UNIT- 1

Signal:

A signal is any physical quantity that carries information, and that varies with time, space, or any other independent variable or variables. Mathematically, a signal is defined as a function of one or more independent variables.

1 – Dimensional signals mostly have time as the independent variable. For example,

Eg., $S_1(t) = 20t^2$

2 – Dimensional signals have two independent variables. For example, image is a 2 – D signal whose independent variables are the two spatial coordinates (x,y)

Eg., $S_2(t) = 3x + 2xy + 10y^2$

Video is a 3 – dimensional signal whose independent variables are the two spatial coordinates, (x,y) and time (t).
Similarly, a 3 – D picture is also a 3 – D signal whose independent variables are the three spatial coordinates (x,y,z).

Signals $S_1(t)$ and $S_2(t)$ belong to a class that are precisely defined by specifying the functional dependence on the independent variables.
Natural signals like speech signal, ECG, EEG, images, videos, etc. belong to the class which cannot be described functionally by mathematical expressions.

System

A system is a physical device that performs an operation on a signal. For example, natural signals are generated by a system that responds to a stimulus or force.
For eg., speech signals are generated by forcing air through the vocal cords. Here, the vocal cord and the vocal tract constitute the system (also called the vocal cavity). The air is the stimulus.
The stimulus along with the system is called a signal source.

An electronic filter is also a system. Here, the system performs an operation on the signal, which has the effect of reducing the noise and interference from the desired information – bearing signal.

When the signal is passed through a system, the signal is said to have been processed.

Processing

The operation performed on the signal by the system is called **Signal Processing.** The system is characterized by the type of operation that it performs on the signal. For example, if the operation is linear, the system is called linear system, and so on.

Digital Signal Processing

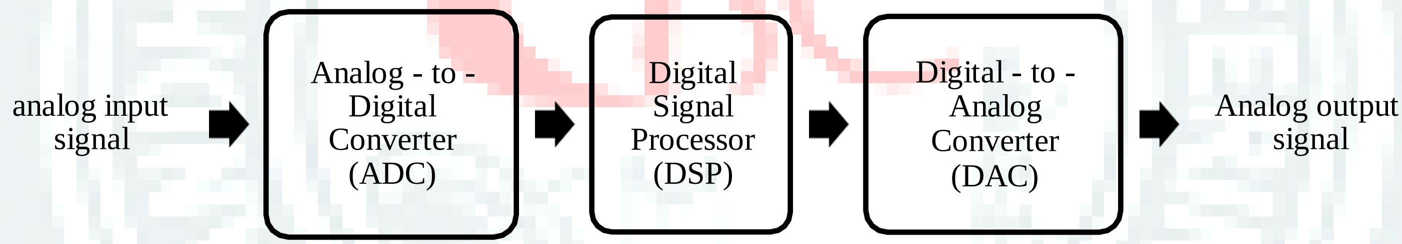
Digital Signal Processing of signals may consist of a number of mathematical operations as specified by a software program, in which case, the program represents an implementation of the system in software. Alternatively, digital processing of signals may also be performed by digital hardware (logic circuits). So, a digital system can be implemented as a combination of digital hardware and software, each of which performs its own set of specified operations.

Basic elements of a Digital Signal Processing System

Most of the signals encountered in real world are analog in nature .i.e., the signal value and the independent variable take on values in a continuous range. Such signals may be processed directly by appropriate analog systems, in which case, the processing is called **analog signal processing**. Here, both the input and output signals are in analog form.

These analog signals can also be processed digitally, in which case, there is a need for an interface between the analog signal and the **Digital Signal Processor**. This interface is called the **Analog – to – Digital Converter (ADC)**, whose output is a digital signal that is appropriate as an input to the digital processor.
In applications such as speech communications, that require the digital output of the digital signal processor to be given to the user in analog form, another interface from digital domain to analog domain is required. This interface is called the **Digital – to – Analog Converter (DAC)**.
In applications like radar signal processing, the information extracted from the radar signal, such as the position of the aircraft and its speed are required in digital format. So, there is no need for a DAC in this case.

Block Diagram Representation of Digital Signal Processing



Advantages of Digital Signal Processing over Analog Signal Processing

1. A digital programmable system allows flexibility in reconfiguring the digital signal processing operations simply by changing the program.
   Reconfiguration of an analog system usually implies a redesign of the hardware followed by testing and verification.
2. Tolerances in analog circuit components and power supply make it extremely difficult to control the accuracy of analog signal processor.
   A digital signal processor provides better control of accuracy requirements in terms of word length, floating – point versus fixed – point arithmetic, and similar factors.
3. Digital signals are easily stored on magnetic tapes and disks without deterioration or loss of signal fidelity beyond that introduced in A/D conversion. So the signals become transportable and can be processed offline.
4. Digital signal processing is cheaper than its analog counterpart.
5. Digital circuits are amenable for full integration. This is not possible for analog circuits because inductances of respectable value (μH or mH) require large space to generate flux.
6. The same digital signal processor can be used to perform two operations by time multiplexing, since digital signals are defined only at finite number of time instants.

7. Different parts of digital signal processor can work at different sampling rates.
8. It is very difficult to perform precise mathematical operations on signals in analog form but theseoperations can be routinely implemented on a digital computer using software.
9. Several filters need several boards in analog signal processing, whereas in digital signal processing,same DSP processor is used for many filters.

Disadvantages of Digital Signal Processing over Analog Signal Processing

1. Digital signal processors have increased complexity.
2. Signals having extremely wide bandwidths require fast – sampling – rate ADCs. Hence the frequencyrange of operation of DSPs is limited by the speed of ADC.
3. In analog signal processor, passive elements are used, which dissipate very less power. In digital signal processor, active elements like transistors are used, which dissipate more power.

The above are some of the advantages and disadvantages of digital signal processing over analog signalprocessing.

Discrete – time signals

A discrete time signal is a function of an independent variable that is an integer, and is represented by x [n] ,where n represents the sample number (**and not the time at which the sample occurs).**

A discrete time signal is not defined at instants between two successive samples, or in other words, for non –integer values of n. (**But, it is not zero, if n is not an integer).**

Continuous- time signal:

The signals that are defined for every instant of time are known as continuous time signals. They are denoted by x(t).

Digital signal:

The signals that are discrete in time and quantized in amplitude are digital signals.

Continuous valued versus Discrete valued signals:

- The values of a continuous time or discrete time signal can be continuous or discrete. If signal takes on all possible values on a finite or an infinite range , it is said to be a continuous valued signal.
- Alternatively, if signal takes on values from a finite set of possible values , it is said to be a discrete valued signal.

Multichannel and Multidimensional Signals:

- In some application signals are generated by multiple sources or multiple sensors. Such signals, in turn can be represented in vector form.
- Example , Earthquake : The acceleration is the result of three basic types of elastic waves. The primary (P) waves and the secondary (s) waves propagates within the body of rock and are longitudinal and transversal respectively. The third type of elastic waves is called the surface wave, because it propagates near the ground surface. Such a vector of signals is called

Multichannel 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 multidimensional / M- dimensional if its value is a function of M independent variables. E.g Video signal

Concept of Frequency in Continuous time and discrete time signals:

- For design of a radio receiver , a high fidelity system or a spectral filter for color photography.
- From physics we know that the frequency is closely related to a specific type of periodic motion called harmonic oscillation, which is described by sinusoidal functions. The concept of frequency is directly related to concept of time. It has the dimension of inverse time. Thus we should except the nature of time (continuous or discrete) would affect the nature of the frequency accordingly.

Continuous –Time Sinusoidal Signals:

A simple harmonic oscillation is mathematically described by the following continuous time sinusoidal signal;

$$X_a(t) = A\cos(\Omega t + \emptyset) \quad -\infty < t < +\infty$$

The signal is completely characterized by three parameters: " A" is the amplitude of the sinusoid ,

"Ω " is the frequency in radians per second(rad/s), $\emptyset$ is the phase in radians. Instead of "Ω " , we often use the frequency F in cycles per second or hertz(Hz), where

$$\Omega = 2\pi F$$

Discrete time Sinusoidal Signals :

A discrete time sinusoidal signal may be expressed as

$$X(n) = A\cos(wn + \emptyset)$$

"W" is the frequency in radians per sample

If instead of "w" we use the frequency variable " f" defined by

$W = 2\pi f$ where f has dimensions of cycles per sample.

Analog to Digital Conversion:

Three steps involved in conversion of analog signal to digital signal

- Sampling
- Quantization
- Encoding

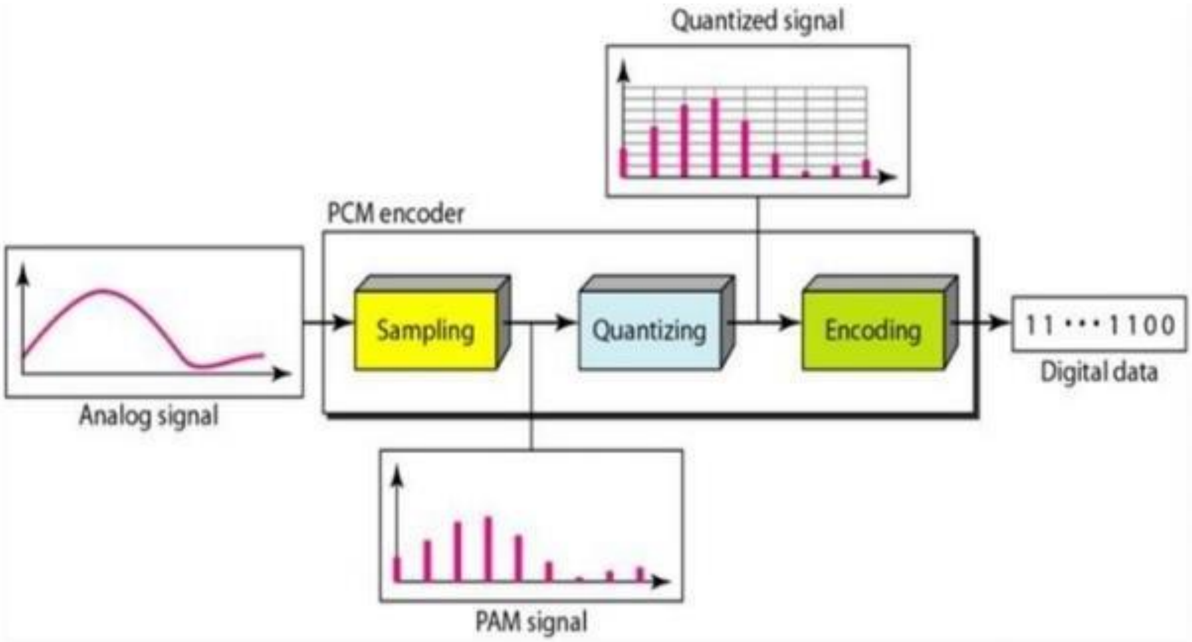


Fig. 2 Conversion of Analog Signal to Digital Signal

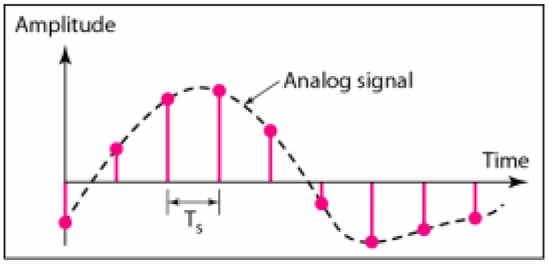
A) Sampling of analog signal:

- Process of converting analog signal into discrete signal.
- Sampling is common in all pulse modulation techniques

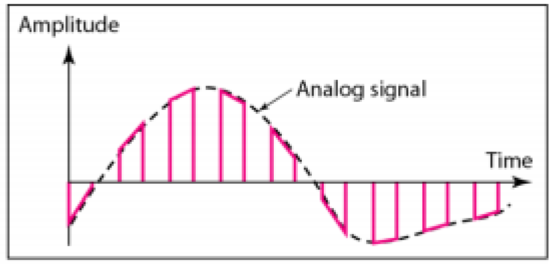
- The signal is sampled at regular intervals such that each sample is proportional to amplitude of signal at that instant
- Analog signal is sampled every T_s $Secs$, called sampling interval. f_s=1/T_S is called sampling rate or sampling frequency.
- f_s=2f_m is Min. sampling rate called **Nyquist rate.** Sampled spectrum (ω) is repeating periodically without overlapping.
- Original spectrum is centered at ω=0 and having bandwidth of ω_m. Spectrum can be recovered by passing through low pass filter with cut-off ω_m.
- For f_s<2f_m sampled spectrum will overlap and cannot be recovered back. This is called **aliasing.**

Sampling methods:

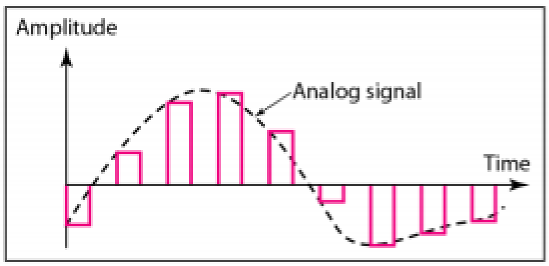
- Ideal – An impulse at each sampling instant.
- Natural – A pulse of Short width with varying amplitude.
- Flat Top – Uses sample and hold, like natural but with single amplitude value.



