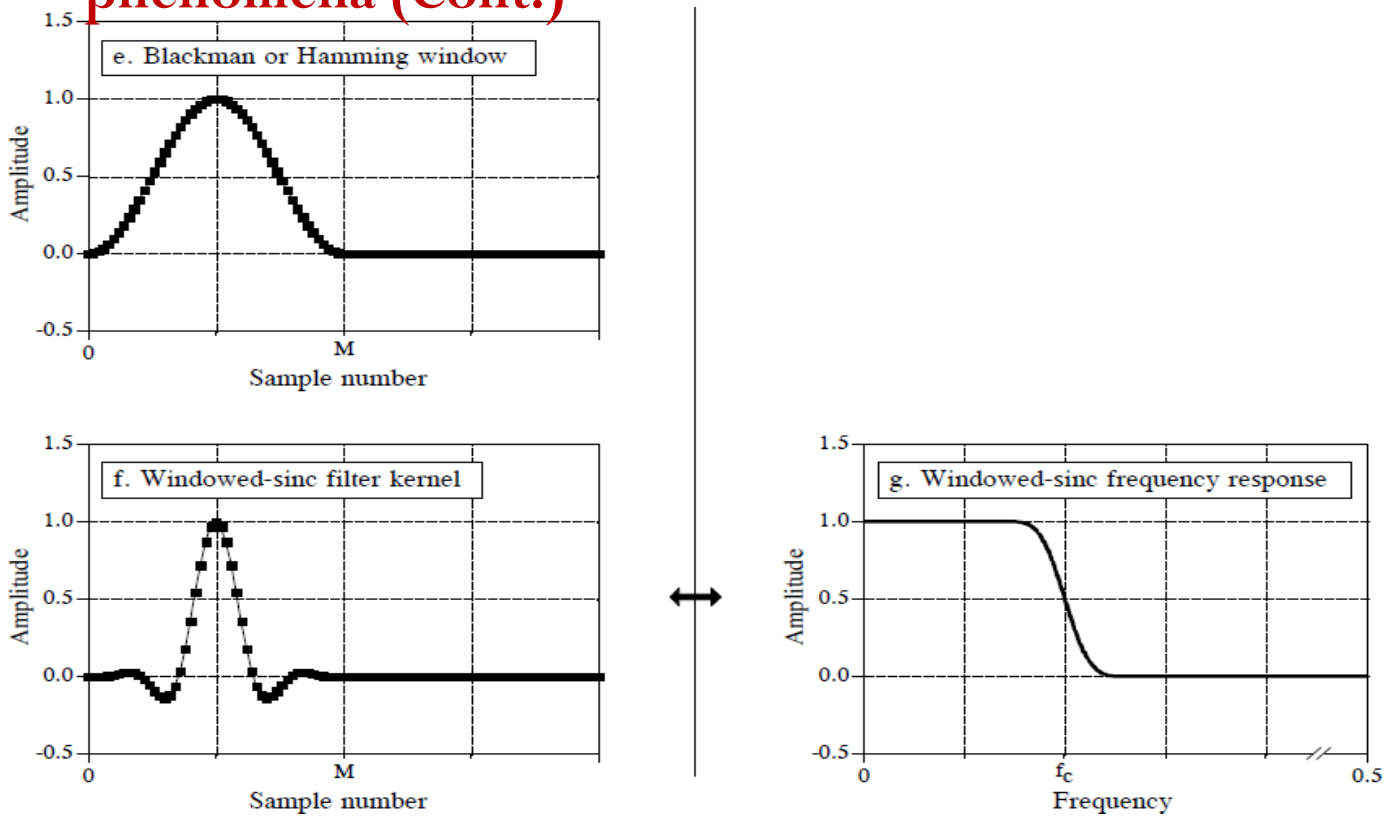


Figure (e) shows a smoothly tapered curve called a Blackman window. Multiplying the truncated-sinc, (c), by the Blackman window, (e), results in the windowed-sinc filter kernel shown in (f). The idea is to reduce the abruptness of the truncated ends and thereby improve the frequency response. Figure (g) shows this improvement. The passband is now flat, and the stopband attenuation is so good it cannot be seen in this graph. [See Figure in next slide]

Window function to reduce Gibbs phenomena (Cont.)



Different window functions

1. Rectangular window:

$$w_{rec}(n) = 1, \quad -M \leq n \leq M. \tag{7.15}$$

2. Triangular (Bartlett) window:

$$w_{tri}(n) = 1 - \frac{|n|}{M}, \quad -M \leq n \leq M. \tag{7.16}$$

3. Hanning window:

$$w_{han}(n) = 0.5 + 0.5\cos\left(\frac{n\pi}{M}\right), \quad -M \leq n \leq M. \tag{7.17}$$

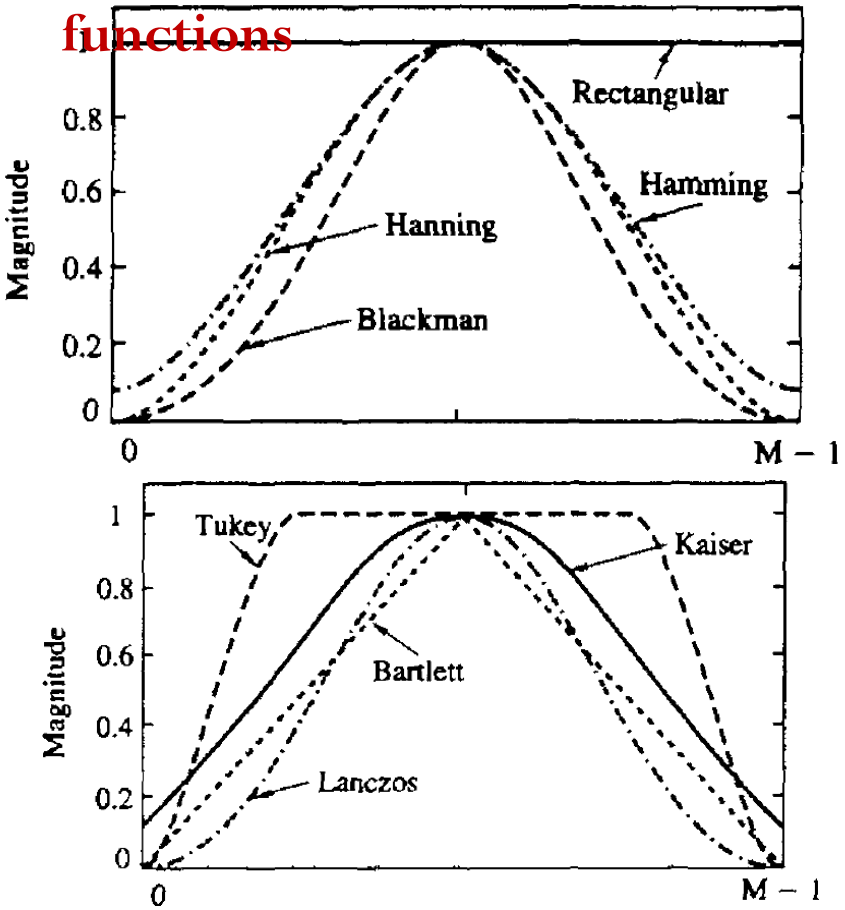
4. Hamming window:

$$w_{ham}(n) = 0.54 + 0.46\cos\left(\frac{n\pi}{M}\right), \quad -M \leq n \leq M. \tag{7.18}$$

5. Blackman window:

$$w_{black}(n) = 0.42 + 0.5\cos\left(\frac{n\pi}{M}\right) + 0.08\cos\left(\frac{2n\pi}{M}\right), \quad -M \leq n \leq M. \tag{7.19}$$

Shapes of several window functions



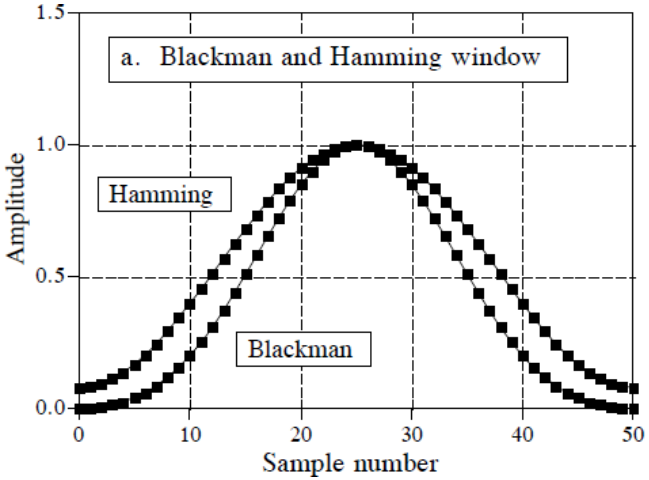
Comparison of Hamming and Blackman Window functions

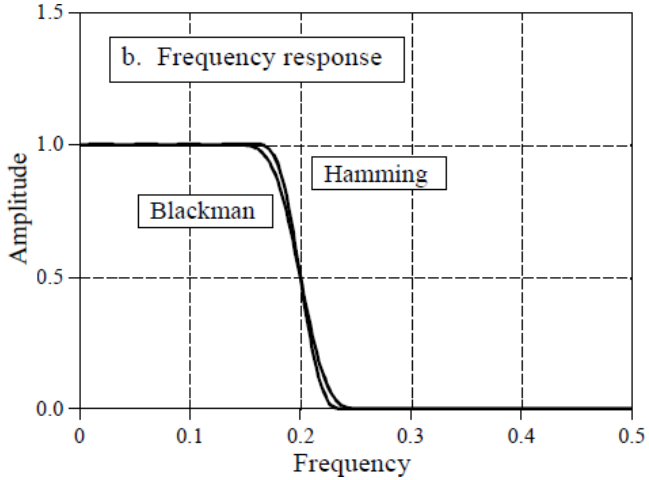
- Although several different windows are available, only two windows are worth using, the Hamming window and the Blackman window.

