



CAPITAL UNIVERSITY - KODERMA

DIGITAL SIGNAL PROCESSING - ASSIGNMENT

Name : Arshad Nazir

Electrical and Electronics Engineering

Signature:

Date :

PART 1

1. Define Signal.

Anything that carries information can be called as signal. It can also be defined as a physical quantity that varies with time, temperature, pressure or with any independent variables such as speech signal or video signal.

2. Define system

A system is a group of interacting or interrelated elements that act according to a set of rules to form a unified whole. A system, surrounded and influenced by its environment, is described by its boundaries, structure and purpose and expressed in its functioning.

3. What are the steps involved in digital signal processing?

- Sampling. The audio data we wish to treat will generally be present in the form of electric oscillations.
- Power and Energy.
- Fourier Transform.
- Convolution, Filtering and Linear Systems.
- Windowing.

4. Give some applications of DSP?

DSP is used primarily in areas of the audio signal, speech processing, RADAR, seismology, audio, SONAR, voice recognition, and some financial signals. For example, Digital Signal Processing is used for speech compression for mobile phones, as well as speech transmission for mobile phones.

5. Write the classifications of DT Signals.

Discrete time signals can be classified as follows:

- Even and odd signals.
- Periodic and non-periodic signals.
- Deterministic and random signals.
- Energy signals and power signals.
- Multichannel and multidimensional signals.

6. What is an Energy and Power signal?

Energy signal is a signal whose energy is finite and power is zero whereas Power signal is a signal whose power is finite and energy is infinite

7. What is Discrete Time Systems?

A discrete-time system is anything that takes a discrete-time signal as input and generates a discrete-time signal as output. The concept of a system is very general. It may be used to model the response of an audio equalizer or the performance of the US economy.

8. Write the Various classifications of Discrete-Time systems.

The discrete time systems can be classified as follows:

- Static/Dynamic
- Causal/Non-Causal
- Time invariant/Time variant
- Linear/Non-Linear
- Stable/Unstable

9. Define linear system

In [systems theory](#), a linear system is a [mathematical model](#) of a [system](#) based on the use of a [linear operator](#). Linear systems typically exhibit features and properties that are much simpler than the [nonlinear](#) case. As a mathematical abstraction or idealization, linear systems find important applications in [automatic control](#) theory, [signal processing](#), and [telecommunications](#). For example, the propagation medium for wireless communication systems can often be modeled by linear systems.

10. Define Static & Dynamic systems

STATIC SYSTEMS lack movement. Fixed by design; often viewed as out-of-fashion, monotonous or uninteresting. Examples of these types of systems, for our purpose, include manual business forms, paperback books, manual time sheets, cork bulletin boards, etc. DYNAMIC SYSTEMS change states.

A dynamic system is a system or process in which motion occurs, or includes active forces, as opposed to static conditions with no motion. Dynamic systems by their very nature are constantly moving or must change states to be useful. These types of systems include: Vehicles.

11. Define causal and non-causal signals.

A causal system is one whose output depends only on the present and the past inputs. A noncausal system's output depends on the future inputs. In a sense, a noncausal system is just the opposite of one that has memory. ... It cannot because real systems cannot react to the future.

12. Define odd and even signal

Even signals are symmetric around vertical axis, and Odd signals are symmetric about origin. Even Signal: A signal is referred to as an even if it is identical to its time-reversed counterparts; $x(t) = x(-t)$. Odd Signal: A signal is odd if $x(t) = -x(-t)$.

13. Define deterministic and random signal

A signal is said to be deterministic if there is no uncertainty with respect to its value at any instant of time. ... Non-deterministic signals are random in nature hence they are called random signals. Random signals cannot be described by a mathematical equation.

14. Define periodic and Aperiodic signals.

A signal is said to be periodic signal if it has a definite pattern and repeats itself at a regular interval of time. Whereas, the signal which does not at the regular interval of time is known as an aperiodic signal or non-periodic signal.

15. Define causal and non-causal systems

A causal system is one whose output depends only on the present and the past inputs. A noncausal system's output depends on the future inputs. In a sense, a noncausal system is just the opposite of one that has memory. ... It cannot because real systems cannot react to the future.

