Part - B

Experiments based on MATLAB

Expt. No: 1

Generation of Waveforms (Continuous and Discrete)

Aim: To plot the following wave forms (continuous and discrete)

1. Unit Impulse signal
2. Unit Step Signal
3. Unit Ramp Signal
4. Sine signal
5. Cosine signal

Theory:

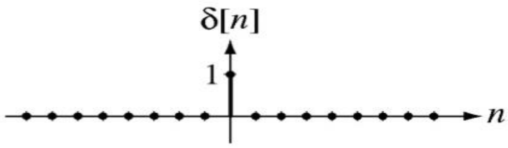
In signal processing experiments, we need the help of some fundamental signals such as sine wave, cosine wave, square wave, ramp wave, unit step, unit impulse etc.

A digital signal can be either a deterministic signal that can be predicted with certainty, or a random signal that is unpredictable. Due to ease in signal generation and need for predictability, deterministic signal can be used for system simulation studies. Standard forms of some deterministic signals that are frequently used in DSP are discussed below:

Discrete Time Signals

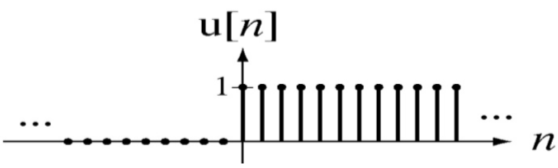
1. **The *unit sample sequence is*** denoted as ***δ(n)*** and is defined as

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



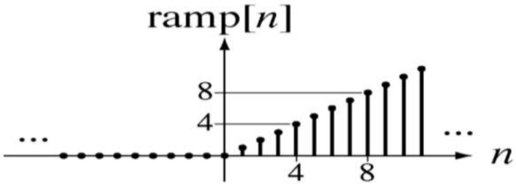
2. The ***unit step signal*** is denoted as u(n) and is defined as

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

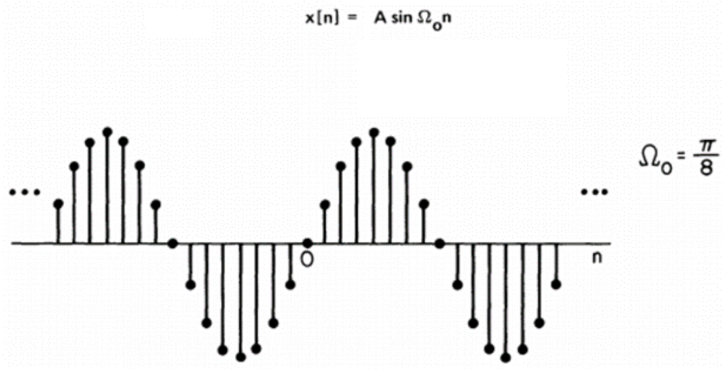


3. The ***unit ramp signal*** is denoted as $u_r(in)$ and is defined as

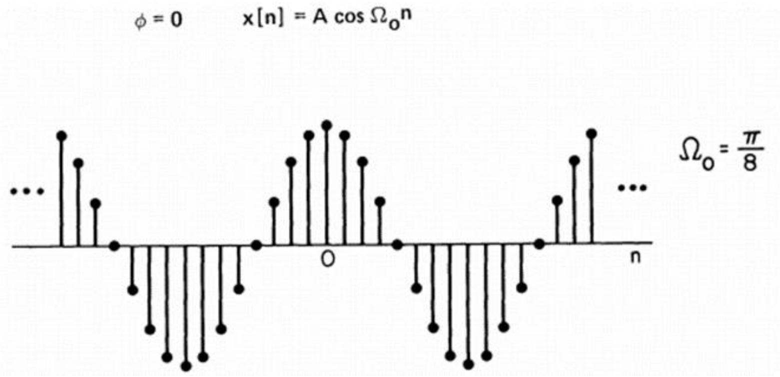
$$\text{ramp}[n] = \begin{cases} n, & n \geq 0 \\ 0, & n < 0 \end{cases} = \sum_{m=-\infty}^{n} u[m-1]$$



4. *Sine Signal*



5. *Cosine signal*



Matlab code:

OUTPUT

Discrete time signals

Continuous time signals

RESULT

The MATLAB program to generate continuous and discrete waveforms were executed and obtained output waveforms.