Hamming window: $w_{ham}(n) = 0.54 + 0.46\cos\left(\frac{n\pi}{M}\right), \quad -M \le n \le M.$

Blackman window: $w_{black}(n) = 0.42 + 0.5\cos\left(\frac{n\pi}{M}\right) + 0.08\cos\left(\frac{2n\pi}{M}\right), \quad -M \le n \le M.$

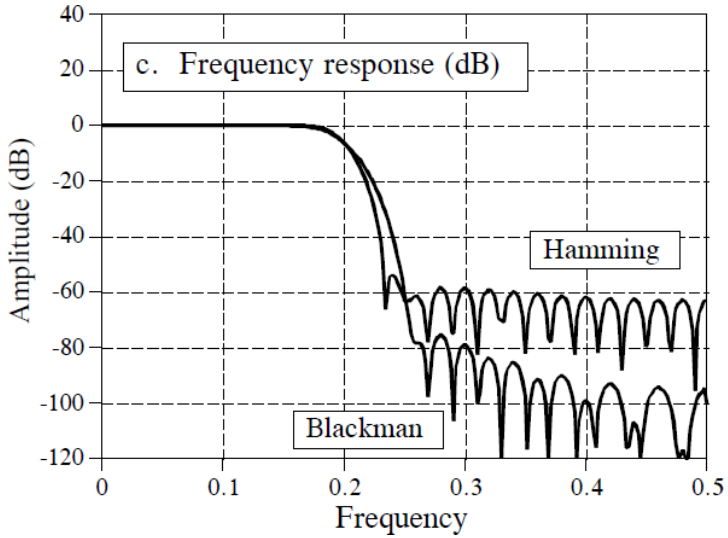
- Fig. (a) shows the shape of Hamming window and the Blackman window.
- Fig. (b) shows that, the hamming window has about 20% faster roll-off than the Blackman.





Comparison of Hamming and Blackman Window functions (Cont.)

- Fig. (c) shows that the Blackman has a better stopband attenuation. The stopband attenuation for the Blackman is -74dB (-0.02%), while the Hamming is only -53dB (-0.2%).
- Although it cannot be seen in these graphs, the Blackman has a passband ripple of only about 0.02%, while the Hamming is typically 0.2%.
- In general, the Blackman should be your first choice; a slow roll-off is easier to handle than poor stopband attenuation.



Filter lengths vs roll-off of the windowed-sinc filter

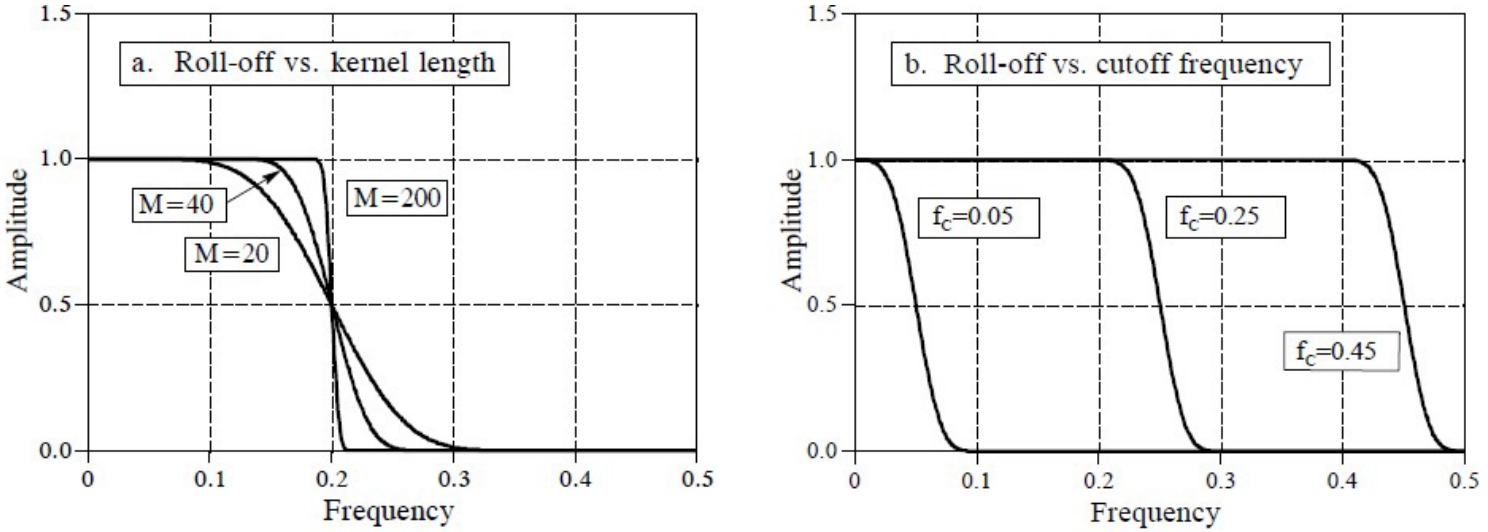
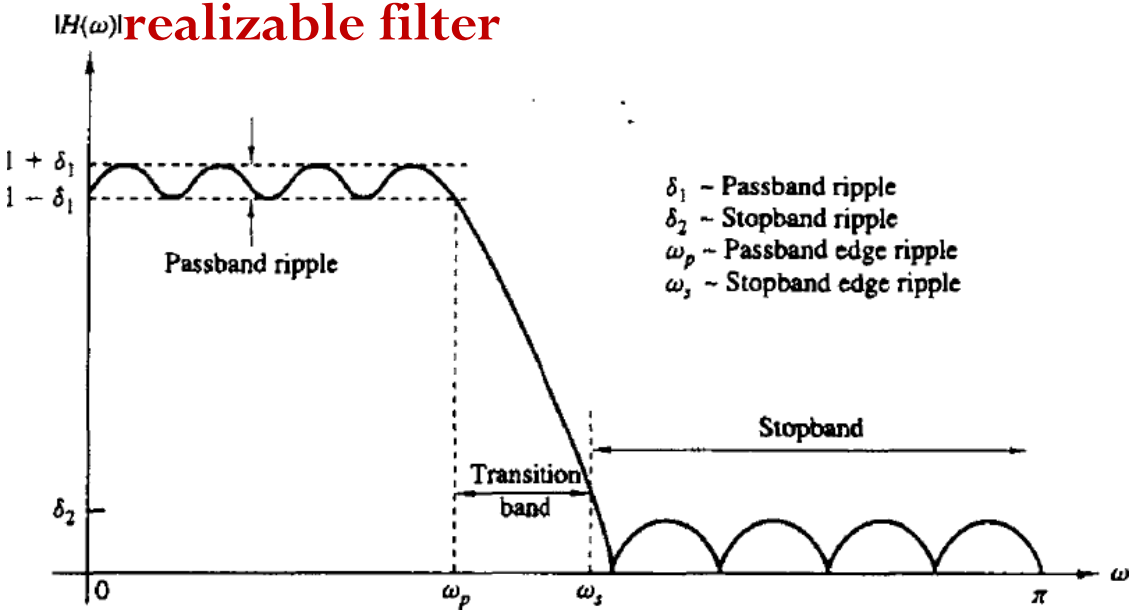


FIGURE 16-3
Filter length vs. roll-off of the windowed-sinc filter. As shown in (a), for $M = 20$, 40, and 200, the transition bandwidths are $BW = 0.2$, 0.1, and 0.02 of the sampling rate, respectively. As shown in (b), the shape of the frequency response does not change with different cutoff frequencies. In (b), $M = 60$.

Characteristics of physically realizable filter



In any filter design problem we can specify (1) the maximum tolerable pass-band ripple, (2) the maximum tolerable stop-band ripple,(3) the pass-band edge frequency (ω_p) , and (4) the stop-band edge frequency (ω_s) . Based on these specifications, we can select the filter coefficients.