16. Define stable and unstable system

The system is said to be stable only when the output is bounded for bounded input. For a bounded input, if the output is unbounded in the system then it is said to be unstable. Note: For a bounded signal, amplitude is finite. ... Hence, the system is unstable.

17. Define time variant and time invariant system.

A system is said to be time invariant if the response of the system to an input is not a function of time. On the other hand a system is time variant if the response to an input alters with time i.e. the system has varying response to the same input at different instants of time.

18. Define recursive and non recursive system.

This is a recursive system which means the output at time n depends on any number of a past output values. So, a recursive system has feed back output of the system into the input. This feed back loop contains a delay element. If $y(n)$ depends only on the present and past input, it is called non recursive.

19. Define IIR & FIR systems

In signal processing, a finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. ... FIR filters can be discrete-time or continuous-time, and digital or analogue.

The infinite impulse response (IIR) filter is a recursive filter in that the output from the filter is computed by using the current and previous inputs and previous outputs. Because the filter uses previous values of the output, there is feedback of the output in the filter structure.

20. Why linear convolution is important in DSP?

Linear convolution gives the output we get after passing the input through a system (eg. filter). So, if the impulse response of a system is known, then the response for any input can be determined using convolution operation.

21. Define Z transform.

In mathematics and signal processing, the Z-transform converts a discrete-time signal, which is a sequence of real or complex numbers, into a complex frequency-domain representation. It can be considered as a discrete-time analogue of the Laplace transform.

22. What are the basic elements used to construct the block diagram of discrete time system?

Using the fundamental building blocks, the block diagrams of discrete time systems can be prepared. The fundamental building blocks are adders, multipliers, delay and advance elements which are discussed in this article.

23. What is ROC in Z-Transform?

The region of convergence (ROC) is the set of points in the complex plane for which the Z-transform summation converges.

Region of Convergence (ROC) of Z-Transform

The variation of z for which z-transform converges is known as region of convergence of z-transform.

24. List any four properties of Z-Transform.

- Linearity.
- Symmetry.
- Time Scaling.
- Time Shifting.
- Convolution.
- Time Differentiation.
- Parseval's Relation.
- Modulation (Frequency Shift)

25. What are the different methods of evaluating inverse z-transform?

1. Inspection
2. Partial-Fraction Expansion
3. Power Series Expansion
4. Contour Integration

26. What are the properties of convolution?

Linear convolution has three important properties:

1. Commutative property
2. Associative property
3. Distributive property

27. Define DTFT.

In mathematics, the discrete-time Fourier transform (DTFT) is a form of Fourier analysis that is applicable to a sequence of values. The DTFT is often used to analyze samples of a continuous function.

28. State the condition for existence of DTFT?

Sufficient Condition for Existence of the DTFT

A sequence $x[n]$ satisfying is said to be absolutely summable, and when holds, the infinite sum defining the DTFT $X(e^{j\omega})$ is said to converge to a finite result for all ω .

29. List the properties of DTFT.

- Linearity.
- Symmetry.
- Time Scaling.
- Time Shifting.
- Convolution.
- Time Differentiation.
- Parseval's Relation.
- Modulation (Frequency Shift)

30. What is the DTFT of unit sample?

A single unit sample has a DTFT that is 1. Addition of a pair of unit samples at ± 1 adds a cosine wave of frequency 1 to the DTFT. Addition of a pair of unit samples at ± 2 adds a cosine of frequency 2 to the DTFT. As more unit samples are added, $x[n] \rightarrow 1$ and the DTFT approaches a periodic impulse train of frequency 1.

31. What is DFT?

In mathematics, the discrete Fourier transform (DFT) converts a finite sequence of equally-spaced samples of a function into a same-length sequence of equally-spaced samples of the discrete-time Fourier transform (DTFT), which is a complex-valued function of frequency.

32. Define N point DFT.

Definition. An N-point DFT is expressed as the multiplication , where $x[n]$ is the original input signal, W_N is the N-by-N square DFT matrix, and $X[k]$ is the DFT of the signal.