Expt. No: 2

TIME AND FREQUENCY RESPONSE OF LTI SYSTEMS

Aim: Obtain time and frequency response of LTI systems

i. First order systems

a). $y(n) = x(n) - x(n-1)$

b). $y(n) = 0.2x(n) + 0.2x(n-1) + 0.6y(n-1)$

ii. Second order systems

a). $y(n) = 0.5x(n) + x(n-1) + 0.5x(n-2)$

b). $y(n) = x(n) + 0.9x(n-2) - 0.4y(n-2)$

Theory:

Linear Time Systems (LTI Systems) are a class of systems used in DSP that are both linear and time invariant. Linear systems are systems whose output for a linear combination of inputs are the same as a linear combination of individual responses to those inputs. The time invariant systems are systems where the output does not depend on when an input was applied. These properties make LTI systems easy to represent and understand graphically.

The general form an N^{th} order Linear Time Invariant Discrete Time System (LTI-DT) system is given by the difference equation,

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

Where $\{a_k\}$ and $\{b_k\}$ are constants

Taking Z-Transform on both sides

$$Y(z) = -\sum_{k=1}^{N} a_k\, Y(z) z^{-k} + \sum_{k=0}^{M} b_k\, X(z) z^{-k}$$

Transfer Function : $H(z)$

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

Frequency Response: $H(e^{j\omega})$

$$H(e^{j\omega}) = \frac{Y(e^{j\omega})}{X(e^{j\omega})} = \frac{\sum_{k=0}^{M} b_k\, e^{-j\omega k}}{1 + \sum_{k=1}^{N} a_k\, e^{-j\omega k}}$$

Magnitude response : $|H(e^{j\omega})|$

Phase response : $\angle H(e^{j\omega})$

Matlab code:

Output

RESULT

The MATLAB program to obtain the time and frequency response of an LTI system is executed and output is obtained.

Expt. No: 3

LINEAR CONVOLUTION & CIRCULAR CONVOLUTION

Aim: To perform the following signal operations

1. Linear Convolution
2. Circular Convolution
3. Linear Convolution using Circular Convolution

Theory:

Linear Convolution:

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

y(n) = output response

h(n) = impulse response of system

x(n) = input signal

For Causal finite sequences linear convolution become,

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

The convolution performed between the discrete time signals is usually referred to as convolution sum. If x[n] is an M-point sequence and h[n] is an N-point sequence, the y[n] will be (M+N-1) – point sequence.

Circular Convolution:

A convolution operation that contains a circular shift is called circular convolution. Circular Convolution of two sequences h(k) and y(k), each of length N is given by,

$$y(n) = x_1(k) \odot x_2(k) \quad = \sum_{k=0}^{N-1} x(k)h\big((n-k)\big)_N \quad 0 \le n \le N-1$$

If the length of sequence is not equal, zero padding is done to get the maximum length among the two. The resulting convolved signal would be zero outside the range n= 0,1, ..., N-1.

Linear Convolution using Circular convolution:

Increase the length of the sequences x(n) and h(n) to L+M-1 points (using zero padding) and circularly convolve the resulting sequence. That is the linear convolution of x[n] and h[n] can be computed using N-point circular convolution given in equation of linear convolution.

- L = M + N – 1

- N – 1 zeros are appended to x[n] to make its length equal to L = M+N-1
- M – 1 zeros are appended to h[n] to make its length equal to L = M+N-1

MATLAB code:

Output:

RESULT

a) The MATLAB program to find the linear convolution of two input sequences is executed and output is verified using inbuilt function.

b) The MATLAB program to find the circular convolution of two input sequences is executed and output is verified using inbuilt function.

c) The MATLAB program to find the linear convolution using circular convolution of two input sequences is executed and output is verified using inbuilt function.