Design equations (Impulse Response) of FIR Filter

Lowpass Filter

$$h(n) = \begin{cases} \frac{\Omega_c}{\pi}, & n = 0 \\ \frac{\sin(n\Omega_c)}{n\pi}, & for\ n \neq 0, \quad -M \leq n \leq M \end{cases}$$

$\Omega_c = 2\pi f_c$

Where, Ω_c is the angular cutoff frequency (Range: 0 to π) and f_c is normalized cutoff frequency (Range: 0 to 0.5)

High pass Filter

Following spectral inversion process we can convert the design equation of lowpass filter to high pass filter

$$h(n) = \begin{cases} \frac{\pi - \Omega_c}{\pi}, & n = 0 \\ -\frac{\sin(n\Omega_c)}{n\pi}, & for\ n \neq 0, \quad -M \leq n \leq M \end{cases}$$

Design equations (Impulse Response) of FIR Filter (Cont.)

Band reject Filter

The impulse response of low pass filter having angular cutoff frequency Ω_L

$$h_L(n) = \begin{cases} \frac{\Omega_L}{\pi}, & n = 0 \\ \frac{\sin(n\Omega_L)}{n\pi}, & for\ n \neq 0, \quad -M \leq n \leq M \end{cases}$$

The impulse response of high pass filter having angular cutoff frequency Ω_H

$$h_H(n) = \begin{cases} \frac{\pi - \Omega_H}{\pi}, & n = 0 \\ -\frac{\sin(n\Omega_H)}{n\pi}, & for\ n \neq 0, \quad -M \leq n \leq M \end{cases}$$

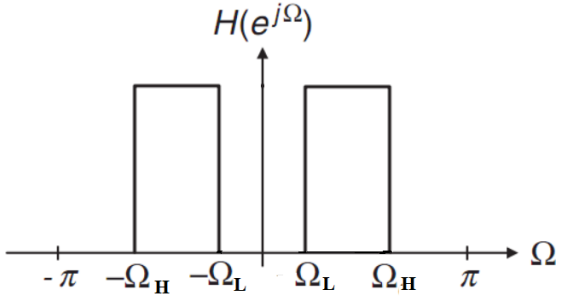
The impulse response of band reject filter having lower angular cutoff frequency $\Omega_L$and higher angular cutoff frequency Ω_H is obtained by adding $h_L(n)$ and $h_H(n)$.

$$\text{h}(n) = \begin{cases} \frac{\pi - \Omega_H - \Omega_L}{\pi}, & n = 0 \\ -\frac{\sin(n\Omega_H)}{n\pi} + \frac{\sin(n\Omega_L)}{n\pi}, & for\ n \neq 0, \ -M \leq n \leq M \end{cases}$$

Design equations (Impulse Response) of FIR Filter (Cont.)

Band pass Filter

$$H(e^{j\Omega}) = \begin{cases} 0 & |\Omega| < \Omega_L \\ 1 & \Omega_L < |\Omega| < \Omega_H \\ 0 & |\Omega| > \Omega_H \end{cases}$$



$$h(n) = \frac{1}{2\pi}\int_{-\Omega_H}^{-\Omega_L} e^{j\Omega n}d\Omega + \frac{1}{2\pi}\int_{\Omega_L}^{\Omega_H} e^{j\Omega n}d\Omega$$

$$= -\frac{\sin(\Omega_L n)}{n\pi} + \frac{\sin(\Omega_H n)}{n\pi}$$

$$h(0) = \frac{1}{2\pi}\int_{-\Omega_H}^{-\Omega_L} d\Omega + \frac{1}{2\pi}\int_{\Omega_L}^{\Omega_H} d\Omega = \frac{\Omega_H - \Omega_L}{\pi}$$

$$h(n) = \begin{cases} \frac{\Omega_H - \Omega_L}{\pi}, & n = 0 \\ -\frac{\sin(n\Omega_L)}{n\pi} + \frac{\sin(n\Omega_H)}{n\pi}, & for\ n \neq 0,\ -M \leq n \leq M \end{cases}$$

TABLE 7.1 Summary of ideal impulse responses for standard FIR filters.

Filter Type	Ideal Impulse Response $h(n)$ (noncausal FIR coefficients)
Lowpass:	$h(n) = \begin{cases} \frac{\Omega_c}{\pi} & n = 0 \\ \frac{\sin(\Omega_c n)}{n\pi} \text{ for } n \neq 0 & -M \leq n \leq M \end{cases}$
Highpass:	$h(n) = \begin{cases} \frac{\pi - \Omega_c}{\pi} & n = 0 \\ -\frac{\sin(\Omega_c n)}{n\pi} \text{ for } n \neq 0 & -M \leq n \leq M \end{cases}$
Bandpass:	$h(n) = \begin{cases} \frac{\Omega_H - \Omega_L}{\pi} & n = 0 \\ \frac{\sin(\Omega_H n)}{n\pi} - \frac{\sin(\Omega_L n)}{n\pi} \text{ for } n \neq 0 & -M \leq n \leq M \end{cases}$
Bandstop:	$h(n) = \begin{cases} \frac{\pi - \Omega_H + \Omega_L}{\pi} & n = 0 \\ -\frac{\sin(\Omega_H n)}{n\pi} + \frac{\sin(\Omega_L n)}{n\pi} \text{ for } n \neq 0 & -M \leq n \leq M \end{cases}$

Causal FIR filter coefficients: shifting $h(n)$ to the right by M samples.
Transfer function:

$$H(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots b_{2M} z^{-2M}$$

where $b_n = h(n - M)$, $n = 0, 1, \cdots, 2M$

Ref: Digital Signal Processing (Fundamentals and Application) by Li Tan

Example 7.2.

a. Calculate the filter coefficients for a 3-tap FIR lowpass filter with a cutoff frequency of 800 Hz and a sampling rate of 8,000 Hz using the Fourier transform method.

b. Determine the transfer function and difference equation of the designed FIR system.

Solution:

a. Calculating the normalized cutoff frequency leads to $\Omega_c = 2\pi f_c T_s = 2\pi \times 800/8000$
$= 0.2\pi$ radians.

Since $2M + 1 = 3$ in this case, using the equation in Table 7.1 results in

$$h(0) = \frac{\Omega_c}{\pi} \quad \text{for } n = 0 \qquad h(0) = \frac{0.2\pi}{\pi} = 0.2$$

$$h(n) = \frac{\sin(\Omega_c n)}{n\pi} = \frac{\sin(0.2\pi n)}{n\pi}, \quad \text{for } n \neq 1. \qquad h(1) = \frac{\sin[0.2\pi \times 1]}{1 \times \pi} = 0.1871.$$

Using the symmetry leads to

$$h(-1) = h(1) = 0.1871.$$

Thus delaying $h(n)$ by $M = 1$ sample using Equation $b_n = h(n - M)$

$b_0 = h(-1) = 0.1871$ $b_1 = h(0) = 0.2$ and $b_2 = h(1) = 0.1871.$

b. The transfer function is achieved as

$$H(z) = 0.1871 + 0.2z^{-1} + 0.1871z^{-2}.$$

$$\frac{Y(z)}{X(z)} = H(z) = 0.1871 + 0.2z^{-1} + 0.1817z^{-2}.$$

Multiplying $X(z)$ leads to

$$Y(z) = 0.1871X(z) + 0.2z^{-1}X(z) + 0.1871z^{-2}X(z).$$

Applying the inverse z-transform on both sides, the difference equation is yielded as

$$y(n) = 0.1871x(n) + 0.2x(n-1) + 0.1871x(n-2).$$

Example 7.3.

a. Calculate the filter coefficients for a 5-tap FIR bandpass filter with a lower cutoff frequency of 2,000 Hz and an upper cutoff frequency of 2,400 Hz at a sampling rate of 8,000 Hz.

Solution:

a. Calculating the normalized cutoff frequencies leads to

$$\Omega_L = 2\pi f_L/f_s = 2\pi \times 2000/8000 = 0.5\pi \text{ radians}$$
$$\Omega_H = 2\pi f_H/f_s = 2\pi \times 2400/8000 = 0.6\pi \text{ radians.}$$

Since $2M + 1 = 5$ in this case, using the equation in Table 7.1 yields

$$h(n) = \begin{cases} \frac{\Omega_H - \Omega_L}{\pi} & n = 0 \\ \frac{\sin(\Omega_H n)}{n\pi} - \frac{\sin(\Omega_L n)}{n\pi} & n \neq 0 \quad -2 \leq n \leq 2 \end{cases}$$

Calculations for noncausal FIR coefficients are listed as

$$h(0) = \frac{\Omega_H - \Omega_L}{\pi} = \frac{0.6\pi - 0.5\pi}{\pi} = 0.1.$$

$$h(1) = \frac{\sin[0.6\pi \times 1]}{1 \times \pi} - \frac{\sin[0.5\pi \times 1]}{1 \times \pi} = -0.01558$$

$$h(2) = \frac{\sin[0.6\pi \times 2]}{2 \times \pi} - \frac{\sin[0.5\pi \times 2]}{2 \times \pi} = -0.09355.$$

Using the symmetry leads to

$$h(-1) = h(1) = -0.01558$$
$$h(-2) = h(2) = -0.09355.$$

Thus, delaying $h(n)$ by $M = 2$ samples gives

$$b_0 = b_4 = -0.09355,$$
$$b_1 = b_3 = -0.01558, \text{ and } b_2 = 0.1.$$

Example 7.6.

a. Design a 5-tap FIR band reject filter with a lower cutoff frequency of 2,000 Hz, an upper cutoff frequency of 2,400 Hz, and a sampling rate of 8,000 Hz using the Hamming window method.

b. Determine the transfer function.