33. List the properties of DFT.

- Translation.
- Distributive and scaling.
- Rotation.
- Periodicity and Conjugate Symmetry.
- Separability (kernel separating)
- Linearity.
- Convolution and Correlation.

34. State Linearity property of DFT.

The linearity property states that if $x_1[n]$ and $x_2[n]$ are two signals, then the DFT of their linear combination is equal to the same linear combination of their DFTs.

35. What is the disadvantage of direct computation of DFT?

For the computation of N-point DFT, N^2 complex multiplications and $N(N-1)$ complex additions are required. If the value of N is large (e.g., into lakhs), this proves the inefficiency of direct DFT computation.

36. Define IIR filter?

The infinite impulse response (IIR) filter is a recursive filter in that the output from the filter is computed by using the current and previous inputs and previous outputs. Because the filter uses previous values of the output, there is feedback of the output in the filter structure.

37. What is prewarping?

Prewarping. Frequency warping follows a known pattern, and there is a known relationship between the warped frequency and the known frequency. We can use a technique called prewarping to account for the nonlinearity, and produce a more faithful mapping.

38. State the frequency relationship in bilinear transformation?

Frequency warping transformation is a process where one spectral representation on a certain frequency scale (e.g., Hz, f-domain) and with a certain frequency resolution (most often uniform) is transformed to another representation on a new frequency scale (e.g., Bark or ERB-rate scale, v-domain).

39. What is the frequency response of Butterworth filter?

The frequency response of the Butterworth filter is maximally flat (i.e. has no ripples) in the passband and rolls off towards zero in the stopband. When viewed on a logarithmic Bode plot, the response slopes off linearly towards negative infinity.

40. What is FIR filters?

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

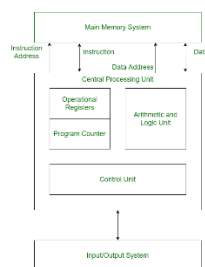
PART 1A

1. Draw the block diagram of Harvard architecture and explain.

In a normal computer that follows von Neumann architecture, instructions and data both are stored in the same memory. So same buses are used to fetch instructions and data. This means the CPU cannot do both things together (read the instruction and read/write data).

Harvard Architecture is the computer architecture that contains separate storage and separate buses (signal path) for instruction and data. It was basically developed to overcome the bottleneck of Von Neumann Architecture. The main advantage of having separate buses for instruction and data is that the CPU can access instructions and read/write data at the same time.

Structure of Harvard Architecture:



Structure of Harvard Architecture

Buses

Buses are used as signal pathways. In Harvard architecture, there are separate buses for both instruction and data. Types of Buses:

- **Data Bus:** It carries data among the main memory system, processor, and I/O devices.
- **Data Address Bus:** It carries the address of data from the processor to the main memory system.
- **Instruction Bus:** It carries instructions among the main memory system, processor, and I/O devices.
- **Instruction Address Bus:** It carries the address of instructions from the processor to the main memory system.

Operational Registers

There are different types of registers involved in it which are used for storing addresses of different types of instructions. For example, the Memory Address Register and Memory Data Register are operational registers.

Program Counter

It has the location of the next instruction to be executed. The program counter then passes this next address to the memory address register.

Arithmetic and Logic Unit

The arithmetic logic unit is that part of the CPU that operates all the calculations needed. It performs addition, subtraction, comparison, logical Operations, bit Shifting Operations, and various arithmetic operations.

Control Unit

The Control Unit is the part of the CPU that operates all processor control signals. It controls the input and output devices and also controls the movement of instructions and data within the system.

Input/Output System

Input devices are used to read data into main memory with the help of CPU input instruction. The information from a computer as output is given through Output devices. The computer gives the results of computation with the help of output devices.

Advantage of Harvard Architecture:

Harvard architecture has two separate buses for instruction and data. Hence, the CPU can access instructions and read/write data at the same time. This is the major advantage of Harvard architecture.