a. Ideal sampling



b. Natural sampling



c. Flat-top sampling

Statement of sampling theorem

1) *A band limited signal of finite energy, which has no frequency components higher than W Hertz, is completely described by specifying the values of the signal at instants of time separated by* $\frac{1}{2W}$ *seconds and*

2) *A band limited signal of finite energy, which has no frequency components higher than W Hertz, may be completely recovered from the knowledge of its samples taken at the rate of 2W samples per second.*

The first part of above statement tells about sampling of the signal and second part tells about reconstruction of the signal. Above statement can be combined and stated alternately as follows :

A continuous time signal can be completely represented in its samples and recovered back if the sampling frequency is twice of the highest frequency content of the signal. i.e.,

$$f_s \geq 2W$$

Here f_s is the sampling frequency and

W is the higher frequency content

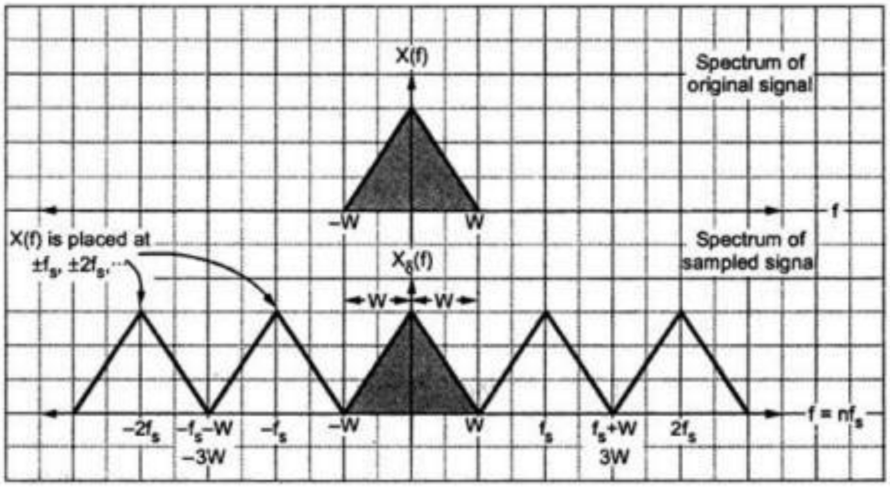


Fig. 6 **Spectrum of original signal and sampled signal**

Quantization:

- **Quantization** is representing the sampled values of the amplitude by a finite set of levels, which means converting a continuous-amplitude sample into a discrete-time signal
- Both sampling and quantization result in the loss of information.
- The quality of a Quantizer output depends upon the number of quantization levels used.
- The discrete amplitudes of the quantized output are called as **representation levels** or **reconstruction levels**.
- The spacing between the two adjacent representation levels is called a **quantum** or **step-size**.
- There are two types of Quantization
  - Uniform Quantization
  - Non-uniform Quantization.
- The type of quantization in which the quantization levels are uniformly spaced is termed as a **Uniform Quantization**.
- The type of quantization in which the quantization levels are unequal and mostly the relation between them is logarithmic, is termed as a **Non-uniform Quantization.**

Uniform Quantization:

- There are two types of uniform quantization.
  - Mid-Rise type
  - Mid-Tread type.
- The following figures represent the two types of uniform quantization.

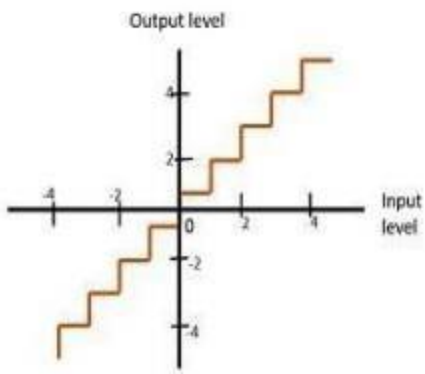


Fig 1 : Mid-Rise type Uniform Quantization

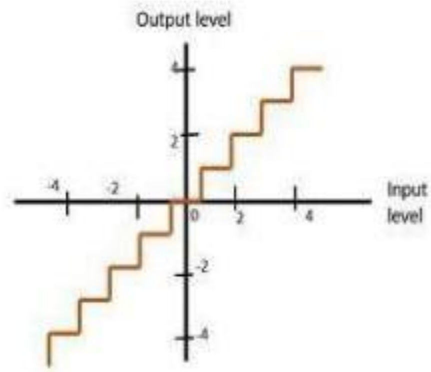


Fig 2 : Mid-Tread type Uniform Quantization

- The **Mid-Rise** type is so called because the origin lies in the middle of a raising part of the stair-case like graph. The quantization levels in this type are even in number.
- The **Mid-tread** type is so called because the origin lies in the middle of a tread of the stair-case like graph. The quantization levels in this type are odd in number.
- Both the mid-rise and mid-tread type of uniform quantizer is symmetric about the origin.

Coding of Quantized Samples(Encoding):

- The coding process in an A/D converter assigns a unique binary number to each quantization level. If we have L levels we need at least L different binary numbers .
- With a word length of "b" bits we can create 2^b different binary numbers
- Hence we have $2^b >= L$ or equivalently $b >= \log_2 L$.
- For example if we have 16 quantized voltage level then we need at least 4 bits to encode each quantized voltage level with a unique binary code

Digital to Analog Conversion :

- To convert a Digital signal into an Analog signal we can use a Digital to analog converter(D/A)
- For a practical view point the simplest D/A converter is the zero order hold circuit which simply holds constant value of one sample until the next one is received.
- Some practical example of Digital to analog converter circuit is R-2R ladder and weighted resistor D/A converter.

Analysis of Digital Signals and Systems Vs Discrete Time Signals and Systems:

We have seen that a digital signal is defined as a function of an integer independent variable and its values are taken from a finite set of possible values. The usefulness of such signals is a consequence of the possibilities offered by digital computers. Computers operate on numbers which are represented by a string of 0's and 1's. The length of this string(word length) is fixed and finite and usually is 8, 16, 32 bits .The effects of finite word length in computations cause complications in the analysis of digital signal processing systems. To avoid these complications we neglect the quantized nature of digital signals and systems in much of our analysis and consider then as discrete- time signals and systems.

UNIT- 2

Representation of Discrete time signals:

There are different types of representation for discrete time signals. They are

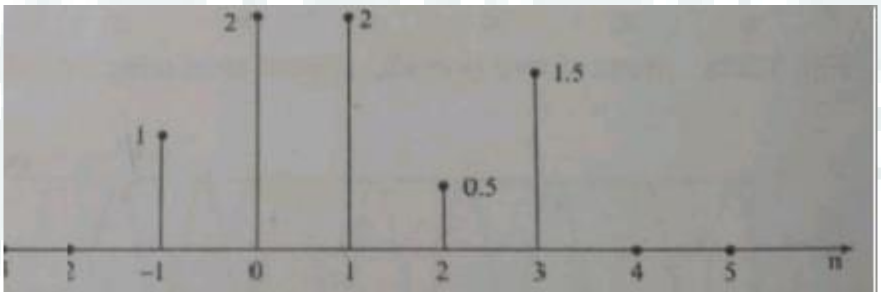
1) Graphical representation

2) Functional representation

3) Tabular representation

4) Sequence representation

<u>Graphical representation :</u> Let us consider a signal x(n) with values x(-1)=1; x(0)=2; x(1)=2; x(2)=0.5 and x(3)=1.5 .This discrete time signal can be represented graphically as shown



1.3: Graphical representation of a discrete-time signal.

<u>Functional Representation :</u>

The discrete time signal shown in fig 1.3 can be represented as below

$$x(n) = \begin{cases} 1 & \text{for } n = -1 \\ 2 & \text{for } n = 0, 1 \\ 0.5 & \text{for } n = 2 \\ 1.5 & \text{for } n = 3 \\ 0 & \text{otherwise} \end{cases}$$

<u>Tabular representation :</u>

The discrete time signal can also be represented as

n	-1	0	1	2	3
X(n)	1	2	2	0.5	1.5

<u>Sequence Representation:</u>

A finite duration sequence with time origin(n=0) indicated by the symbol ↑ is represented as

X(n)={1, 2, 2 , 0.5 , 1.5 }
↑

Elementary Discrete- time signals

Unit step sequence: The unit step sequence is defined as

$u(n) = 1$ for $n \geq 0$

$= 0$ for $n \leq 0$

The graphical representation of u(n) as shown

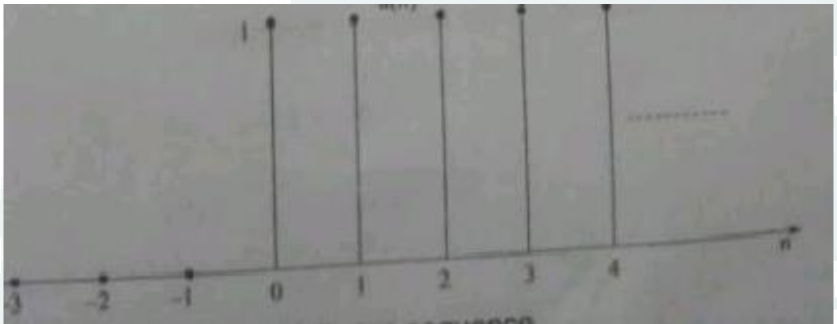


Fig. 1.33 Unit step sequence

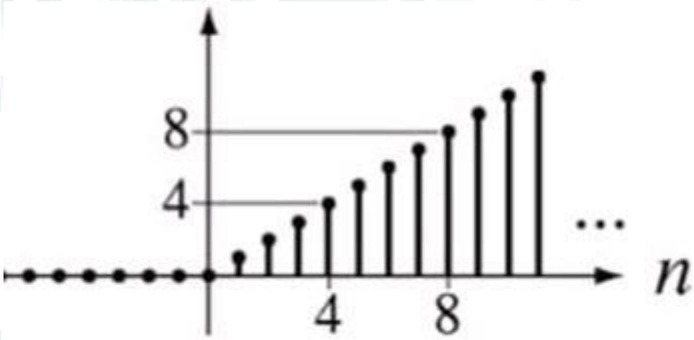
Unit Ramp Sequence :

The unit ramp sequence is defined as

$r(n)= n$ for $n\geq0$

$= 0$ for $n\leq 0$

The graphical representation of r(n) is shown



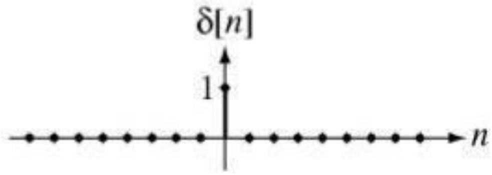
Unit impulse sequence:

The unit impulse sequence is defined as

$\delta [n]= 1$ for $n=0$

$= 0$ for $n \neq 0$

The graphical representation of δ [n] is shown



The unit impulse function has the following properties

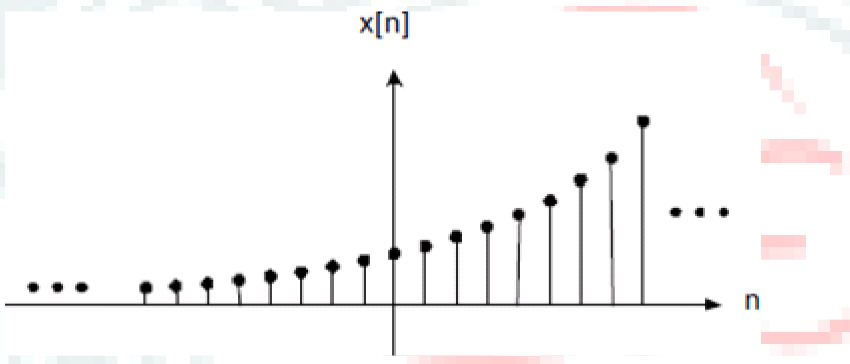
$$\delta(n) = u(n) - u(n-1)$$

Exponential Sequence:

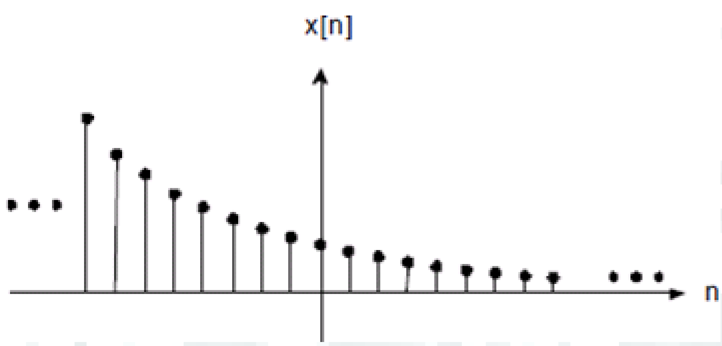
It is defined as x(n) = a^n for all n

Case 1: If a is real x[n] is real exponential

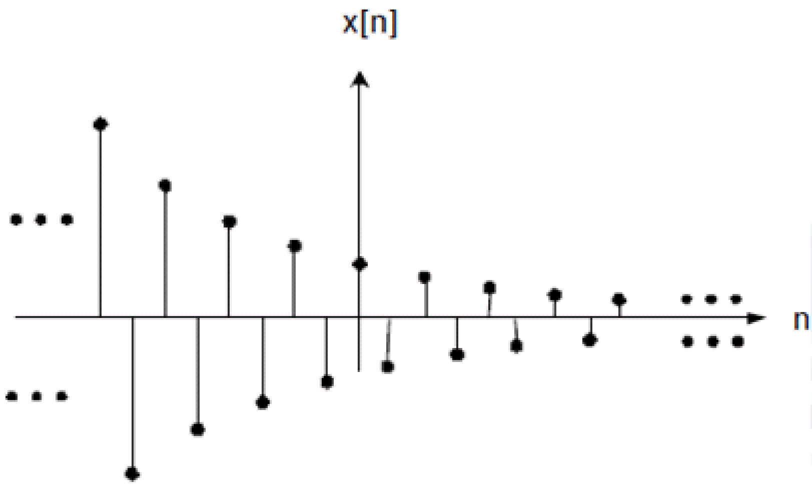
For a>1



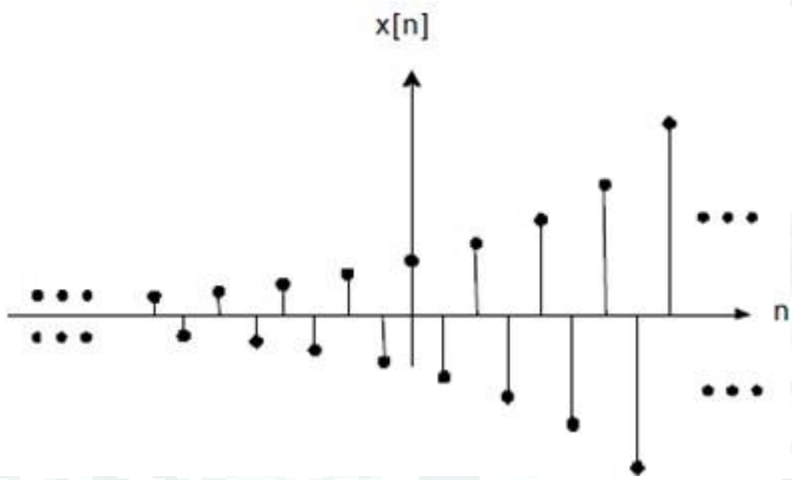
a < 1



-1 < a < 0



a < -1



Case 2: If "a" is complex valued then a can be expressed as a = $re^{j\theta}$, so that x[n] can be represented as

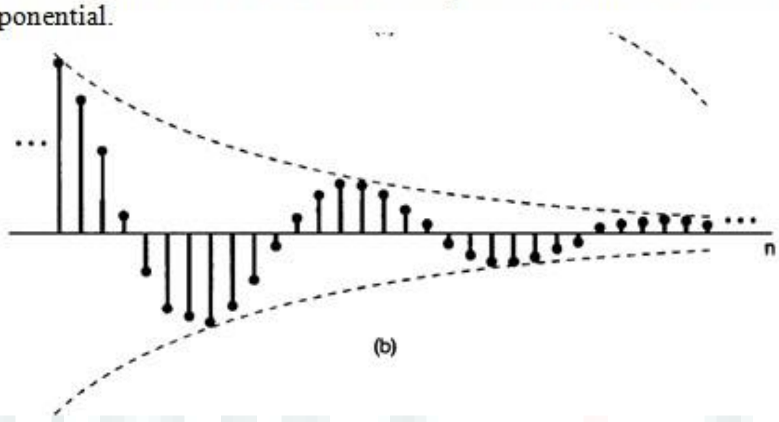
$$[n] = r^n e^{jn\theta}$$

$$= [\cos n\theta + j \sin n\theta]$$

So, x [n] is represented graphically by plotting the real part and imaginary parts separately as functions of n, which are

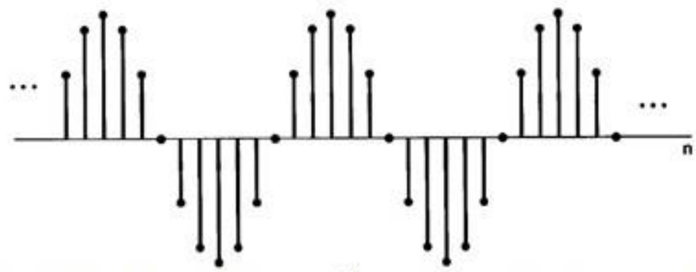
$$x_R[n] = r^n \cos n\theta$$
$$x_I[n] = r^n \sin n\theta$$

If r < 1, the above two functions are damped cosine and sine functions, whose amplitude is a decaying exponential.

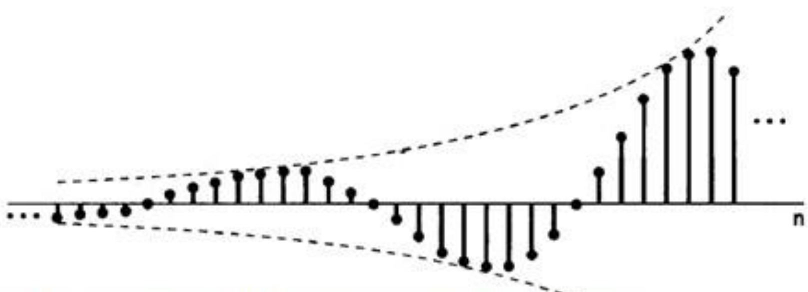


(b)

If r = 1, then both the functions have fixed amplitude of unity.



If r > 1, then they are cosine and sine functions respectively, with exponentially growing amplitudes.

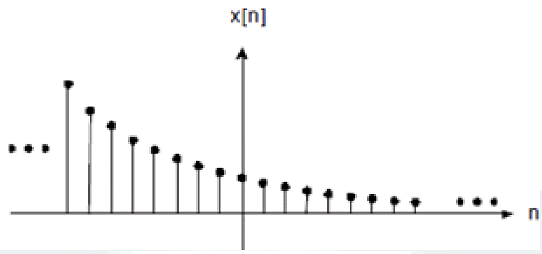


Alternatively, x [n] can be represented by the amplitude and phase functions:

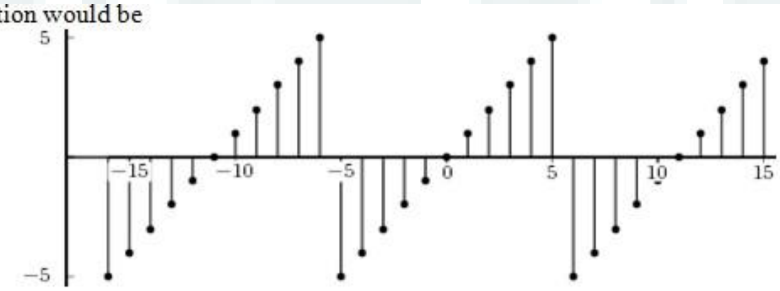
Amplitude function, $A[n] = |x[n]| = r^n$

Phase function, $\emptyset[n] = \angle x[n] = n\theta$

For example, for r < 1, the amplitude function would be



And the phase function would be



Although the phase function $\emptyset[n] = n\theta$ is a linear function of n, it is defined only over an interval of 2π (since it is an angle).i.e., over an interval $-\pi<\theta<\pi$ or $0<\theta<2\pi$.

Classification of Discrete time signal:

1. **Energy Signals and Power Signals**

The energy of a signal x[n] is defined as

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

If this energy is finite, i.e., $0 < E < \infty$, then x[n] is called an **Energy Signal**.

For signals having infinite energy, the average power can be calculated, which is defined as

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

$$or, P_{av} = \lim_{N\to\infty} \frac{1}{2N+1} E_N, \quad where$$

E_N = signal energy of x[n] over the finite interval $-N \le n \le N$, .i.e.,

$$E = \lim_{N\to\infty} E_N$$

- For signals with finite energy .i.e., for Energy Signals, E is finite, thus resulting in zero average power.

- Signals with infinite energy may have finite or infinite average power. If the average power is finite and nonzero, such signals are called **Power Signals**.
- Signals with finite power have infinite energy.

If both energy E as well as average power, P_{avg} of a signal are infinite, then the signal is neither an energy signal nor a power signal.

- Periodic signals have infinite energy. Their average power is equal to its average power over one period.
- A signal cannot both be an energy signal and a power signal.
- All practical signals are energy signals.

2. Periodic and aperiodic signals

A signal x[n] is periodic with period N if and only if

$$x[n+N] = x[n] \ \forall\, n$$

The smallest N for which the above relation holds is called the **fundamental period.**

If no finite value of N satisfies the above relation, the signal is said to be **aperiodic** or **non – periodic.**

The sum of M periodic Discrete – time sequences with periods N_1, N_2, …, N_M, is always periodic with period N where

$$N = LCM\,(N_1, N_2, \ldots\ldots N_M)$$

3. Even and Odd Signals

A real – valued discrete – time signal is called an **Even Signal** if it is identical with its reflection about the origin .i.e., it must be symmetrical about the vertical axis.

$$x[n] = x[-n] \ \ \forall n$$

A real – valued discrete – time signal is called an **Odd Signal** if it is anti symmetrical about the vertical axis.

$$x[n] = -x[-n] \ \forall n$$

From the above relation, it can be inferred that an odd signal must be zero at time origin, n = 0.

Every signal x[n] can be expressed as the sum of its even and odd components.

$$x[n] = x_e[n] + x_o[n]$$

Where

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

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

- Product of even and odd sequences results in an odd sequence.
- Product of two odd sequences results in an even sequence.
- Product of two even sequences results in an even sequence.

4. Conjugate Symmetric and Conjugate Anti symmetric sequences

A complex discrete – time signal is **conjugate – symmetric** if

$$x[n] = x^*[-n] \ \ \forall n$$

And **conjugate – anti symmetric** if

$$x[n] = -x^*[-n] \ \ \forall n$$

Any complex signal can be expressed as the sum of conjugate – symmetric and conjugate –anti symmetric parts

$$x[n] = x_{cs}[n] + x_{ca}[n]$$

Where

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

And

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

5. **Bounded and Unbounded sequences**

A discrete – time sequence x[n] is said to be **bounded** if each of its samples is of finite magnitude .i.e.,

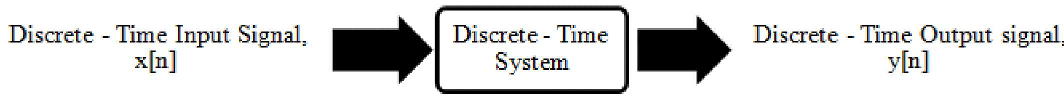
$$|x[n]| \le M_x < \infty \ \ \forall n$$

For example,

The unit step sequence u[n] is a bounded sequence,
but the sequence **nu[n]** is an unbounded sequence.

Discrete – Time Systems

A system accepts an input such as voltage, displacement, etc. and produces an output in response to this input.A system can be viewed as a process that results in transforming input signals into output signals.



A discrete – time system can be represented as

$$x[n] \ \rightarrow \ y[n]$$

or,

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

Classification of Discrete time systems

Discrete time system are classified according to their general properties and characteristics. They are

1. Static and Dynamic systems
2. Causal and Non causal systems
3. Linear and Non-linear systems
4. Time invariant and Time variant systems
5. FIR and IIR systems
6. Stable and Unstable system

1. Static and Dynamic systems

- A discrete time system is called static or memory less if its output at any instant "n" depends on the input samples at the same time , but not on the past or future samples of the input.
- In any other case the system is said to be dynamic or to have memory.

The system described by the following equations

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

$$y(n)= a\, x^2(n)$$

are static.

On the other hand , the systems described by the following euations

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

$$y(n)= x(n+1) + x(n)$$

are dynamic systems.

2. Causal and Non causal Systems

- A system is said to be causal if the output of the system at any instant of time "n" depends only at present and past inputs, but does not depend on future inputs. This can be represented mathematically as

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

- If the output of a system depends on future inputs, the system is said to be **non causal** or anticipatory.

Example

$y(n)= x(n) + x(n-1)$ causal system

$y(n)= x(2n)$ Non causal system

- It is not possible to build a practical system that responds to future inputs .Hence a non causal system is physically unrealizable.

Linear and Non linear Systems

- A system that satisfies the superposition principles is said to be a linear system. Superposition principle states that the response of the system to a weighted sum of signals should be equal to the corresponding weighted sum of the outputs of the system to each of the individual signals.
- A system is linear if and only if

$$T[a_1x_1(n) + a_2x_2(n)] = a_1T\,[x_1(n)] + a_2T[x_2(n)]$$

For any arbitary constant a_1 and a_2

- A system that does not satisfy the superposition principle is called non linear.

Time variant and Time invariant systems

A system is said to be time- invariant or shift invariant if the characteristics of the system do not change with time. For a time invariant system if y(n) is the response of the system to the input x(n), then the response of a system to the input x(n-k) is y(n-k). That is if the input sequence is shifted by "k" samples the generated output sequence is the original sequence shifted by "k" samples.