Solution:

a. Calculating the normalized cutoff frequencies leads to

$$\Omega_L = 2\pi f_L T = 2\pi \times 2000/8000 = 0.5\pi \text{ radians}$$
$$\Omega_H = 2\pi f_H T = 2\pi \times 2400/8000 = 0.6\pi \text{ radians}.$$

Since $2M + 1 = 5$ in this case,

$$h(n) = \begin{cases} \frac{\pi - \Omega_H + \Omega_L}{\pi} & n = 0 \\ -\frac{\sin(\Omega_H n)}{n\pi} + \frac{\sin(\Omega_L n)}{n\pi} & n \neq 0 \end{cases} \quad -2 \leq n \leq 2.$$

When $n = 0$, we have

$$h(0) = \frac{\pi - \Omega_H + \Omega_L}{\pi} = \frac{\pi - 0.6\pi + 0.5\pi}{\pi} = 0.9.$$

$$h(1) = \frac{\sin[0.5\pi \times 1]}{1 \times \pi} - \frac{\sin[0.6\pi \times 1]}{1 \times \pi} = 0.01558$$

$$h(2) = \frac{\sin[0.5\pi \times 2]}{2 \times \pi} - \frac{\sin[0.6\pi \times 2]}{2 \times \pi} = 0.09355.$$

Using the symmetry leads to

$$h(-1) = h(1) = 0.01558$$
$$h(-2) = h(2) = 0.09355.$$

Applying the Hamming window function in Equation (7.18),

$$w_{ham}(0) = 0.54 + 0.46\cos\left(\frac{0 \times \pi}{2}\right) = 1.0$$

$$w_{ham}(1) = 0.54 + 0.46\cos\left(\frac{1 \times \pi}{2}\right) = 0.54$$

$$w_{ham}(2) = 0.54 + 0.46\cos\left(\frac{2 \times \pi}{2}\right) = 0.08.$$

Using the symmetry of the window function gives

$$w_{ham}(-1) = w_{ham}(1) = 0.54$$
$$w_{ham}(-2) = w_{ham}(2) = 0.08.$$

The windowed impulse response is calculated as

$$h_w(0) = h(0)w_{ham}(0) = 0.9 \times 1 = 0.9$$
$$h_w(1) = h(1)w_{ham}(1) = 0.01558 \times 0.54 = 0.00841$$
$$h_w(2) = h(2)w_{ham}(2) = 0.09355 \times 0.08 = 0.00748$$
$$h_w(-1) = h(-1)w_{ham}(-1) = 0.00841$$
$$h_w(-2) = h(-2)w_{ham}(-2) = 0.00748.$$

Thus, delaying $h_w(n)$ by $M = 2$ samples gives

$$b_0 = b_4 = 0.00748,\ b_1 = b_3 = 0.00841,\ \text{and}\ b_2 = 0.9.$$

b. The transfer function is achieved as

$$H(z) = 0.00748 + 0.00841z^{-1} + 0.9z^{-2} + 0.00841z^{-3} + 0.00748z^{-4}.$$

TABLE 7.7 **FIR filter length estimation using window functions (normalized transition width $\Delta f = |f_{stop} - f_{pass}|/f_s$).**

Window Type	Window Function $w(n)$, $-M \le n \le M$	Window Length, N	Passband Ripple (dB)	Stopband Attenuation (dB)
Rectangular	1	$N = 0.9/\Delta f$	0.7416	21
Hanning	$0.5 + 0.5\cos\left(\frac{\pi n}{M}\right)$	$N = 3.1/\Delta f$	0.0546	44
Hamming	$0.54 + 0.46\cos\left(\frac{\pi n}{M}\right)$	$N = 3.3/\Delta f$	0.0194	53
Blackman	$0.42 + 0.5\cos\left(\frac{n\pi}{M}\right) + 0.08\cos\left(\frac{2n\pi}{M}\right)$	$N = 5.5/\Delta f$	0.0017	74

Example 7.9.

a. Design a highpass FIR filter with the following specifications:

Stopband = 0–1,500 Hz Passband = 2,500–4,000 Hz Stopband attenuation = 40 dB

Passband ripple = 0.1 dB Sampling rate = 8,000 Hz

Solution:

a. Based on the specifications, the Hanning window will do the job, since it has a passband ripple of 0.0546 dB and a stopband attenuation of 44 dB.

Then $\Delta f = |1500 - 2500|/8000 = 0.125 \qquad N = 3.1/\Delta f = 24.2$. Choose $N = 25$.

Hence, we choose 25 filter coefficients using the Hanning window method. The cutoff frequency is $(1500 + 2500)/2 = 2000$ Hz. The normal-ized cutoff frequency can be easily found as

$$\Omega_c = \frac{2000 \times 2\pi}{8000} = 0.5\pi \text{ radians.}$$

And notice that $2M + 1 = 25$.

$$h(n) = \begin{cases} \dfrac{\pi - 0.5\pi}{\pi}, & n = 0 \\ -\dfrac{\sin(0.5\pi n)}{n\pi}, & for\ n \neq 0, \quad -12 \leq n \leq 12 \end{cases}$$

$$w_{han}(n) = 0.5 + 0.5\cos\left(\frac{n\pi}{12}\right), -12 \leq M \leq 12$$

h(0)=0.5

$$w_{han}(0) = 0.5 + 0.5\cos\left(\frac{0 \times \pi}{12}\right) = 1 \qquad h_w(0) = h(0) \times w_{han}(0) = 0.5$$

$$h(1) = -\frac{\sin(0.5\pi)}{\pi} = -0.3183 \qquad w_{han}(1) = 0.5 + 0.5\cos\left(\frac{1 \times \pi}{12}\right) = 0.982963 \qquad h_w(1) = -0.312877$$

Similarly, we can calculate other coefficients of filter

n	h(n)	$w_{han}(n)$	$h_w(n)$
2	0	0.933012	0
3	0.106103	0.853553	0.090565
4	0	0.749999	0
5	-0.06366	0.629408	-0.04007
6	0	0.499998	0
7	0.045473	0.370588	0.016852
8	0	0.249998	0
9	-0.03537	0.146445	-0.00518
10	0	0.066986	0
11	0.028937	0.017036	0.000493
12	0	0	0

After shifting the coefficients by 12 positions and considering the symmetry property, the coefficients of desired filter is listed below

FIR Filter Coefficients (Hanning window)

$b_0 = b_{24} = 0.000000$
$b_2 = b_{22} = 0.000000$
$b_4 = b_{20} = 0.000000$
$b_6 = b_{18} = 0.000000$
$b_8 = b_{16} = 0.0000000$
$b_{10} = b_{14} = 0.000000$
$b_{12} = 0.500000$

$b_1 = b_{23} = 0.000493$
$b_3 = b_{21} = -0.005179$
$b_5 = b_{19} = 0.016852$
$b_7 = b_{17} = -0.040069$
$b_9 = b_{15} = 0.090565$
$b_{11} = b_{13} = -0.312887$

Example 7.11.

a. Design a bandstop FIR filter with the following specifications:

Lower cutoff frequency = 1,250 Hz Upper cutoff frequency = 2,850 Hz
Lower transition width = 1,500 Hz Upper transition width = 1,300 Hz
Stopband attenuation = 60 dB Passband ripple = 0.02 dB Sampling rate = 8,000 Hz

Solution:

a. We can directly compute the normalized transition widths:

$$\Delta f_1 = 1500/8000 = 0.1875, \text{ and } \Delta f_2 = 1300/8000 = 0.1625.$$

The filter lengths are determined using the Blackman window as:

$$N_1 = 5.5/0.1875 = 29.33, \text{ and } N_2 = 5.5/0.1625 = 33.8.$$ We choose an odd number $N = 35$.