Expt. No: 4

DFT AND IDFT OF A SEQUENCE

Aim:

To find out the DFT and IDFT of a sequence

Theory:

Discrete Fourier Transform is the transformation used to represent the finite duration frequencies. DFT of a discrete sequence x(n) is obtained by performing sampling operations in both time domain and frequency domain. It is the frequency domain representation of a discrete digital signal.

DFT:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} = \sum_{n=0}^{N-1} x(n) \left(e^{-j2\pi/N}\right)^{nk}$$

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

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

Inverse DFT:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi nk/N}$$

MATLAB code:

Output:

RESULT

The MATLAB program to find DFT and IDFT of input sequence is executed and output is obtained.

Expt. No: 5

Linear convolution using DFT

AIM:

Linear convolution of two given sequences using DFT and IDFT.

OBJECTIVE:

To find the Linear convolution of a given sequence using the inbuilt MATLAB functions "FFT and IFFT" for DFT and IDFT and verify the result using the function "CONV"

THEORY:

Convolution is an integral concatenation of two signals. It has many applications in numerous areas of signal processing. The most popular application is the determination of the output signal of a linear time-invariant system by convolving the input signal with the impulse response of the system. Note that convolving two signals is equivalent to multiplying
the Fourier Transform of the two signals.

MATLAB code:

OUTPUT:

RESULT:

Linear convolution of two given sequences found using DFT and the results are verified.

Expt. No: 6

FFT and IFFT for the given input sequence

AIM

Write a MATLAB program to find the FFT and IFFT of the given input.

THEORY

The most common tools used to perform Fourier analysis and synthesis are called the fast Fourier transform (FFT) and the inverse fast Fourier transform (IFFT). The FFT and IFFT are optimized (very fast) computer-based algorithms that perform a generalized mathematical process called the *discrete Fourier transform* (DFT). The DFT is the actual mathematical transformation that the data go through when converted from one domain to another (time to frequency). Basically, the DFT is just a slow version of the FFT.

MATLAB code:

OUTPUT:

RESULT:

The MATLAB program to find the DCT and IDCT for the given input was executed and output obtained correctly.

Expt. No: 7

FIR AND IIR FILTER DESIGN USING FILTER DESIGN TOOLBOX

AIM

i. A digital low-pass filter is required to meet the following specifications:

Passband ripple : $\leq$ 1 dB

Passband edge : 4 kHz

Stopband attenuation: $\geq$ 40 dB

Stopband edge : 6kHz

Sampling rate : 24 kHz

Design a) Digital Butterworth, b) Digital Chebyshev filter

ii. A digital band-pass filter is required to meet the following specifications:

$f_{p1} = 20Hz$ $\quad$ $f_{s1} = 10Hz$

$f_{p2} = 30Hz$ $\quad$ $f_{s2} = 40Hz$

$\alpha_p = 0.5dB$

$\alpha_s = 30dB$ $\quad$ $F_S = 100Hz$

Design a digital FIR filter using a) Hamming Window b) Using Kaiser Window

THEORY

Filter Design and Analysis using FDA Tool of MATLAB.

The Filter Design and Analysis Tool (FDA Tool) is a powerful user interface for designing and analysing filters quickly. FDA Tool enables you to design digital FIR or IIR filters by setting filter specifications, by importing filters from your MATLAB workspace, or by adding, moving or deleting poles and zeros. FDA Tool also provides tools for analysing filters, such as magnitude and phase response and pole-zero plots.

To use FDATOOL in MATLAB, Type >> fdatool in command window, FDAtool will be opened. There you can select FIR or IIR filter, order of filter and cutoff frequency of a filter (either HPF, LPF or BPF).

OUTPUT:

Butterworth LPF

Order = 10

Magnitude and Phase response of Butterworth LPF

Chebyshev LPF

Order = 6

Magnitude and Phase response of Chebyshev LPF

FIR filter using Hamming window

Order = 9

Magnitude and Phase response

RESULT:

The MATLAB program to find the DCT and IDCT for the given input was executed and output obtained correctly.