To test if any given system is time invariant, first apply an arbitrary sequence x(n) and find y(n). Now delay the input sequence by "k" samples and find output sequence

$$y(n,k) =T[x(n-k)]$$

Delay the output sequence by "k" samples , denote it as y(n-k) , if

$y(n,k) = y(n-k)$ for all possible value of k , the system is time invariant.

On the other hand if the output y(n,k)≠y(n-k) even for one vale of k, the system is time variant.

FIR and IIR system

FIR system: If the impulse response of the system is of finite duration, then the system is called a Finite Impulse Response.

An example of FIR system is

h(n)={ 1 for n=-1,2

0 , otherwise

IIR system: An infinite impulse response (IIR) system has an impulse response for infinite duration. An example for an IIR system is

$h(n)= a^n u(n)$

Stable and Unstable systems:

A **stable system** is one in which, a bounded input results in a response that does not diverge. Then the system is said to be **BIBO stable**.

For a system, if the input is bounded i.e.

$$if \ |x[n]| \leq M_x < \infty \ \forall n$$

And if the corresponding output is also bounded i.e.,

$$|y[n]| \leq M_y < \infty \ \forall n$$

Then the system is said to be **BIBO stable.**

Properties of Unit Impulse Sequence

Multiplication property

When a sequence x[n] is multiplied by a unit impulse located at k i.e., δ[n-k], picks out a single value/sample of x[n] at the location of the impulse i.e., x[k].

$$x[n]\delta[n-k] = x[k]\delta[n-k]$$
$$= impulse\ with\ strength\ x[k] located\ at\ n = k$$

Sifting property

The impulse function δ[n-k] "sifts" through the function x[n] and pulls out the value x[k]

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

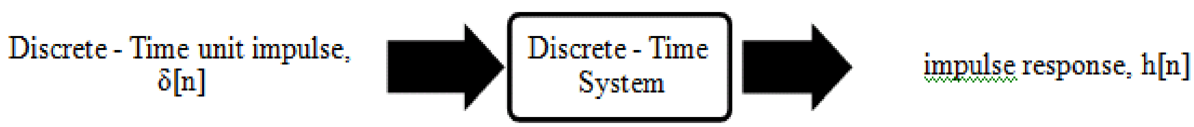
Signal decomposition

Any arbitrary sequence x[n] can be expressed as a weighted sum of shifted impulses.

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

Impulse response

Impulse response of a discrete – time system is defined as the output/response of the system to unit impulse input and is represented by **h[n]**.



$$x[n] \rightarrow y[n]$$

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

If the DT system satisfies the property of time – invariance, then,

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

In addition to being time – invariant, if the system also satisfies linearity (homogeneity and additivity), then,

Homogeneity:

Homogeneity:

$$x[k]\delta[n-k] \rightarrow x[k]h[n-k]$$

Additivity:

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

Combining the above two properties, a **Linear Time – Invariant (LTI)** System can be described by the input – output relation by

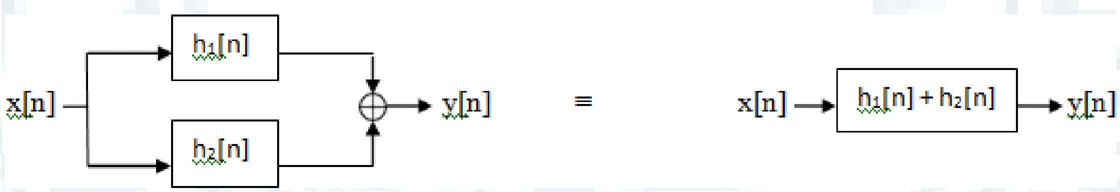
$$\sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \rightarrow \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

The Left hand side is the input x[n] expressed as a weighted sum of shifted impulses (from signal decomposition property of impulse function). So, the right hand side must be the output y[n] of the DT system in response to input x[n].

Interconnection of Discrete time system

1. Parallel connection of systems

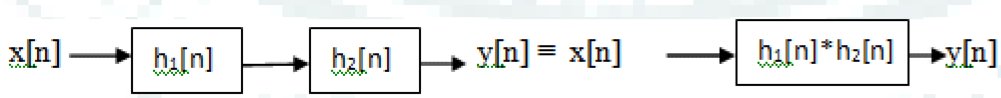
Consider two LTI systems with impulse response $h_1(n)$ and $h_2(n)$ connected in parallel as shown



If two systems are connected I parallel, then the overall impulse response is equal to sum of two impulse response.

2. Cascade connection of two systems

Consider two LTI systems with impulse responses $h_1(n)$ and $h_2(n)$ connected in cascade as shown



The impulse response of of two LTI system connected in cascade is the convolution of the individual impulse responses.

Response of LTI system to arbitrary inputs using convolution sum

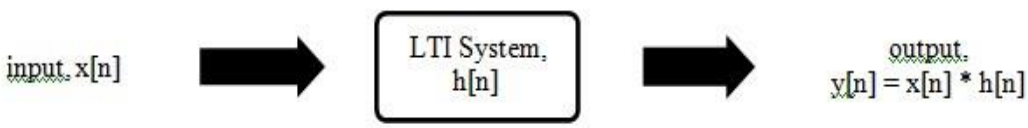
The input signal x(n) is the system excitation , and y(n) is the system response

Thus the output of a **Linear Time – Invariant (LTI) system** can be expressed as

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

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

The above relation is called **Convolution Sum.**

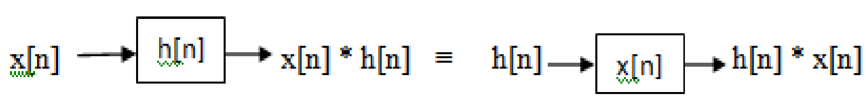


So, the impulse response h[n] of an LTI DT system completely characterizes the system .i.e., a knowledge of h[n] is sufficient to obtain the response of an LTI system to any arbitrary input x[n].

Convolution and interconnection of LTI system properties

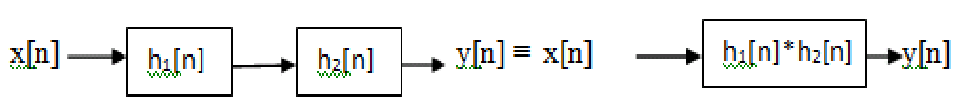
1. **Commutative Property**

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



2. **Associative Property**

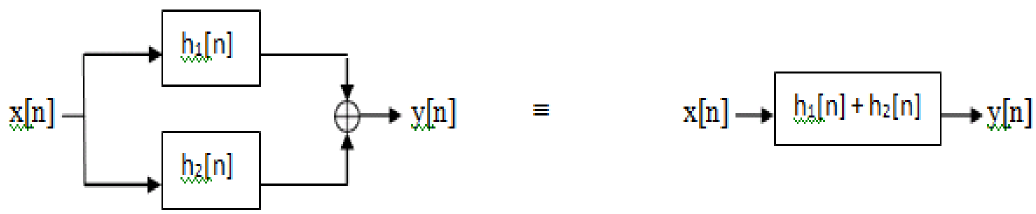
$$x[n] * \{h_1[n] * h_2[n]\} = \{x[n] * h_1[n]\} * h_2[n]$$



From this property it can be inferred that, a cascade combination of LTI systems can be replaced by a single system whose impulse response is the convolution of the individual impulse responses.

3. **Distributive Property**

$$x[n] * \{h_1[n] + h_2[n]\} = \{x[n] * h_1[n]\} + \{x[n] * h_2[n]\}$$



From this property, it can be inferred that, a parallel combination of LTI systems can be replaced by a single system whose impulse response is the sum of individual responses.

Study system with finite duration and infinite duration impulse response

Memory

For an LTI system to be memoryless, the impulse response must be zero for nonzero sample positions.

$$h[n] = 0 \; for \; n \neq 0$$

$$h[n] = k\,\delta\,[n] \; where \; k = constant$$

Causality

For an LTI system to be causal, its impulse response must be zero for negative time instants.

$$h[n] = 0 \; for \; n < 0$$

So, for a causal LTI system the output (from the convolution sum equation) can be expressed as

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

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

Stability

An LTI system is BIBO stable if its impulse response is absolutely summable.

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

Invertibility

An LTI system with impulse response h[n] is invertible if we can design another LTI system with impulse response $h_I[n]$ such that

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

Study systems with finite duration and infinite duration impulse response.

Finite impulse Response(FIR):

If the impulse response of the system is of finite duration, then the system is called a Finite Impulse Response. An example of a FIR system is

h(n)= { 1 for n=-1,2

2 for n=1

0 otherwise }

Infinite Impulse Response(IIR):

An Infinite Impulse Response (IIR) system has an impulse response for infinite duration. An example of an IIR system is

$h(n)= a^n u(n)$

UNIT 3 – Z-Transform

Discrete Time Fourier Transform(DTFT) exists for energy and power signals. Z-transform also exists for neither energy nor Power (NENP) type signal, up to a certain extent only. The replacement $Z = e^{j\omega}$ is used for Z-transform to DTFT conversion only for absolutely summable signal.

So, the Z-transform of the discrete time signal x(n) in a power series can be written as-

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

The above equation represents a two-sided Z-transform equation.

Generally, when a signal is Z-transformed, it can be represented as-

$$X(Z) = Z[x(n)]$$

Or $$x(n) \leftrightarrow X(Z)$$

If it is a continuous time signal, then Z-transforms are not needed because Laplace transformations are used. However, Discrete time signals can be analyzed through Z-transforms only.

Region of Convergence:

Region of Convergence is the range of complex variable Z in the Z-plane. The Z- transformation of the signal is finite or convergent. So, ROC represents those set of values of Z, for which X(Z) has a finite value.

Properties of ROC

1. ROC does not include any pole.
2. For right-sided signal, ROC will be outside the circle in Z-plane.
3. For left sided signal, ROC will be inside the circle in Z-plane.
4. For stability, ROC includes unit circle in Z-plane.
5. For Both sided signal, ROC is a ring in Z-plane.
6. For finite-duration signal, ROC is entire Z-plane.

The Z-transform is uniquely characterized by:

1. Expression of X(Z)
2. ROC of X(Z)

Signals and their ROC

x(n)	X(Z)	ROC
δ (n)	1	Entire Z plane
U(n)	$1/(1-Z^{-1})$	Mod(Z)>1
$a^n u(n)$	$1/(1-aZ^{-1})$	Mod(z)>Mod(a)
$-a^n u(-n-1)$	$1/(1-aZ^{-1})$	Mod(z)<Mod(a)
$na^n u(n)$	$aZ^{-1}/(1-aZ^{-1})^2$	Mod(z)>Mod(a)
$-na^n u(-n-1)$	$aZ^{-1}/(1-aZ^{-1})^2$	Mod(z)<Mod(a)
U(n)cos ωn	$(Z^2 - ZCos\omega)/(Z^2 - 2ZCos\omega + 1)$	Mod(z)>1
$U(n)sin\omega n$	$(Zsin\omega)/(Z^2 - 2ZCos\omega + 1)$	Mod(Z)>1

3.1.1 Direct Z-transform

Let us find the Z-transform and the ROC of a signal given as $x(n) = \{7,3,4,9,5\}$, where origin of the series is at 3.

Solution: Applying the formula we have:

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

$$=\sum_{n=-1}^{3} x(n)Z^{-n}$$

$$=x(-1)Z + x(0) + x(1)Z^{-1} + x(2)Z^{-2} + x(3)Z^{-3}$$

$$=7Z + 3 + 4Z^{-1} + 9Z^{-2} + 5Z^{-3}$$

ROC is the entire Z-plane excluding Z = 0, ∞, -∞

Properties of Z-Transform:

In this chapter, we will understand the basic properties of Z-transforms.

Linearity

It states that when two or more individual discrete signals are multiplied by constants, their respective Z-transforms will also be multiplied by the same constants.

Mathematically,

$$a1x1(n) + a2x2(n) = a1X1(z) + a2X2(z)$$

Proof: We know that,

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

$= \sum_{n=-\infty}^{\infty}(a1x1(n) + a2x2(n))Z^{-n}$

$= a1\sum_{n=-\infty}^{\infty} x1(n)Z^{-n} + a2\sum_{n=-\infty}^{\infty} x2(n)Z^{-n}$

$= a1X1(z) + a2X2(z)$ (Hence Proved)

Here, the ROC is $ROC1 \cap ROC2$.