The normalized lower and upper cutoff frequencies are calculated as:

$$\Omega_L = \frac{2\pi \times 1250}{8000} = 0.3125\pi \text{ radian and } \quad \Omega_H = \frac{2\pi \times 2850}{8000} = 0.7125\pi \text{ radians,}$$

and $N = 2M + 1 = 35$.

$$h(n) = \begin{cases} \frac{\pi - 0.7125\pi + 0.3125\pi}{\pi}, & n = 0 \\ -\frac{\sin(0.7125\pi n)}{n\pi} + \frac{\sin(0.3125\pi n)}{n\pi}, & n \neq 0,\ -17 \leq n \leq 17 \end{cases}$$

$$w_{blak}(n) = 0.42 + 0.5\cos\left(\frac{n\pi}{17}\right) + 0.08\cos\left(\frac{2n\pi}{17}\right), \quad -17 \leq M \leq 17$$

h(0)=0.6 $\quad W(0) = 0.42 + 0.5\cos\left(\frac{0\times\pi}{17}\right) + 0.08cos\left(\frac{2\times0\times\pi}{17}\right) = 1 \quad h_w(0) = h(0) \times w(0) = 0.6$

$$h_(1) = -\frac{\sin(0.7125\pi)}{\pi} + \frac{\sin(0.3125\pi)}{\pi} = 0.014692$$

$$W(1) = 0.42 + 0.5\cos\left(\frac{\pi}{17}\right) + 0.08cos\left(\frac{2\times\pi}{17}\right) = 0.98608 \quad h_w(1) = 0.01449$$

Similarly, we can calculate other coefficients of filter

n	h(n)	$w(n)$	$h_w(n)$
1	0.014692	0.986084	0.014488
2	0.301797	0.945357	0.285306
3	-0.02372	0.880767	-0.02089
4	-0.0924	0.796885	-0.07363
5	0	0.699423	0
6	-0.06064	0.594657	-0.03606
7	0.02348	0.488812	0.011477
8	0.071977	0.387495	0.027891
9	-0.01439	0.295226	-0.00425

n	h(n)	$w(n)$	$h_w(n)$
10	0	0.215149	0
11	-0.01424	0.148919	-0.00212
12	-0.04495	0.096788	-0.00435
13	0.022759	0.057876	0.001317
14	0.022788	0.03055	0.000696
15	0	0.012884	0
16	0.01892	0.003111	0.0000589
17	-0.02205	0	0

After shifting the coefficients by 17 positions and considering the symmetry property, the coefficients of desired filter is listed below

n	$h_w(n)$	b
0	0.6	$b_{17} = 0.6$
1	0.014488	$b_{16} = b_{18} = 0.014488$
2	0.285306	$b_{15} = b_{19} = 0.285306$
3	-0.02089	$b_{14} = b_{20} = -0.02089$
4	-0.07363	$b_{13} = b_{21} = -0.07363$
5	0	$b_{12} = b_{22} = 0$
6	-0.03606	$b_{11} = b_{23} = -0.036062$
7	0.011477	$b_{10} = b_{24} = 0.011477$
8	0.027891	$b_9 = b_{25} = 0.027891$
9	-0.00425	$b_8 = b_{26} = -0.00425$
10	0	$b_7 = b_{27} = 0$
11	-0.00212	$b_6 = b_{28} = -0.00212$
12	-0.00435	$b_5 = b_{29} = -0.00435$
13	0.001317	$b_4 = b_{30} = 0.001317$
14	0.000696	$b_3 = b_{31} = 0.000696$
15	0	$b_2 = b_{32} = 0$
16	0.0000589	$b_1 = b_{33} = 0.0000589$
17	0	$b_0 = b_{34} = 0$

University of Global Village (UGV), Barishal
Dept. of Electrical and Electronic Engineering (EEE)

Segment-8A
IIR Filter Design

Course Code: EEE-307/CSE-309
Course Title: Digital Signal Processing

Prepared By
Noor Md Shahriar
Senior Lecturer, Dept. of EEE, UGV

Contents

- ✓ IIR Filter.
- ✓ Design of IIR Filter from analog filter: BLT Method.
- ✓ Some simple and intuitive IIR Filters: Leaky integrator, Resonator, DC removal, Hum removal.

References:

Digital Signal Processing: Fundamentals and Applications, By Li Tan, Jean Jiang

Chapter-8 (Infinite Impulse Response Filter Design)

Lecture slides of Online course on coursera.org (Digital Signal Processing by École Polytechnique Fédérale de Lausanne)

Week 17
Slide 286-303

IIR filter

- Digital filters that has impulse response of infinite length is known as IIR filter. Their impulse responses are composed of decaying exponentials.
- Such filter is also called recursive filter, because the IIR filter output y(n) depends not only on the current input x(n) and past inputs x(n-1), …, bust also on the past output(s) y(n-1), … (recursive terms). Its transfer function is a ratio of the numerator polynomial over the denominator, and its impulse response has an infinite number of terms.

$$y(n) = b_0 x(n) + b_1 x(n-1) + \cdots + b_M x(n-M) - a_1 y(n-1) - \cdots - a_N y(n-N).$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}}$$

- Since the transfer function has the denominator polynomial, the pole(s) of a designed IIR filter must be inside the unit circle on the z-plane to ensure its stability.
- Compared with the finite impulse response (FIR) filter, the IIR filter offers a much smaller filter size. Hence, the filter operation requires a fewer number of computations, but the linear phase is not easily obtained. The IIR filter is thus preferred when a small filter size is called but the application does not require a linear phase.

Design of IIR Filters from Analog Filters

Analog filter design is a mature and well-developed field, so it is not surprising that we begin the design of a digital filter in the analog domain and then convert the design into the digital domain. Following are the methods for converting an analog filter into a digital filter:

- IIR filter design by Approximation of Derivatives.
- IIR Filter Design by Impulse Invariance.
- IIR Filter Design by the Bilinear Transformation.

Analog filter using lowpass prototype transformation

Low pass prototype

$$H_P(s) = \frac{1}{s+1}$$

This is the transfer function of low-pass prototype with a cutoff frequency of 1 radian per second

TABLE : Analog lowpass prototype transformations.

Filter Type	Prototype Transformation
Lowpass	$\frac{s}{\omega_c}$, ω_c is the cutoff frequency
Highpass	$\frac{\omega_c}{s}$, ω_c is the cutoff frequency
Bandpass	$\frac{s^2+\omega_0^2}{sW}$, $\omega_0 = \sqrt{\omega_l \omega_h}$, $W = \omega_h - \omega_l$
Bandstop	$\frac{sW}{s^2+\omega_0^2}$, $\omega_0 = \sqrt{\omega_l \omega_h}$, $W = \omega_h - \omega_l$

Example 8.2.

Given a lowpass prototype $H_P(s) = \dfrac{1}{s+1}$,

a. Determine each of the following analog filters and plot their magnitude responses from 0 to 200 radians per second.

1. The highpass filter with a cutoff frequency of 40 radians per second.
2. The bandpass filter with a center frequency of 100 radians per second and bandwidth of 20 radians per second.

Solution:

1. Applying the lowpass prototype transformation by substituting $s = 40/s$ into the lowpass prototype, we have an analog highpass filter as

$$H_{HP}(s) = \frac{1}{\frac{40}{s}+1} = \frac{s}{s+40}.$$

2. Similarly, substituting the lowpass-to-bandpass transformation $s = (s^2 + 100)/(20s)$ into the lowpass prototype leads to

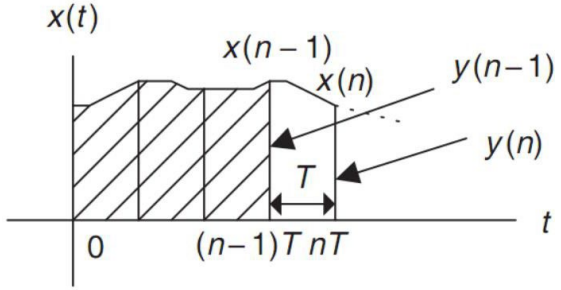
$$H_{BP}(s) = \frac{1}{\frac{s^2+100}{20s}+1} = \frac{20s}{s^2+20s+100}.$$

IIR Filter Design by the Bilinear Transformation

BLT and frequency Warping

The area under the curve can be determined using the following integration:

$$y(t) = \int_0^t x(t)dt,$$



Laplace transform of above equation and Laplace transfer function:

$$Y(s) = \frac{X(s)}{s} \longrightarrow G(s) = \frac{Y(s)}{X(s)} = \frac{1}{s} \quad \ldots\ldots\ldots (1)$$

Now area under curve by numerical integration method: $y(n) = y(n-1) + \frac{x(n) + x(n-1)}{2} T$

z-transform of above equation and z-transfer function:

$$Y(z) = z^{-1}Y(z) + \frac{T}{2}\left(X(z) + z^{-1}X(z)\right) \longrightarrow H(z) = \frac{Y(z)}{X(z)} = \frac{T}{2}\frac{1+z^{-1}}{1-z^{-1}}. \quad \ldots\ldots\ldots (2)$$

Comparing equation (1) and (2), we can write

$$\frac{1}{s} = \frac{T}{2}\frac{1+z^{-1}}{1-z^{-1}} = \frac{T}{2}\frac{z+1}{z-1}. \longrightarrow s = \frac{2}{T}\frac{z-1}{z+1}.$$

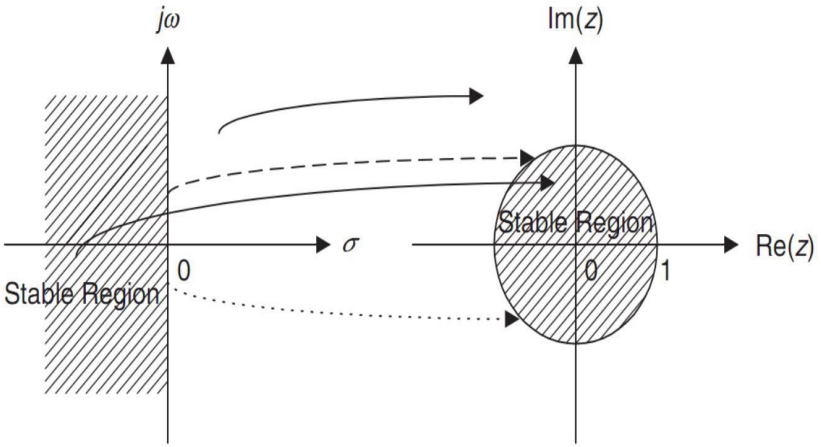
This equation is known as bilinear transformation.