In practice, Modified Harvard Architecture is used where we have two separate caches (data and instruction). This is common and used in X86 and ARM processors.

2. Write short notes on • Memory mapped register addressing • Circular addressing mode • Auxiliary registers

Most registers are memory-mapped—that is, the register has an address in the memory space. A memory-mapped register can be referred to in assembly language in two different ways: either by referring to its mnemonic name or through its address.

Circular buffer allows one to handle a continuous stream of incoming data samples; once the end of the buffer is reached, samples are added to the beginning again. ii. Bit-Reversed Addressing Mode: Address generation unit can be provided with the capability of providing bit-reversed indices.

The purpose of the Auxiliary Control Register is to control processor-specific features that are not architecturally described. The Auxiliary Control Register is: partially banked. accessible in privileged modes only.

3.Explain the advantages and disadvantages of VLIW architecture.

Advantage

- the number of FUs can be increased without needing additional sophisticated hardware to detect parallelism, like in superscalars.
- VLIW is less complex than the Superscalar because VLIW.
- Because VLIW is implemented at the software level, all available storage space can be used. But for superscalar, its implemented in the hardware and some space will have to be allocated for the hardware so not all available storage space can be used.

Disadvantage

- Superscalar machines are able to dynamically issue multiple instructions each clock cycle from a conventional linear instruction stream while VLIW is static.

4. Explain various addressing modes of a digital signal processor.

Addressing modes are an aspect of the [instruction set architecture](#) in most [central processing unit](#) (CPU) designs. The various addressing modes that are defined in a given instruction set architecture define how the [machine language instructions](#) in that architecture identify the [operand\(s\)](#) of each instruction. An addressing mode specifies how to calculate the effective [memory address](#) of an operand by using information held in [registers](#) and/or constants contained within a machine instruction or elsewhere.

In [computer programming](#), addressing modes are primarily of interest to those who write in [assembly languages](#) and to [compiler](#) writers. For a related concept see [orthogonal instruction set](#) which deals with the ability of any instruction to use any addressing mode.

Note that there is no generally accepted way of naming the various addressing modes. In particular, different authors and computer manufacturers may give different names to the same addressing mode, or the same names to different addressing modes.

Furthermore, an addressing mode which, in one given architecture, is treated as a single addressing mode may represent functionality that, in another architecture, is covered by two or more addressing modes.

For example, some [complex instruction set computer](#) (CISC) architectures, such as the [Digital Equipment Corporation \(DEC\) VAX](#), treat registers and [literal or immediate constants](#) as just another addressing mode. Others, such as the [IBM System/360](#) and its successors, and most [reduced instruction set computer](#) (RISC) designs, encode this information within the instruction. Thus, the latter machines have three distinct instruction codes for copying one register to another, copying a literal constant into a register, and copying the contents of a memory location into a register, while the VAX has only a single "MOV" instruction.

5.What is MAC unit? Explain its functions.

The MAC unit is a unit that is mostly demanded in DSP applications. MAC unit performs both multiply and addition functions. It operates in two stages. Firstly, it computes the product of given numbers and forward the result for the second stage operation i.e., addition/accumulate. If both the computing is executed in a single rounding then it is said to be fused multiply-add/accumulate (MAC) unit. There has been a lot of research performed on MAC implementation. This paper provides a comparative study and analysis of the research and investigations held till now.

In many of the DSP applications the operations performed generally involve many multiplications and accumulations. In real-time signal processing, a high speed and high throughput MAC is always a key to achieve high performance in digital signal processing processors.

The main consideration of MAC design is to enhance its speed.

Because speed and throughput rate are always the main objective of digital signal processing system. But in case personal communication, low power design also becomes another main theme because, energy in the battery available for these products hinder the power consumption of the personal communication system.