Time Shifting

Time shifting property depicts how the change in the time domain in the discrete signal will affect the Z-domain, which can be written as;

$$x(n - n0) \leftrightarrow X(Z)Z^{-n}$$

Or $$x(n-1) \leftrightarrow Z^{-1}X(Z)$$

Proof:

Let y(p) =x(p-k)

Y(z) $=\sum_{p=-\infty}^{\infty} y(p)Z^{-p}$

$= \sum_{p=-\infty}^{\infty}(x(p-k))Z^{-p}$

Let s=p-k

$=\sum_{s=-\infty}^{\infty} x(s)Z^{-(s+k)}$

$= \sum_{s=-\infty}^{\infty} x(s)Z^{-s}\,Z^{-K}$

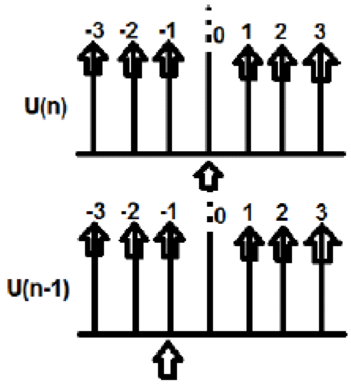
$=Z^{-k}[\sum_{s=-\infty}^{\infty} x(m)Z^{-s}]$

$=Z^{-k}X(Z)$ (Hence Proved)

Here, ROC can be written as Z=0 (p>0) or Z=∞(p<0)

Example:

U(n) and U(n-1) can be plotted as follows



Z-transformation of U(n) cab be written as;

$$\sum_{n=-\infty}^{\infty} [U(n)]Z^{-n} = 1$$

Z-transformation of U(n-1) can be written as;

$$\sum_{n=-\infty}^{\infty} [U(n-1)]Z^{-n} = Z^{-1}$$

So here $x(n-n0) = Z^{-no}X(Z)$ (Hence Proved)

Time Scaling

Time Scaling property tells us, what will be the Z-domain of the signal when the time is scaled in its discrete form, which can be written as;

$$a^n x(n) \leftrightarrow X(a^{-1}z)$$

Proof:

Let $y(p) = a^p x(p)$

Y(p) $= \sum_{p=-\infty}^{\infty} y(p) Z^{-p}$

$= \sum_{p=-\infty}^{\infty} a^p x(p)\, z^{-p}$

$= \sum_{p=-\infty}^{\infty} x(p)\, [a^{-1}z]^{-p}$

$= X(a^{-1}Z)$ (Hence proved)

ROC: $= \mathrm{Mod}(ar1) < \mathrm{Mod}(Z) < \mathrm{Mod}(ar2)$ where Mod = Modulus

Example

Let us determine the Z-transformation of $x(n) = a^n cos\omega n$ using Time scaling property.

Solution:

We already know that the Z-transformation of the signal Cos(ωn) is given by:

$$\sum_{n=-\infty}^{\infty}(Cos\omega n)Z^{-n} = (Z^2 - ZCos\omega)/(Z^2 - 2ZCos\omega + 1)$$

Now, applying Time scaling property, the Z-transformation of $a^n cos\omega n$ can be written as;

$$\sum_{n=-\infty}^{\infty}(a^n cos\omega n)Z^{-n} = X(a^{-1}Z)$$

$$= [(a^{-1}Z)^2 - (a^{-1}ZCos\omega n)]/((a^{-1}Z)^2 - 2(a^{-1}ZCos\omega n) + 1)$$

$$= Z(Z - aCos\omega)/(Z^2 - 2azCos\omega + a^2)$$

Successive Differentiation

Successive Differentiation property shows that Z-transform will take place when we differentiate the discrete signal in time domain, with respect to time. This is shown as below.

$$\frac{dx(n)}{dn} = (1 - Z^{-1})X(Z)$$

Proof:

Consider the LHS of the equation: $\frac{dx(n)}{dn}$

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

$$= x(n) - X(n-1)$$

$$= x(z) - Z^{-1}x(z)$$

$$= (1 - Z^{-1})x(Z)$$ (Hence Proved)

ROC: R1< Mod (Z) <R2

Example

Let us find the Z-transform of a signal given by $x(n) = n^2 u(n)$

By property we can write

$$Z[nU(n)] \quad = -Z\frac{dZ[U(n)]}{dz}$$

$$= -Z\frac{d[\frac{z}{z-1}]}{dz}$$

$$= Z/((z-1)^2$$

$$= y \text{ (let)}$$

Now, Z[n.y] can be found out by again applying the property,

$$Z(n.y) = -Z\frac{dy}{dz}$$

$$= -Z\frac{d\,[Z/((z-1)^2]}{dZ}$$

$$= Z(Z+1)/(Z-1)^2$$

Convolution

This depicts the change in Z-domain of the system when a convolution takes place in the discrete signal form, which can be written as-

$$x1(n) * x2(n) \leftrightarrow X1(z).X2(z)$$

Proof:

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

$$= \sum_{n=-\infty}^{\infty}[\sum_{k=-\infty}^{\infty} x1(k)x2(n-k)]\, Z^{-n}$$

$$= \sum_{k=-\infty}^{\infty} x1(k)[\sum_{n=-\infty}^{\infty} x2(n-k)Z^{-n}]$$

$$= \sum_{k=-\infty}^{\infty} x1(k)[\sum_{n=-\infty}^{\infty} x2(n-k)Z^{-(n-k)}Z^{-k}]$$

Let n-k = l, then the above equation cab be written as:

$$X(z) \quad = \sum_{k=-\infty}^{\infty} x1(k)[Z^{-k}\sum_{l=-\infty}^{\infty} x2(l)Z^{-l}]$$

$$= \sum_{k=-\infty}^{\infty} x1(k)X2(Z)Z^{-k}$$

$$= X2(z)\sum_{k=-\infty}^{\infty} x1(z)Z^{-k}$$

$$= X1(z).X2(z) \qquad \text{(Hence proved)}$$

ROC: ROC$\cap$ $ROC2$

Example

Let us find the convolution given by two signals

x1(n)={ 3,-2,2} ...(eq. 1)

x2(n)={2,0≤n≤4 and 0 elsewhere} ...(eq. 2)

Z-transformation of the first equation can be written as;

$$\sum_{n=-\infty}^{\infty} x1(n)Z^{-n}$$

$$=3 - 2Z^{-1} + 2Z^{-2}$$

Z-transformation of the second signal can be written as;

$$\sum_{n=-\infty}^{\infty} x2(n)Z^{-n}$$

$$=2 + 2Z^{-1} + 2Z^{-2} + 2Z^{-3} + 2Z^{-4}$$

So, the convolution of the above two signals is given by:

X(z)=[x1(z)*x2(z)]

$$=[3 - 2Z^{-1} + 2Z^{-2}] \times [2 + 2Z^{-1} + 2Z^{-2} + 2Z^{-3} + 2Z^{-4}]$$

$$=6 + 2Z^{-1} + 6Z^{-2} + 6Z^{-3} + \dots \dots \dots$$

Taking the inverse Z-transformation we get,

$$x(n) = \{6,2,6,6,6,0,4\}$$

Initial Value Theorem

If x(n) is a causal sequence, which has its Z-transformation as X(z), then the initial value theorem can be written as;

$$X(n) \text{ (at n=0)}= \lim_{z\to\infty} X(z)$$

Proof: We know that,

$$X(Z) = \sum_{n=0}^{\infty} x(n)Z^{-n}$$

Expanding the above series, we get;

$$= X(0)Z^{0} + X(1)Z^{-1} + X(2)Z^{-2} + \cdots ..$$

$$=X(0) \times 1 + X(1)Z^{-1} + X(2)Z^{-2} + \cdots \dots .$$

In the above case if z→ ∞ then $Z^{-n} \to 0$ (Because n>0)

Therefore, we can say;

$$\lim_{z\to\infty} X(z) = X(0)$$ (Hence Proved)

Final Value Theorem

Final Value Theorem states that if the Z-transform of a signal is represented as X(Z) and the poles are all inside the circle, then its final value is denoted as x(n) or X(∞) and can be written as-

$$X(\infty) = \lim_{n\to\infty} X(n) = \lim_{z\to 1}[X(z)(1 - Z^{-1})]$$

Conditions:

1. It is applicable only for causal systems.
2. $X(z)(1 - Z^{-1})$ should have poles inside the unit circle in Z-plane.

Example

Let us find the Initial and Final value of x(n) whose signal is given by

$X(Z) = 2 + 3Z^{-1} + 4Z^{-2}$

Solution: Let us first, find the initial value of the signal by applying the theorem

$$x(0) = \lim_{Z\to\infty} X(Z)$$

$$= \lim_{Z\to\infty}[2 + 3Z^{-1} + 4Z^{-2}]$$

$$= 2 + \left(\frac{3}{\infty}\right) + \left(\frac{4}{\infty}\right) = 2$$

Now let us find the Final value of signal applying the theorem

$$x(\infty) = \lim_{Z\to\infty}[(1 - Z^{-1})X(Z)]$$

$$= \lim_{Z\to\infty}[(1 - Z^{-1})(2 + 3Z^{-1} + 4Z^{-2})]$$

$$= \lim_{Z\to\infty}[2 + Z^{-1} + Z^{-2} - 4Z^{-3}]$$

$$= 2+1+1-4 = 0$$

Differentiation in Frequency

It gives the change in Z-domain of the signal, when its discrete signal is differentiated with respect to time.

$$nx(n) \leftrightarrow -Z\frac{dX(z)}{dz}$$

Its ROC can be written as;

$$\text{r2} < Mod(Z) < r1$$

Example:

Let us find the value of x(n) through Differentiation in frequency, whose discrete signal in Z-domain is given by $x(n) \leftrightarrow X(Z) = \log(1 + aZ^{-1})$

By property, we can write that

$$nx(n) \leftrightarrow -Z\frac{dx(z)}{dz}$$

$$=-Z\left[\frac{-aZ^{-2}}{1+aZ^{-1}}\right]$$

$$=(aZ^{-1})/(1+aZ^{-1})$$

$$=1-1/(1+aZ^{-1})$$

$$nx(n)=\delta(n)-(-a)^n u(n)$$

$$=> x(n)=1/n[\delta(n)-(-a)^n u(n)]$$

Multiplication in Time

It gives the change in Z-domain of the signal when multiplication takes place at discrete signal level.

$$x1(n).x2(n) \leftrightarrow \left(\frac{1}{2\Pi j}\right)[X1(z) * X2(z)]$$

Conjugation in Time

This depicts the representation of conjugated discrete signal in Z-domain.

$$X^*(n) \leftrightarrow X^*(Z^*)$$

Example 1

Let us try to find out the Z-transform of the signal, which is given as

$$x(n)=-(-0.5)^{-n}u(-n)+3^n u(n)$$

$$=-(-2)^n u(n)+3^n u(n)$$

Solution: Here, for $-(-2)^n u(n)$ the ROC is Left sided and Z<2

For $3^n u(n)$ ROC is right sided and Z>3

Hence, here Z-transform of the signal will not exist because there is no common region.

Example 2

Let us try to find out the Z-transform of the signal given by

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

Solution: Here, for $-2^n u(-n-1)$ ROC of the signal is Left sided and Z<2

For signal $(0.5)^n u(n)$ ROC is right sided and Z>0.5

So, the common ROC being formed as 0.5<Z<2

Therefore, Z-transform can be written as;

$$X(Z) = \left\{\frac{1}{1-2Z^{-1}}\right\} + \left\{\frac{1}{(1-0.5Z)^{-1}}\right\}$$

Example 3

Let us try to find out the Z-transform of the signal, which is given as $x(n) = 2^{r(n)}$

Solution: r(n) is the ramp signal. So the signal can be written as;

x(n)= $2^{nu(n)}$ { 1, n<0 (u(n)=0) and 2^n , n≥0 (u(n)=1)}

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

Here, for the signal $u(-n-1)$ and ROC Z<1 and for $2^n u(n)$ with ROC is Z>2.

So, Z-transformation of the signal will not exist.

Rational Z-transform. :

Poles & Zeros:

Most useful z-transforms can be expressed in the form

$$X(z) = \frac{P(z)}{Q(z)},$$

where $P(z)$ and $Q(z)$ are polynomials in z. The values of z for which $P(z) = 0$ are called the **zeros** of $X(z)$, and the values with $Q(z) = 0$ are called the **poles**. The zeros and poles completely specify $X(z)$ to within a multiplicative constant.

Example: right-sided exponential sequence

Consider the signal $x[n] = a^n u[n]$. This has the z-transform

$$X(z) = \sum_{n=-\infty}^{\infty} a^n u[n] z^{-n} = \sum_{n=0}^{\infty} (az^{-1})^n.$$

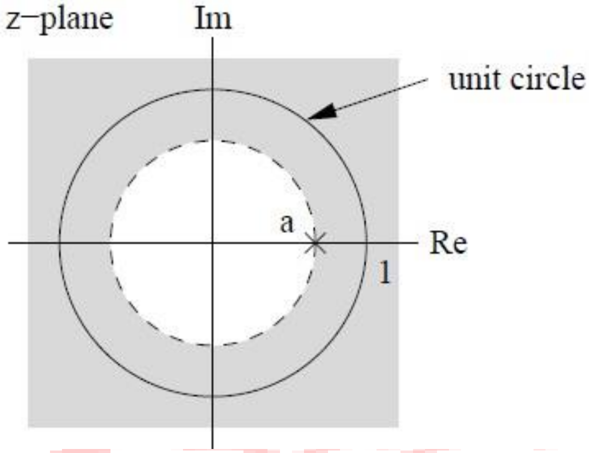
Convergence requires that

$$\sum_{n=0}^{\infty} |az^{-1}|^n < \infty,$$

which is only the case if $|az^{-1}| < 1$, or equivalently $|z| > |a|$. In the ROC, the series converges to

$$X(z) = \sum_{n=0}^{\infty}(az^{-1})^n = \frac{1}{1-az^{-1}} = \frac{z}{z-a}, \qquad |z| > |a|,$$

since it is just a geometric series. The z-transform has a region of convergence for any finite value of a.