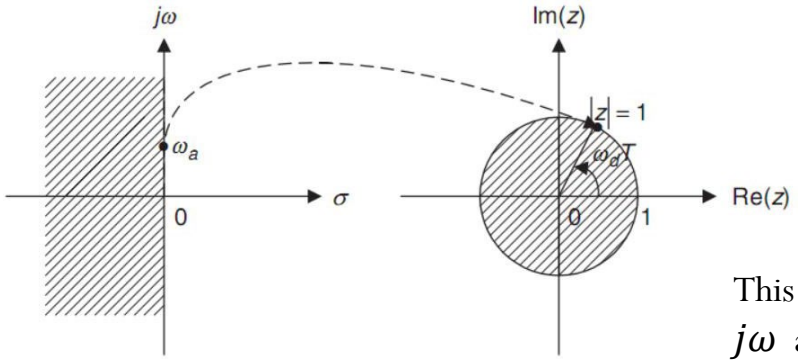
The BLT method is a mapping or transformation of points from the s-plane to the z-plane.

IIR Filter Design by the Bilinear Transformation (Cont.)

The general mapping properties are summarized as following:

- The left-half s-plane is mapped onto the inside of the unit circle of the z-plane.
- The right-half s-plane is mapped onto the outside of the unit circle of the z-plane.
- The positive $j\omega$ axis portion in the s-plane is mapped onto the positive half circle (the dashed-line arrow in Figure) on the unit circle, while the negative $j\omega$ axis is mapped onto the negative half circle (the dotted-line arrow in Figure) on the unit circle.





We substitute $s = j\omega_a$ and $z = e^{j\omega_d T}$ into the BLT in Equation

$$j\omega_a = \frac{2}{T}\frac{e^{j\omega_d T} - 1}{e^{j\omega_d T} + 1} \longrightarrow \omega_a = \frac{2}{T}\tan\left(\frac{\omega_d T}{2}\right)$$

This equation explores the relation between the analog frequency on the $j\omega$ axis and the corresponding digital frequency ω_d on the unit circle.

Example 8.4.

Given an analog filter whose transfer function is

$$H(s) = \frac{10}{s+10},$$

a. Convert it to the digital filter transfer function and difference equation, respectively, when a sampling period is given as $T = 0.01$ second.

Solution: Applying the BLT, we have $$H(z) = H(s)\big|_{s=\frac{2z-1}{Tz+1}} = \frac{10}{s+10}\bigg|_{s=\frac{2z-1}{Tz+1}}$$

Substituting $T = 0.01$, it follows that

$$H(z) = \frac{10}{\frac{200(z-1)}{z+1} + 10} = \frac{0.05}{\frac{z-1}{z+1} + 0.05} = \frac{0.05(z+1)}{z-1+0.05(z+1)} = \frac{0.05z + 0.05}{1.05z - 0.95}.$$

Finally, we get

$$H(z) = \frac{(0.05z + 0.05)/(1.05z)}{(1.05z - 0.95)/(1.05z)} = \frac{0.0476 + 0.0476z^{-1}}{1 - 0.9048z^{-1}}.$$

We can find the difference equation from H(z): $$y(n) = 0.0476x(n) + 0.0476x(n-1) + 0.9048y(n-1)$$

Example 8.6.

The normalized lowpass filter with a cutoff frequency of 1 rad/sec is given as:

$$H_P(s) = \frac{1}{s+1}.$$

a. Use the given $H_p(s)$ and the BLT to design a corresponding digital IIR lowpass filter with a cutoff frequency of 15 Hz and a sampling rate of 90 Hz.

Solution:

a. First, we obtain the digital frequency as $\omega_d = 2\pi f = 2\pi(15) = 30\pi$ rad/sec, and $T = 1/f_s = 1/90$ sec.

We then follow the design procedure:

1. First calculate the prewarped analog frequency as $\omega_a = \frac{2}{T}\tan\left(\frac{\omega_d T}{2}\right) = \frac{2}{1/90}\tan\left(\frac{30\pi/90}{2}\right)$

that is, $\omega_a = 180 \times \tan(\pi/6) = 180 \times \tan(30°) = 103.92$ rad/sec.

2. Then perform the prototype transformation (lowpass to lowpass) as follows:

$$H(s) = H_P(s)_{s=\frac{s}{\omega_a}} = \frac{1}{\frac{s}{\omega_a}+1} = \frac{\omega_a}{s+\omega_a} = \frac{103.92}{s+103.92}.$$

3. Apply the BLT, which yields

$$H(z) = \left.\frac{103.92}{s + 103.92}\right|_{s=\frac{2}{T}\frac{z-1}{z+1}} = \frac{103.92}{180 \times \frac{z-1}{z+1} + 103.92} = \frac{103.92/180}{\frac{z-1}{z+1} + 103.92/180} = \frac{0.5773}{\frac{z-1}{z+1} + 0.5773}.$$

$$H(z) = \frac{0.5773(z+1)}{\left(\frac{z-1}{z+1} + 0.5773\right)(z+1)} = \frac{0.5773z + 0.5773}{(z-1) + 0.5773(z+1)} = \frac{0.5773z + 0.5773}{1.5773z - 0.4227} = \frac{0.3660 + 0.3660z^{-1}}{1 - 0.2679z^{-1}}$$

Some simple and useful IIR Filters: Leaky integrator, Resonator, DC removal, Hum Removal Simple IIR lowpass filter: Leaky integrator

- Leaky integrator is a very simple and computationally efficient filter which is used to remove high frequency components
(e.g. noise).
- It is useful in audio, communication and control systems.

Construction of Leaky integrator

Leaky integrator is developed by modifying the moving average filter. The concept of moving average filter is very simple,
replacing each sample by local average. For instance a 2-point moving average filter do the following operation: $y[n] = (x[n] + x[n-1])/2)$

General mathematical expression of M-point moving average filter is: $y_M[n] = \frac{1}{M}(x[n] + x[n-1] + x[n-2] + \ldots + x[n-M+1])$

Moving average filter over M point:

$$y_M[n] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-k]$$

- Smoothing effect is proportional to M.
- Number of mathematical operations and required storage is also proportional to M.

Construction of Leaky integrator (Cont.)

Recall the M-point MA filter: $$y_M[n] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-k]$$

Delaying the output of MA filter by 1 position, we get

$$y_M[n-1] = \frac{1}{M}\sum_{k=0}^{M-1} x[(n-1)-k] = \frac{1}{M}\sum_{k=0}^{M-1} x[n-(k+1)] = \frac{1}{M}\sum_{k=0+1}^{M-1+1} x[n-k] = \frac{1}{M}\sum_{k=1}^{M} x[n-k]$$

Modifying the equation of above MA filter over M-1 point

$$y_{M-1}[n-1] = \frac{1}{M-1}\sum_{k=1}^{M-1} x[n-k]$$

Moving average filter over M-1 points, delayed by one.