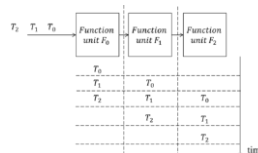
A conventional MAC unit consists of multiplier, adder and an accumulator unit that contains the sum of the previous consecutive products.
The function of the MAC unit: iB_i

6. Explain about pipelining in DSP.

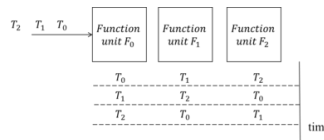
Pipelining is an important technique used in several applications such as [digital signal processing](#) (DSP) systems, [microprocessors](#), etc. It originates from the idea of a water pipe with continuous water sent in without waiting for the water in the pipe to come out. Accordingly, it results in speed enhancement for the critical path in most DSP systems. For example, it can either increase the [clock speed](#) or reduce the power consumption at the same speed in a DSP system.

Pipelining allows different functional units of a system to run concurrently. Consider an informal example in the following figure. A system includes three sub-function units (F_0 , F_1 and F_2). Assume that there are three independent tasks (T_0 , T_1 and T_2) being performed by these three function units. The time for each function unit to complete a task is the same and will occupy a slot in the schedule.

If we put these three units and tasks in a sequential order, the required time to complete them is five slots.



However, if we pipeline T_0 to T_2 concurrently, the aggregate time is reduced to three slots.



Therefore, it is possible for an adequate pipelined design to achieve significant enhancement on speed.

7. Explain impulse invariant method of designing IIR filter.

Impulse invariance is a technique for designing discrete-time [infinite-impulse-response](#) (IIR) filters from continuous-time filters in which the impulse response of the continuous-time system is sampled to produce the impulse response of the discrete-time system. The frequency response of the discrete-time system will be a sum of shifted copies of the frequency response of the continuous-time system; if the continuous-time system is approximately band-limited to a frequency less than the [Nyquist frequency](#) of the sampling, then the frequency response of the discrete-time system will be approximately equal to it for frequencies below the Nyquist frequency.

The Impulse Invariance Method is used to design a discrete filter that yields a similar frequency response to that of an analog filter. Discrete filters are amazing for two very significant reasons:

1. You can separate signals that have been fused and,
2. You can use them to retrieve signals that have been distorted.

We can design this filter by finding out one very important piece of information i.e., the impulse response of the analog filter. By sampling the response, we will get the time-domain impulse response of the discrete filter.

When observing the impulse responses of the continuous and discrete responses, it is hard to miss that they correspond with each other. The analog filter can be represented by a transfer function, $H_c(s)$.

Zeros are the roots of the numerator and poles are the roots of the denominator. For every pole of the transfer function of the analog filter, it can be mapped to a pole on the transfer function of the [IIR filter](#)'s transfer function given by $H(z)$. Let us delve deeper into how we can go about doing this.

8. Explain the following properties of DFT. • Convolution. • Time shifting • Conjugate Symmetry.

In mathematics (in particular, functional analysis), convolution is a mathematical operation on two functions (f and g) that produces a third function (\cdot) that expresses how the shape of one is modified by the other. The term convolution refers to both the result function and to the process of computing it.

Time shifting is, as the name suggests, the shifting of a signal in time. This is done by adding or subtracting a quantity of the shift to the time variable in the function.

In communications, the term time shifting refers to the transmission of messages or data to be read, heard, or viewed by the recipient at a later time. E-mail, voice mail, and fax are common examples.

9. Explain the classification of discrete signal.

Discrete time signals can be classified as follows:

- Even and odd signals
- Periodic and non-periodic signals
- Deterministic and random signals
- Energy signals and power signals
- Multichannel and multidimensional signals

10. Explain the classification of discrete systems

The discrete time systems can be classified as follows:

- Static/Dynamic
- Causal/Non-Causal
- Time invariant/Time variant
- Linear/Non-Linear
- Stable/Unstable

Static and Dynamic Systems

The system is said to be static if its output depends only on the present input. On the other hand, if the output of the system depends on the past input, the system is said to be dynamic.

For example, $y(n) = 5x(n)$ and $y(n) = 2x^2(n) + 3x(n)$ are the static systems whereas $y(n) = x(n) + 2x(n-1)$ represents the dynamic system.

Causal and Non-Causal Systems

If the output of the system depends on the past and presents input only, the system is said to be a causal system. On the other hand, if the output of the system ...