The Fourier transform of $x[n]$ only exists if the ROC includes the unit circle, which requires that $|a| < 1$. On the other hand, if $|a| > 1$ then the ROC does not include the unit circle, and the Fourier transform does not exist. This is consistent with the fact that for these values of a the sequence $a^n u[n]$ is exponentially growing, and the sum therefore does not converge.

Example: left-sided exponential sequence

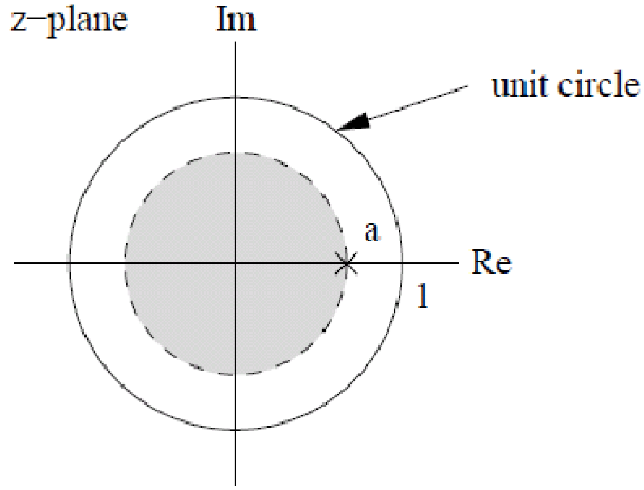
Now consider the sequence $x[n] = -a^n u[-n-1]$. This sequence is left-sided because it is nonzero only for $n \le -1$. The z-transform is

$$X(z) = \sum_{n=-\infty}^{\infty} -a^n u[-n-1] z^{-n} = -\sum_{n=-\infty}^{-1} a^n z^{-n}$$

$$= -\sum_{n=1}^{\infty} a^{-n} z^n = 1 - \sum_{n=0}^{\infty} (a^{-1}z)^n.$$

For $|a^{-1}z| < 1$, or $|z| < |a|$, the series converges to

$$X(z) = 1 - \frac{1}{1-a^{-1}z} = \frac{1}{1-az^{-1}} = \frac{z}{z-a}, \qquad |z| < |a|.$$



Pole location & Time domain behavior for causal Signal:

Causal system can be defined as $h(n) = 0, n < 0$. For causal system, ROC will be outside the circle in Z-plane.

$$H(z) = \sum_{n=0}^{\infty} h(n)Z^{-n}$$

Expanding the above equation,

$$H(z) = h(0) + h(1)Z^{-1} + h(2)Z^{-2} + \cdots \ldots \ldots$$

=N(Z)/D(Z)

For causal systems, expansion of Transfer Function does not include positive powers of Z. For causal system, order of numerator cannot exceed order of denominator. This can be written as-

$$\lim_{z \to \infty} H(z) = h(0) = 0 \; or \; Finite$$

For stability of causal system, poles of Transfer function should be inside the unit circle in Z-plane.

Inverse Z-Transform

If we want to analyze a system, which is already represented in frequency domain, as discrete time signal then we go for Inverse Z-transformation.

Mathematically, it can be represented as;

$$x(n) = Z^{-1}X(Z)$$

where x(n) is the signal in time domain and X(Z) is the signal in frequency domain.

If we want to represent the above equation in integral format then we can write it as

$$x(n) = (\frac{1}{2\Pi j})\oint X(Z)Z^{-1}dz$$

Here, the integral is over a closed path C. This path is within the ROC of the x(z) and it does contain the origin.

Methods to Find Inverse Z-Transform

When the analysis is needed in discrete format, we convert the frequency domain signal back into discrete format through inverse Z-transformation. We follow the following four ways to determine the inverse Z-transformation.

1. Long Division Method
2. Partial Fraction expansion method
3. Residue or Contour integral method

Long Division Method

In this method, the Z-transform of the signal x (z) can be represented as the ratio of polynomial as shown below;

$$x(z) = N(Z)/D(Z)$$

Now, if we go on dividing the numerator by denominator, then we will get a series as shown below

$$X(z) = x(0) + x(1)Z^{-1} + x(2)Z^{-2} + \cdots\cdots\cdots\cdots$$

The above sequence represents the series of inverse Z-transform of the given signal (for $n \geq 0$) and the above system is causal.

However for $n<0$ the series can be written as;

$$x(z) = x(-1)Z^{1} + x(-2)Z^{2} + x(-3)Z^{3} + \cdots\cdots\cdots\cdots$$

Partial Fraction Expansion Method

Here also the signal is expressed first in N (z)/D (z) form.

If it is a rational fraction it will be represented as follows;

$$x(z) = (b0 + b1Z^{-1} + b2Z^{-2} + \cdots\cdots\cdots + bmZ^{-m})/(a0 + a1Z^{-1} + a2Z^{-2} + \cdots\cdots\cdots + anZ^{-N})$$

The above one is improper when $m<n$ and $an \neq 0$

If the ratio is not proper (i.e. Improper), then we have to convert it to the proper form to solve it.

Residue or Contour Integral Method

In this method, we obtain inverse Z-transform x(n) by summing residues of $[x(z)Z^{n-1}]$ at all poles. Mathematically, this may be expressed as

$$x(n) = \sum_{all\ poles\ X(z)} residues\ of\ [x(z)Z^{n-1}]$$

Here, the residue for any pole of order m at $z = \beta$ is

$$Residue = \frac{1}{(m-1)!}\lim_{z\to\beta}\{\frac{d^{m-1}}{dZ^{m-1}}\{(z-\beta)^m X(z)Z^{n-1}\}$$

Example 1

Find the response of the system $s(n+2) - 3s(n+1) + 2s(n) = \delta(n)$, when all the initial conditions are zero.

Solution: Taking Z-transform on both the sides of the above equation, we get

$$S(z)Z^2 - 3S(z)Z^1 + 2S(z) = 1$$

⇨ $S(z)\{Z^2 - 3Z + 2\} = 1$

⇨ $S(z) = \frac{1}{\{Z^2-3Z+2\}} = \frac{1}{(z-2)(z-1)} = \frac{\alpha_1}{Z-2} + \frac{\alpha_2}{Z-1}$

⇨ $S(z) = \frac{1}{Z-2} - \frac{1}{Z-1}$

Taking the inverse Z-transform of the above equation, we get

$$S(n) = Z^{-1}\left[\frac{1}{Z-2}\right] - Z^{-1}[\frac{1}{Z-1}]$$

$$= 2^{n-1} - 1^{n-1} = -1 + 2^{n-1}$$

Example 2

Find the system function H(z) and unit sample response h(n) of the system whose difference equation is described as under

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

where, y(n) and x9n) are the output and input of the system, respectively.

Solution: Taking the Z-transform of the above difference equation, we get

$$y(z) = \frac{1}{2}Z^{-1}Y(Z) + 2X(z)$$

$$= Y(Z)\left[1 - \frac{1}{2}Z^{-1}\right] = 2X(Z)$$

$$= H(Z) = \frac{Y(Z)}{X(Z)} = \frac{2}{\left[1 - \frac{1}{2}Z^{-1}\right]}$$

This system has a pole at $Z = \frac{1}{2} and\ Z = 0\ \ and\ H(Z) = \frac{2}{\left[1-\frac{1}{2}Z^{-1}\right]}$

Hence, taking the inverse Z-transform of the above, we get

$$h(n) = 2(\frac{1}{2})^n U(n)$$

UNIT-4

Any signal can be decomposed in terms of sinusoidal (or complex exponential) components. Thus the analysis of signals can be done by transforming time domain signals into frequency domain and vice-versa. This transformation between time and frequency domain is performed with the help of Fourier Transform(FT) But still it is not convenient for computation by DSP processors hence Discrete Fourier Transform(DFT) is used.

Time domain analysis provides some information like amplitude at sampling instant but does not convey frequency content & power, energy spectrum hence frequency domain analysis is used.

For Discrete time signals x(n) , Fourier Transform is denoted as x(ω) & given by

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)\, e^{-j\omega n} \qquad \textbf{FT ....(1)}$$

DFT is denoted by x(k) and given by (ω= 2 ∏ k/N)

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{-j2\pi kn/N} \qquad \textbf{DFT ....(2)}$$

IDFT is given as

$$x(n) = 1/N \sum X(k)\, e^{j2\pi kn/N} \qquad \textbf{IDFT......(3)}$$

DIFFERENCE BETWEEN FT & DFT

S. No	Fourier Transform (FT)	Discrete Fourier Transform (DFT)
1	FT x(ω) is the continuous function of x(n).	DFT x(k) is calculated only at discrete values of ω. Thus DFT is discrete in nature.
2	The range of ω is from - ∏ to ∏ or 0 to 2∏.	Sampling is done at N equally spaced points over period 0 to 2∏. Thus DFT is sampled version of FT.
3	FT is given by equation (1)	DFT is given by equation (2)

	of infinite sequences.	duration sequences
5	In DSP processors & computers applications of FT are limited because x(ω) is continuous function of ω.	In DSP processors and computers DFT's are mostly used. **APPLICATION** a) Spectrum Analysis b) Filter Design

Tutorial problems:

Q) Prove that FT x(ω) is periodic with period 2∏.
Q) Determine FT of $x(n)= a^n u(n)$ for $-1< a < 1$.
Q) Determine FT of x(n)= A for $0 \le n \le L-1$.

Q) Determine FT of x(n)= u(n)

Q) Determine FT of x(n)= δ(n)

Q) Determine FT of $x(n)= e^{-at} u(t)$

CALCULATION OF DFT & IDFT

For calculation of DFT & IDFT two different methods can be used. First method is using mathematical equation & second method is 4 or 8 point DFT. If x(n) is the sequence of N samples then consider $W_N= e^{-j2 \prod / N}$ (twiddle factor)

Four POINT DFT (4-DFT)

S. No	$W_N=W_4=e^{-j \prod/2}$	Angle	Real	Imaginary	Total
1	W_4^0	0	1	0	1
2	W_4^1	- ∏/2	0	-j	-j
3	W_4^2	- ∏	-1	0	-1
4	W_4^3	- 3 ∏ /2	0	J	J

		n=0	n=1	n=2	n=3
$[W_N]$ =	k=0	W_4^0	W_4^0	W_4^0	W_4^0
		W_4^0	W_4^1	W_4^2	W_4^3
	k=2	W_4^0	W_4^2	W_4^4	W_4^6
	k=3	W_4^0	W_4^3	W_4^6	W_4^9

Thus 4 point DFT is given as $X_N = [W_N] X_N$

$$[W_N] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

EIGHT POINT DFT (8-DFT)

S. No	$W_N = W_8 = e^{-j\Pi/4}$	Angle	Magnitude	Imaginary	Total
1	W_8^{0}	0	1	----	1
2	W_8^{1}	$-\Pi/4$	$1/\sqrt{2}$	$-j\,1/\sqrt{2}$	$1/\sqrt{2} - j\,1/\sqrt{2}$
3	W_8^{2}	$-\Pi/2$	0	-j	-j
4	W_8^{3}	$-3\Pi/4$	$-1/\sqrt{2}$	$-j\,1/\sqrt{2}$	$-1/\sqrt{2} - j\,1/\sqrt{2}$
5	W_8^{4}	$-\Pi$	-1	----	-1
6	W_8^{5}	$-5\Pi/4$	$-1/\sqrt{2}$	$+j\,1/\sqrt{2}$	$-1/\sqrt{2} + j\,1/\sqrt{2}$
7	W_8^{6}	$-7\Pi/4$	0	J	J
8	W_8^{7}	-2Π	$1/\sqrt{2}$	$+j\,1/\sqrt{2}$	$1/\sqrt{2} + j\,1/\sqrt{2}$

Remember that $W_8^{0} = W_8^{8} = W_8^{16} = W_8^{24} = W_8^{32} = W_8^{40}$ (Periodic Property)
Magnitude and phase of x(k) can be obtained as,
$|x(k)| = \text{sqrt}\,(X_r(k)^2 + X_I(k)^2)$
Angle $x(k) = \tan^{-1} (X_I(k) / X_R(k))$

Tutorial problems:

Q) Compute DFT of x(n) = {0,1,2,3} Ans: x4=[6, -2+2j, -2, -2-2j]
Q) Compute DFT of x(n) = {1,0,0,1} Ans: x4=[2, 1+j, 0, 1-j]
Q) Compute DFT of x(n) = {1,0,1,0} Ans: x4=[2, 0, 2, 0]
Q) Compute IDFT of x(k) = {2, 1+j, 0, 1-j } Ans: x4=[1,0,0,1]

DIFFERENCE BETWEEN DFT & IDFT

S.No	DFT (Analysis transform)	IDFT (Synthesis transform)
1	DFT is finite duration discrete frequency sequence that is obtained by sampling one period of FT.	IDFT is inverse DFT which is used to calculate time domain representation (Discrete time sequence) form of x(k).
2	DFT equations are applicable to causal finite duration sequences.	IDFT is used basically to determine sample response of a filter for which we know only transfer function.
3	Mathematical Equation to calculate DFT is given by $$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{-j2\Pi kn/N}$$	Mathematical Equation to calculate IDFT is given by $$x(n) = 1/N \sum_{n=0}^{N-1} X(k) e^{j2\Pi kn/N}$$
4	Thus DFT is given by **$X(k)= [W_N][xn]$**	In DFT and IDFT difference is of factor 1/N & sign of exponent of twiddle factor. Thus **$x(n)= 1/N\, [W_N]^{-1}[X_K]$**

PROPERTIES OF DFT

$$x(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x(k)$$

1. Periodicity

Let x(n) and x(k) be the DFT pair then if

$x(n+N) = x(n)$ for all n then

$X(k+N) = X(k)$ for all k

Thus periodic sequence xp(n) can be given as

$$xp(n) = \sum_{l=-\infty}^{\infty} x(n-lN)$$

2. Linearity

The linearity property states that if

$$x1(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X1(k) \text{ And}$$

$$x2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} X2(k) \text{ Then}$$

Then

$$a1\,x1(n) + a2\,x2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} a1\,X1(k) + a2\,X2(k)$$

DFT of linear combination of two or more signals is equal to the same linear combination of DFT of individual signals.