Summation of M-discrete samples can be expressed as

$$\sum_{k=0}^{M-1} x[n-k] = x[n] + \sum_{k=1}^{M-1} x[n-k] \quad \Rightarrow \quad My_M[n] = x[n] + (M-1)y_{M-1}[n-1]$$

Construction of Leaky integrator (Cont.)

$$My_M[n] = x[n] + (M-1)y_{M-1}[n-1] \quad \Rightarrow \quad y_M[n] = \frac{M-1}{M}y_{M-1}[n-1] + \frac{1}{M}x[n]$$

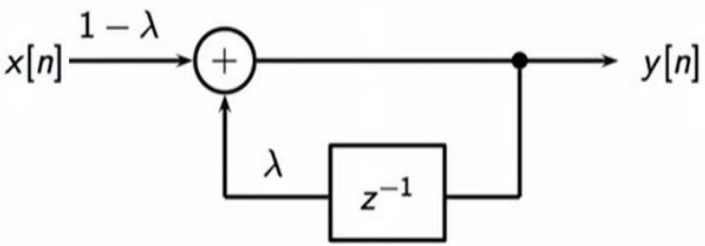
$$y_M[n] = \lambda y_{M-1}[n-1] + (1-\lambda)x[n], \qquad \lambda = \frac{M-1}{M}$$

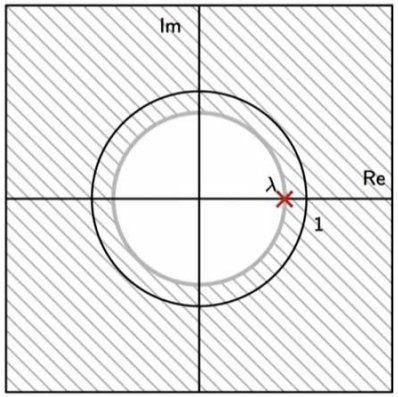
When M is large, $y_{M-1}[n] \approx y_M[n]$ $(and\ \lambda \approx 1)$. The above equation can be written as

$$y[n] = \lambda y[n-1] + (1-\lambda)x[n]$$

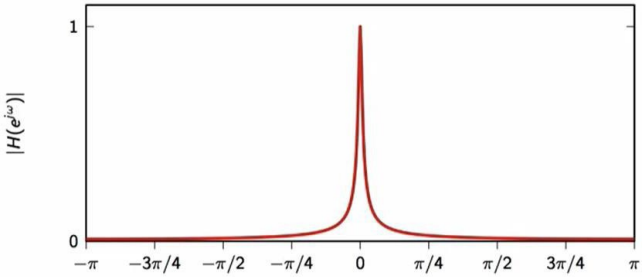
This is the expression of leaky integrator.

The transfer function of leaky integrator. $H(z) = \dfrac{(1-\lambda)}{1-\lambda z^{-1}}$





Frequency response of leaky integrator for $\lambda = 0.98$

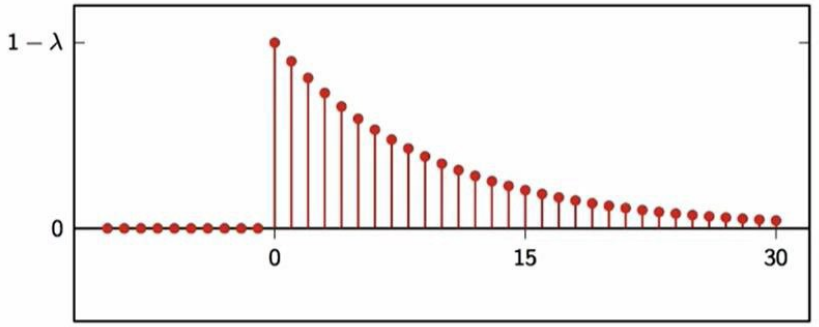


Impulse Response of Leaky Integrator

$$y[n] = \lambda y[n-1] + (1-\lambda)\delta[n]$$

- $y[n] = 0$ for all $n < 0$
- $y[0] = \lambda y[-1] + (1-\lambda)\delta[0] = (1-\lambda)$
- $y[1] = \lambda y[0] + (1-\lambda)\delta[1] = \lambda(1-\lambda)$
- $y[2] = \lambda y[1] + (1-\lambda)\delta[2] = \lambda^2(1-\lambda)$
- $y[3] = \lambda y[2] + (1-\lambda)\delta[3] = \lambda^3(1-\lambda)$
- ...

$$h[n] = (1-\lambda)\lambda^n u[n]$$



Leaky Integrator: Why the name

Discrete-time integrator is a boundless accumulator:

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

We can rewrite the integrator as

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

To prevent "explosion" pick $\lambda < 1$

$$y[n] = \lambda y[n-1] + (1-\lambda)x[n]$$

keep only a fraction λ of the accumulated value so far and forget ("leak") a fraction $1-\lambda$

add only a fraction $1-\lambda$ of the current value to the accumulator

Resonator

- A resonator is a narrow bandpass filter.
- User to detect the presence of a sinusoid of a given frequency.
- Useful in communication systems and telephony (DTMF)

Idea!
Shift the passband of the Leaky Integrator

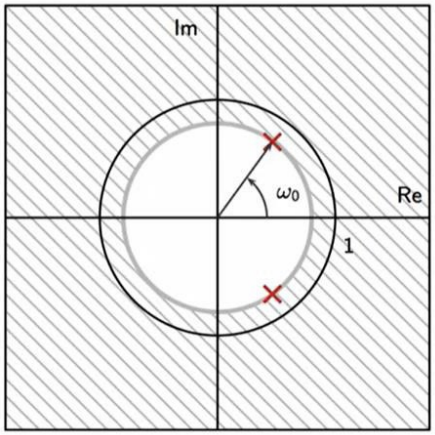
$$H(z) = \frac{G_0}{(1 - pz^{-1})(1 - p^*z^{-1})} \qquad p = \lambda e^{j\omega_0}$$

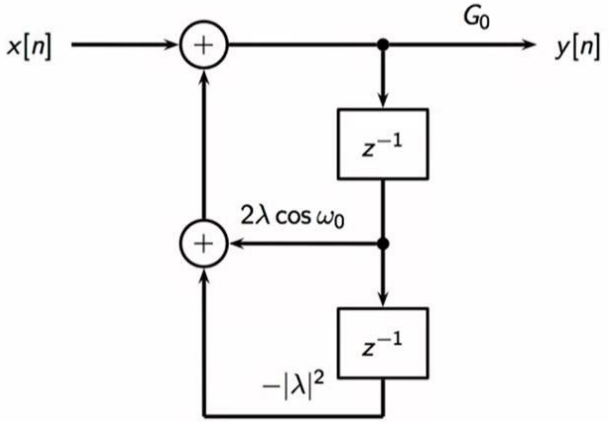
$$= \frac{G_0}{1 - 2\Re\{p\}\, z^{-1} + |p|^2 z^{-2}}$$

$$= \frac{G_0}{1 - 2\lambda \cos\omega_0\, z^{-1} + |\lambda|^2 z^{-2}}$$

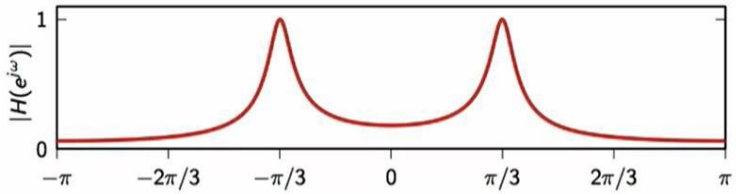
$$a_1 = 2\lambda \cos\omega_0 \qquad a_2 = -|\lambda|^2$$

$$y[n] = G_0 x[n] - a_1 y[n-1] - a_2 y[n-2]$$

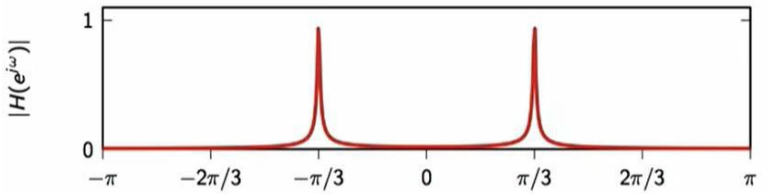




Frequency response of resonator for: $\lambda = 0.9, \omega_0 = \pi/3$



$\lambda = 0.99, \omega_0 = \pi/3$

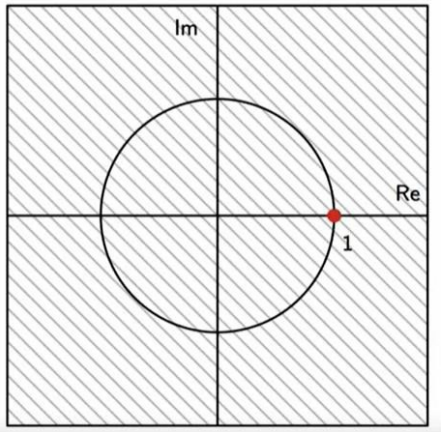


DC removal

- DC offset of a signal carries no information but is troublesome in signal analysis and responsible to waster energy in circuit.
- A DC-balanced signal has zero sum:

i.e. there is no Direct current component. $\lim_{n\to\infty} \sum_{n=-N}^{N} x[n] = 0$

- Its DTFT value at zero is zero.
- we want to remove the DC bias from a non zero-centered signal.
- We want kill the frequency component at $\omega = 0$.

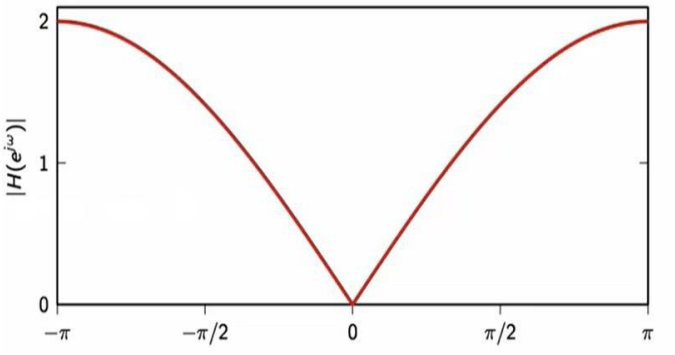


The way to kill the frequency at $\omega = 0$, is simply to place a zero at z=1 in the argand diagram.The transfer function will be:

$$H(z) = 1 - z^{-1}$$

and $y[n] = x[n] - x[n-1]$ The system is non-recursive.