3. Circular Symmetries of a sequence

A) A sequence is said to be circularly even if it is symmetric about the point zero on the circle. Thus $X(N-n) = x(n)$

B) A sequence is said to be circularly odd if it is anti symmetric about the point zero on the circle. Thus $X(N-n) = - x(n)$

C) A circularly folded sequence is represented as $x((-n))_N$ and given by $x((-n))_N = x(N-n)$.

D) Anticlockwise direction gives delayed sequence and clockwise direction gives advance sequence. Thus delayed or advances sequence x`(n) is related to x(n) by the circular shift.

4. Symmetry Property of a sequence

A) Symmetry property for real valued x(n) i.e $x_I(n)=0$

This property states that if x(n) is real then $X(N-k) = X^*(k)=X(-k)$

B) Real and even sequence x(n) i.e $x_I(n)=0$ & $X_I(K)=0$

This property states that if the sequence is real and even x(n)= x(N-n) then DFT becomes

$$X(k) = \sum_{n=0}^{N-1} x(n) \cos(2\prod kn/N)$$

C) Real and odd sequence x(n) i.e $x_I(n)=0$ & $X_R(K)=0$

This property states that if the sequence is real and odd x(n)=-x(N-n) then DFT becomes

$$X(k) = -j \sum_{n=0}^{N-1} x(n) \sin(2\prod kn/N)$$

D) Pure Imaginary x(n) i.e $x_R(n)=0$

This property states that if the sequence is purely imaginary x(n)=j $X_I(n)$ then DFT becomes

$$X_R(k) = \sum_{n=0}^{N-1} x_I(n) \sin(2\prod kn/N)$$

$$X_I(k) = \sum_{n=0}^{N-1} x_I(n) \cos(2\prod kn/N)$$

5. Circular Convolution

The Circular Convolution property states that if

$$x1(n) \xleftrightarrow[N]{DFT} X1(k) \text{ And}$$

$$x2(n) \xleftrightarrow[N]{DFT} X2(k) \text{ Then}$$

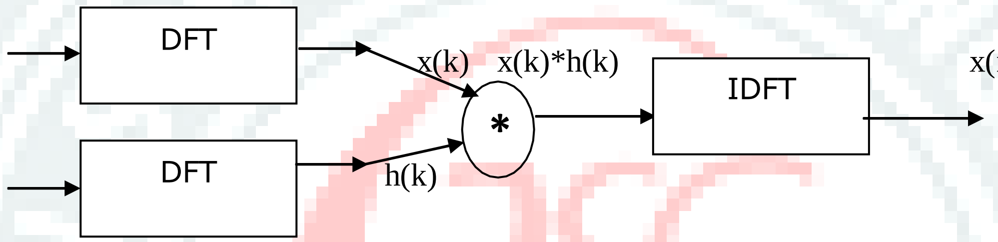
$$\text{Then } x1(n) \quad x2(n) \xleftrightarrow[N]{DFT} x1(k)\, x2(k)$$

It means that circular convolution of x1(n) & x2(n) is equal to multiplication of their DFT's.

Thus circular convolution of two periodic discrete signal with period N is given by

$$y(m) = \sum_{n=0}^{N-1} x1(n)\, x2(m-n)_N \qquad \text{.......(4)}$$

Multiplication of two sequences in time domain is called as Linear convolution while Multiplication of two sequences in frequency domain is called as circular convolution. Results of both are totally different but are related with each other.



There are two different methods are used to calculate circular convolution
1) Graphical representation form
2) Matrix approach

DIFFERENCE BETWEEN LINEAR CONVOLUTION & CIRCULAR CONVOLUTION

S. No	Linear Convolution	Circular Convolution
1	In case of convolution two signal sequences input signal x(n) and impulse response h(n) given by the same system, output y(n) is calculated	Multiplication of two DFT's is called as circular convolution.
2	Multiplication of two sequences in time domain is called as Linear convolution	Multiplication of two sequences in frequency domain is called as circular convolution.
3	Linear Convolution is given by the equation y(n) = x(n) * h(n) & calculated as $y(n) = \sum_{k=-\infty}^{\infty} x(k)\, h(n-k)$	Circular Convolution is calculated as $y(m) = \sum_{n=0}^{N-1} x1(n)\, x2(m-n)_N$
4	Linear Convolution of two signals returns N-1 elements where N is sum of elements in both sequences.	Circular convolution returns same number of elements that of two signals.

Tutorial problems:

Q) The two sequences x1(n)={2,1,2,1} & x2(n)={1,2,3,4}. Find out the sequence x3(m) which is equal to circular convolution of two sequences. Ans: X3(m)={14,16,14,16}

Q) x1(n)={1,1,1,1,-1,-1,-1,-1} & x2(n)={0,1,2,3,4,3,2,1}. Find out the sequence x3(m) which is equal to circular convolution of two sequences. Ans: X3(m)={-4,-8,-8,-4,4,8,8,4}

Q) Perform Linear Convolution of x(n)={1,2} & h(n)={2,1} using DFT & IDFT.

Q) Perform Linear Convolution of x(n)={1,2,2,1} & h(n)={1,2,3} using 8 Pt DFT & IDFT.

6. Multiplication

The Multiplication property states that if

$$X1(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x1(k) \text{ and}$$

$$X2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x2(k) \text{ then}$$

Then $x1(n)\ x2(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} 1/N\ x1(k)\ \ x2(k)$

It means that multiplication of two sequences in time domain results in circular convolution of their DFT's in frequency domain.

7. Time reversal of a sequence

The Time reversal property states that if

$$X(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x(k) \text{ and}$$

Then $x((-n))_N = x(N-n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x((-k))_N = x(N-k)$

It means that the sequence is circularly folded its DFT is also circularly folded.

8. Circular Time shift

The Circular Time shift states that if

$$X(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x(k) \text{ And}$$

Then $x((n-l))_N \overset{DFT}{\underset{N}{\longleftrightarrow}} x(k)\ e^{-j2\prod k l / N}$

Thus shifting the sequence circularly by 'l' samples is equivalent to multiplying its DFT by $e^{-j2\prod k l / N}$

9. Circular frequency shift

The Circular frequency shift states that if

$$X(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x(k) \quad \text{And}$$

$$\text{Then} \quad x(n)\, e^{j2\Pi l n / N} \overset{DFT}{\underset{N}{\longleftrightarrow}} x((n-l))_N$$

Thus shifting the frequency components of DFT circularly is equivalent to multiplying its time domain sequence by $e^{-j2\Pi k l / N}$

10. Complex conjugate property

The Complex conjugate property states that if

$$X(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x(k) \quad \text{then}$$

$$x^*(n) \overset{DFT}{\underset{N}{\longleftrightarrow}} x^*((-k))_N = x^*(N-k) \quad \text{And}$$

$$x^*((-n))_N = x^*(N-k) \overset{DFT}{\underset{N}{\longleftrightarrow}} x^*(k)$$

11. Circular Correlation

The Complex correlation property states

$$r_{xy}(l) \overset{DFT}{\underset{N}{\longleftrightarrow}} R_{xy}(k)= x(k)\, Y^*(k)$$

Here $r_{xy}(l)$ is circular cross correlation which is given as

$$r_{xy}(l) = \sum_{n=0}^{N-1} x(n)\, y^*((n-l))_N$$

This means multiplication of DFT of one sequence and conjugate DFT of another sequence is equivalent to circular cross-correlation of these sequences in time domain.

12. Parseval's Theorem

The Parseval's theorem states

$$\sum_{n=0}^{N-1} X(n)\, y^*(n) = 1/N \sum_{n=0} x(k)\, y^*(k)$$

This equation give energy of finite duration sequence in terms of its frequency components.

APPLICATION OF DFT

1. DFT FOR LINEAR FILTERING

Consider that input sequence x(n) of Length L & impulse response of same system is h(n) having M samples. Thus y(n) output of the system contains N samples where N=L+M-1. If DFT of y(n) also contains N samples then only it uniquely represents y(n) in time domain. Multiplication of two DFT's is equivalent to circular convolution of corresponding time domain sequences. But the length of x(n) & h(n) is less than N. Hence these sequences are appended with zeros to make their length N called as ―Zero padding‖. The N point circular convolution and linear convolution provide the same sequence. Thus linear convolution can be obtained by circular convolution. Thus linear filtering is provided by DFT.

When the input data sequence is long then it requires large time to get the output sequence. Hence other techniques are used to filter long data sequences. Instead of finding the output of complete input sequence it is broken into small length sequences. The output due to these small length sequences are computed fast. The outputs due to these small length sequences are fitted one after another to get the final output response.

UNIT- 5

Efficient Computation of the DFT: FFT algorithms

The *N*-point DFT of a sequence *x*(*n*) converts the time domain *N*-point sequence *x*(*n*) to a frequency domain *N*-point sequence *X*(*k*). The direct computation of an *N*-point DFT requires *N* x *N* complex multiplications and *N*(*N* – 1) complex additions. Many methods were developed for reducing the number of calculations involved. The most popular of these is the Fast Fourier Transform (FFT), a method developed by Cooley and Turkey. The FFT may be defined as an algorithm (or a method) for computing the DFT efficiently (with reduced number of calculations). The computational efficiency is achieved by adopting a divide and conquer approach. This approach is based on the decomposition of an *N*-point DFT into successively smaller DFTs and then combining them to give the total transform. Based on this basic approach, a family of computational algorithms were developed and they are collectively known as FFT algorithms. Basically there are two FFT algorithms; Decimation- in-time (DIT) FFT algorithm and Decimation-in-frequency (DIF) FFT algorithm. In this chapter, we discuss DIT FFT and DIF FFT algorithms and the computation of DFT by these methods.

FAST FOURIER TRANSFORM

The DFT of a sequence $x(n)$ of length N is expressed by a complex-valued sequence $X(k)$ as

$$X(K)=\sum_{n=0}^{N-1} x(n)e^{-j2\pi nk/N}, K=0,1,2,....N-1 \text{ where}$$

Let W_N be the complex valued phase factor, which is an N^{th} root of unity given by

$$W_N = e^{-j2\pi nk/N}$$

Thus,

$X(k)$ becomes,

$$X(K)=\sum_{n=0}^{N-1} x(n)W_N^{nk}, K=0,1,2,....N-1$$

Similarly, IDFT is written as

$$x(n)=\sum_{n=0}^{N-1} X(K)W_N^{-nk}, n=0,1,2,....N-1$$

From the above equations for $X(k)$ and $x(n)$, it is clear that for each value of k, the direct computation of $X(k)$ involves N complex multiplications ($4N$ real multiplications) and $N-1$ complex additions ($4N-2$ real additions). Therefore, to compute all N values of DFT, N^2 complex multiplications and $N(N-1)$ complex additions are required. In fact the DFT and IDFT involve the same type of computations.

If $x(n)$ is a complex-valued sequence, then the N-point DFT given in equation for $X(k)$ can be expressed as

$$X(k) = X_R(k) + jX_I(k)$$

Direct Computation of DFT:

The direct computation of the DFT needs $2N^2$ evaluations of trigonometric functions, $4N^2$ real multiplications and $4N(N-1)$ real additions. Also this is primarily inefficient as it cannot exploit the symmetry and periodicity properties of the phase factor W_N, which are

Symmetry property $W_N^{k+N/2} = -W_N^K$

Periodicity property $W_N^{k+N} = W_N^K$

FFT algorithm exploits the two symmetry properties and so is an efficient algorithm for DFT computation.

By adopting a divide and conquer approach, a computationally efficient algorithm can be developed. This approach depends on the decomposition of an N-point DFT into successively smaller size DFTs. An N-point sequence, if N can be expressed as $N = r_1r_2r_3, \ldots, r_m$. where $r_1 = r_2 = r_3 = \ldots = r_m$, then $N = r^m$, can be decimated into r-point sequences. For each r-point sequence, r-point DFT can be computed. Hence the DFT is of size r. The number r is called the radix of the FFT algorithm and the number m indicates the number of stages in computation. From the results of r-point DFT, the r^2-point DFTs are computed. From the results of r^2-point DFTs, the r^3-point DFTs are computed and so on, until we get r^m-point DFT. If $r = 2$, it is called radix-2 FFT.

Divide & Conquer approach to computation of the DFT

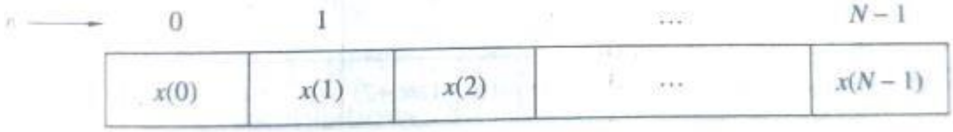
The development of computationally efficient algorithms for the DFT is made possible if we adopt a divide-and-conquer approach. This approach is based on the decomposition of an N-point DFT into successively smaller DFTs. This basic approach leads to a family of computationally efficient algorithms known collectively as FFT algorithms.

To illustrate the basic notions, let us consider the computation of an N-point DFT, where N can be factored as a product of two integers, that is,

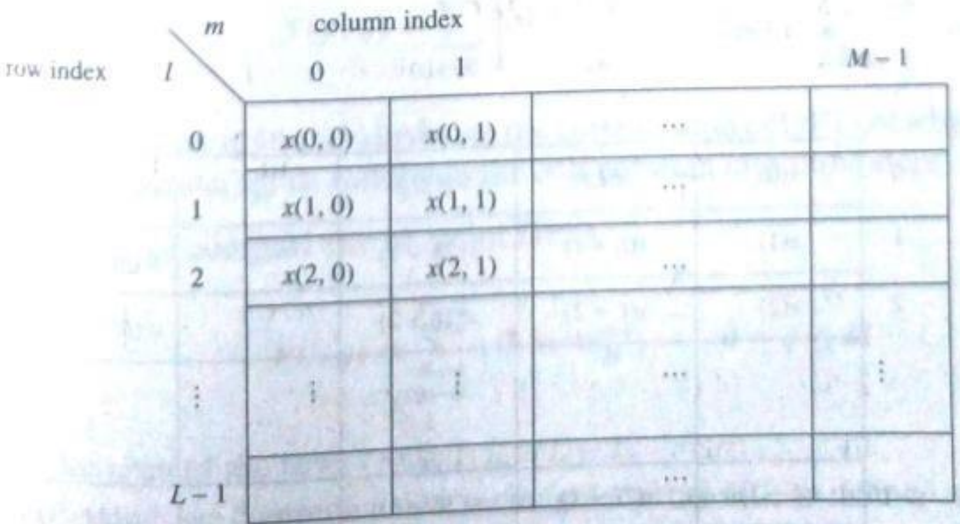
$$N = LM \tag{8.1.8}$$

The assumption that N is not a prime number is not restrictive, since we can pad any sequence with zeros to ensure a factorization of the form (8.1.8).

Now the sequence $x(n)$, $0 \le n \le N-1$, can be stored either in a one-dimensional array indexed by n or as a two-dimensional array indexed by l and m, where $0 \le l \le L-1$ and $0 \le m \le M-1$ as illustrated in Fig. 8.1.1. Note that l is the row index and m is the column index. Thus, the sequence $x(n)$ can be stored in a rectangular



(a)



(b)

DECIMATION IN TIME (DIT) RADIX-2 FFT

In Decimation in time (DIT) algorithm, the time domain sequence $x(n)$ is decimated and smaller point DFTs are computed and they are combined to get the result of N-point DFT.

In general, we can say that, in DIT algorithm the N-point DFT can be realized from two numbers of $N/2$-point DFTs, the $N/2$-point DFT can be realized from two numbers of $N/4$-point DFTs, and so on.

In DIT radix-2 FFT, the N-point time domain sequence is decimated into 2-point sequences and the 2-point DFT for each decimated sequence is computed. From the results of 2-point DFTs, the 4-point DFTs, from the results of 4-point DFTs, the 8-point DFTs and so on are computed until we get N-point DFT.

For performing radix-2 FFT, the value of r should be such that, $N = 2^m$. Here, the decimation can be performed m times, where $m = \log_2 N$. In direct computation of N-point DFT, the total number of complex additions are $N(N-1)$ and the total number of complex multiplications are N^2. In radix-2 FFT, the total number of complex additions are reduced to $N \log_2 N$ and the total number of complex multiplications are reduced to $(N/2) \log_2 N$.

Let $x(n)$ be an N-sample sequence, where N is a power of 2. Decimate or break this sequence into two sequences $f_1(n)$ and $f_2(n)$ of length $N/2$, one composed of the even indexed values of $x(n)$ and the other of odd indexed values of $x(n)$.

Given sequence $x(n): x(0), x(1), x(2), \ldots\ldots x(\frac{N}{2}-1) \ldots . x(N-1)$

Even indexed sequence $f_1(n) = x(2n): x(0), x(2), x(4), \ldots\ldots . x(N-2)$

Odd indexed sequence $f_2(n) = x(2n+1): x(1), x(3), x(5), \ldots\ldots . x(N-1)$

We know that the transform X(k) of the N-point sequence x(n) is given by

$$X(K) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, K = 0,1,2,\ldots . N-1$$

Breaking the sum into two parts, one for the even and one for the odd indexed values, gives

$$X(K) = \sum_{n=0}^{N/2-1} x(n) W_N^{nk} + \sum_{n=N/2}^{N-1} x(n) W_N^{nk}, K = 0,1,2,\ldots . N-1.$$

$$X(K) = \sum_{n=even}^{N/2-1} x(n) W_N^{nk} + W_N^{nk} \sum_{n=odd}^{N-1} x(n) W_N^{nk}$$

When n is replaced by 2n, the even numbered samples are selected and when n is replaced by 2n + 1, the odd numbered samples are selected. Hence,

$$X(K) = \sum_{n=0}^{N/2-1} x(2n) W_N^{2nk} + \sum_{n=0}^{N/2-1} x(2n+1) W_N^{(2n+1)k}$$

Rearranging each part of X(k) into (N/2)-point transforms using

$$W_N^{2nk} = (W_N^2)^{nk} = \left[e^{-j\frac{2\pi}{N}} \right]^{2nk} = W_{N/2}^{nk} \text{ and } W_N^{(2n+1)k} = (W_N^k) W_{N/2}^{nk}$$

We can write

$$X(K) = \sum_{n=0}^{N/2-1} f_1(n) W_{N/2}^{nk} + W_N^{k} \sum_{n=0}^{N/2-1} f_2(n) W_{N/2}^{nk}$$

By definition of DFT, the N/2-point DFT of f1(n) and f2(n) is given by

$$F_1(K) = \sum_{n=0}^{N/2-1} f_1(n) W_{N/2}^{nk} \,\&\, F_2(K) + \sum_{n=0}^{N/2-1} f_2(n) W_{N/2}^{nk}$$

$$X(k) = F_1(K) + W_N^{k} F_2(K), \ldots k = 0,1,2,3,\ldots . N-1$$

The implementation of this equation for X(k) is shown in the following Figure . This first step in the decomposition breaks the N-point transform into two (N/2)-point transforms and the k WN provides the N-point combining algebra. The DFT of a sequence is periodic with period given by the number of points of DFT. Hence, F1(k) and F2(k) will be periodic with period N/2.

$$F_1(k+N/2)=F_1(K), \& F_2(k+N/2)=F_2(K)$$
$$F_1(k+N/2)=F_1(K), \& F_2(k+N/2)=F_2(K)$$

In addition, the phase factor $W_N^{(k+N/2)} = -(W_N^k)$

Therefore, for $k \geq N/2$, X(k) is given by

$$X(K)=F_1(k-N/2)-W_N{}^kF_2(K-N/2)$$

The implementation using the periodicity property is also shown in following Figure

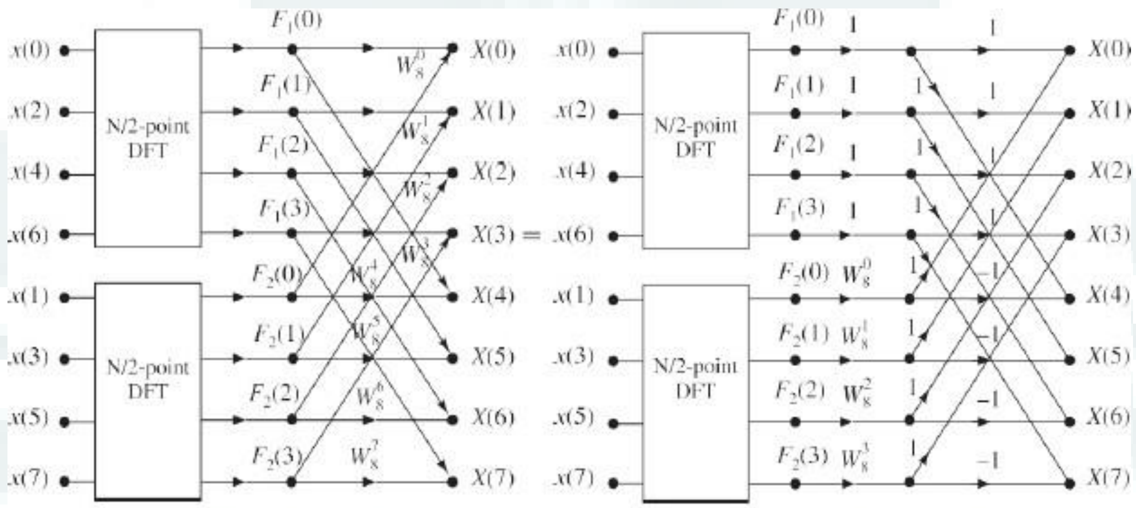


Figure 2.1 Illustration of flow graph of the first stage DIT FFT algorithm for $N = 8$.

EXAMPLE 1 Draw the butterfly line diagram for 8-point FFT calculation and briefly explain. Use decimation-in-time algorithm.

Solution: The butterfly line diagram for 8-point DIT FFT algorithm is shown in following Figure

<u>Solution:</u> For 8-point DIT FFT

1. The input sequence $x(n) = \{x(0), x(1), x(2), x(3), x(4), x(5), x(6), x(7)\}$,
2. bit reversed order, of input as i.e. as $x_r(n) = \{x(0), x(4), x(2), x(6), x(1), x(5), x(3), x(7)\}$. Since $N = 2^m = 2^3$, the 8-point DFT computation
3. Radix-2 FFT involves 3 stages of computation, each stage involving 4 butterflies. The output $X(k)$ will be in normal order.
4. In the first stage, four 2-point DFTs are computed. In the second stage they are combined into two 4-point DFTs. In the third stage, the two 4-point DFTs are combined into one 8-point DFT.
5. The 8-point FFT calculation requires 8 $\log_2 8$ = 24 complex additions and (8/2) $\log_2 8$ = 12 complex multiplications.

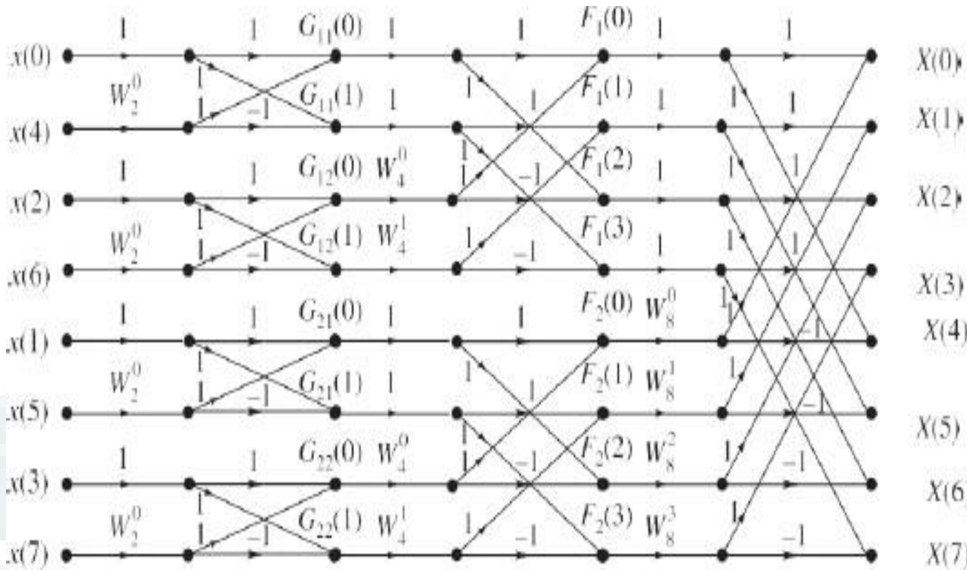


Figure : Butterfly Fine diagram for 8–point DIT FFT algorithm for $N = 8$.