From the frequency response, it is clear that the system remove DC offset. However, it also introduce a very big attenuation over all most the entire frequency range.

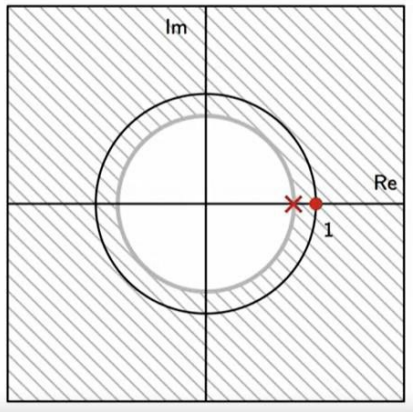


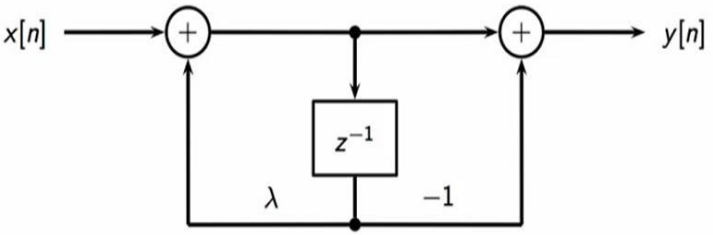
DC removal, improved (DC notch)

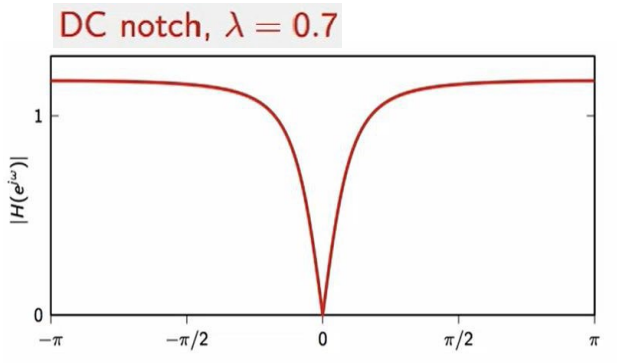
We can push up z-transform near zero by putting a pole in the vicinity of the zero. Therefore, we paced a pole close to 1 and inside the unit circle. Thus, we get the following transfer function which is the combination of notch in zero and a leaky integrator.

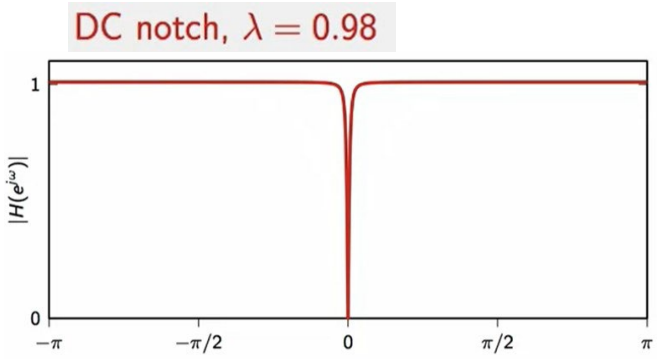
$$H(z) = \frac{1 - z^{-1}}{1 - \lambda z^{-1}}$$

$$y[n] = \lambda y[n-1] + x[n] - x[n-1]$$









Hum Removal (Notch Filter)

- Similar to DC removal but we want to remove a specific non zero frequency.

  Very useful for musicians: amplifiers for electric guitars pick up the hum from the electric mains (50 Hz in Europe and 60 Hz in North America)

- We need to tune the hum removal according to country.

Idea!

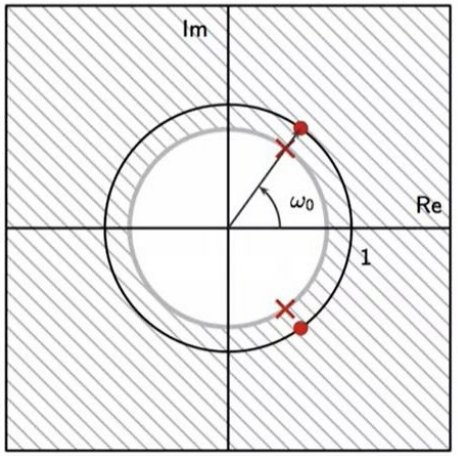
Shift the passband of the DC Notch

$$H(z) = \frac{(1 - e^{j\omega_0} z^{-1})(1 - e^{-j\omega_0} z^{-1})}{(1 - \lambda e^{j\omega_0} z^{-1})((1 - \lambda e^{-j\omega_0} z^{-1}))}$$

$$= \frac{1 - e^{-j\omega_0} z^{-1} - e^{j\omega_0} z^{-1} + z^{-2}}{1 - \lambda e^{-j\omega_0} z^{-1} - \lambda e^{j\omega_0} z^{-1} + \lambda^2 z^{-2}}$$

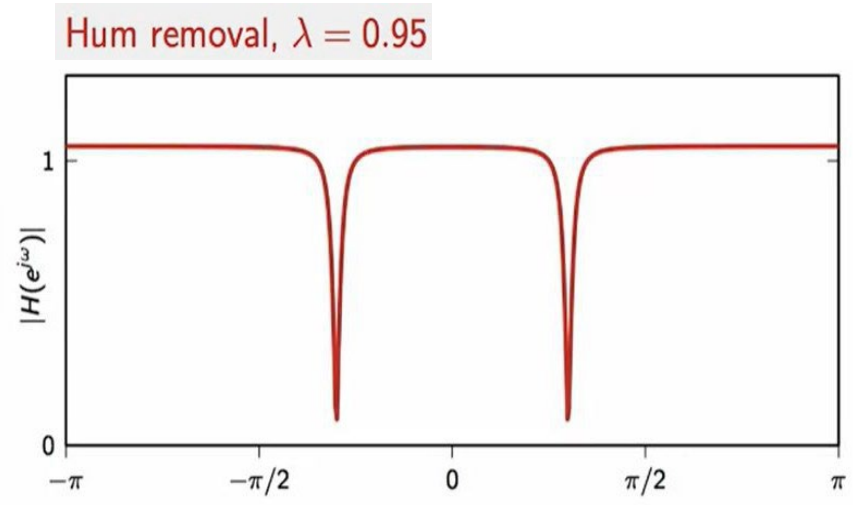
$$= \frac{1 - 2z^{-1}(\frac{e^{j\omega_0} + e^{-j\omega_0}}{2}) + z^{-2}}{1 - 2\lambda z^{-1}(\frac{e^{j\omega_0} + e^{-j\omega_0}}{2}) + \lambda^2 z^{-2}}$$

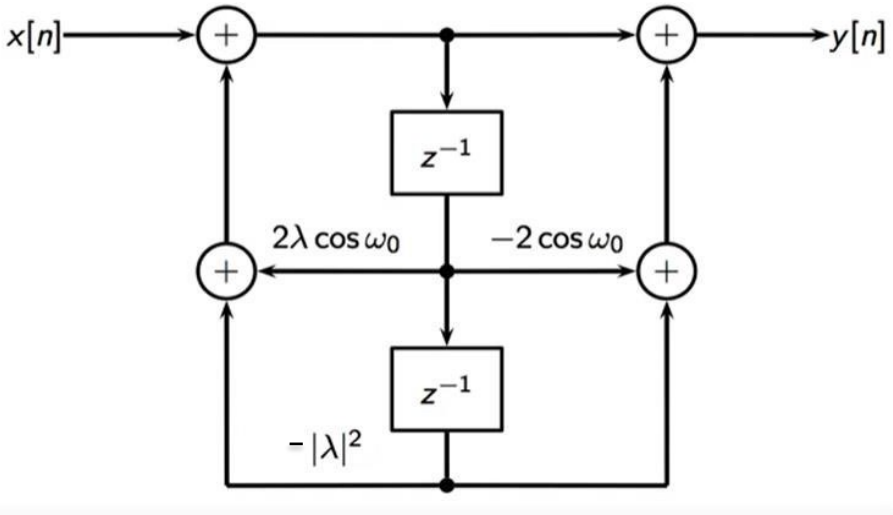
$$= \frac{1 - 2\cos\omega_0 z^{-1} + z^{-2}}{1 - 2\lambda\cos\omega_0 z^{-1} + \lambda^2 z^{-2}}$$



Hum Removal (Cont.)

$$H(z) = \frac{1 - 2\cos\omega_0\, z^{-1} + z^{-2}}{1 - 2\lambda\cos\omega_0\, z^{-1} + \lambda^2 z^{-2}}$$





Class Test Next Week

Syllabus: Slide 253-300

Assignment: Given in Google Drive Folder

Submission Deadline : Before Final Exam