Chapter 3

# Filtering:

# DFT In Linear Filtering

# Introduction to Filters

- Let us first take a glance at ideal response and specifications of filters i.e.
- Basic terminology and definitions: filtering, filter, analogue filtering, digital/discrete-time filtering.
- Basic parameter specification required for filter design.

# Definitions of Basic Terms

**Filtering:** Process of extraction of desired signal from noise.

**Filter:** System performing filtering.

**Analogue filtering:** filtering performed on continuous-time signals and yields continuous-time signals.

**Digital/discrete-time filtering:** filtering performed on digital/discrete-time signals and yields digital/ discrete-time signals.

# Examples of filtering applications

## Noise suppression

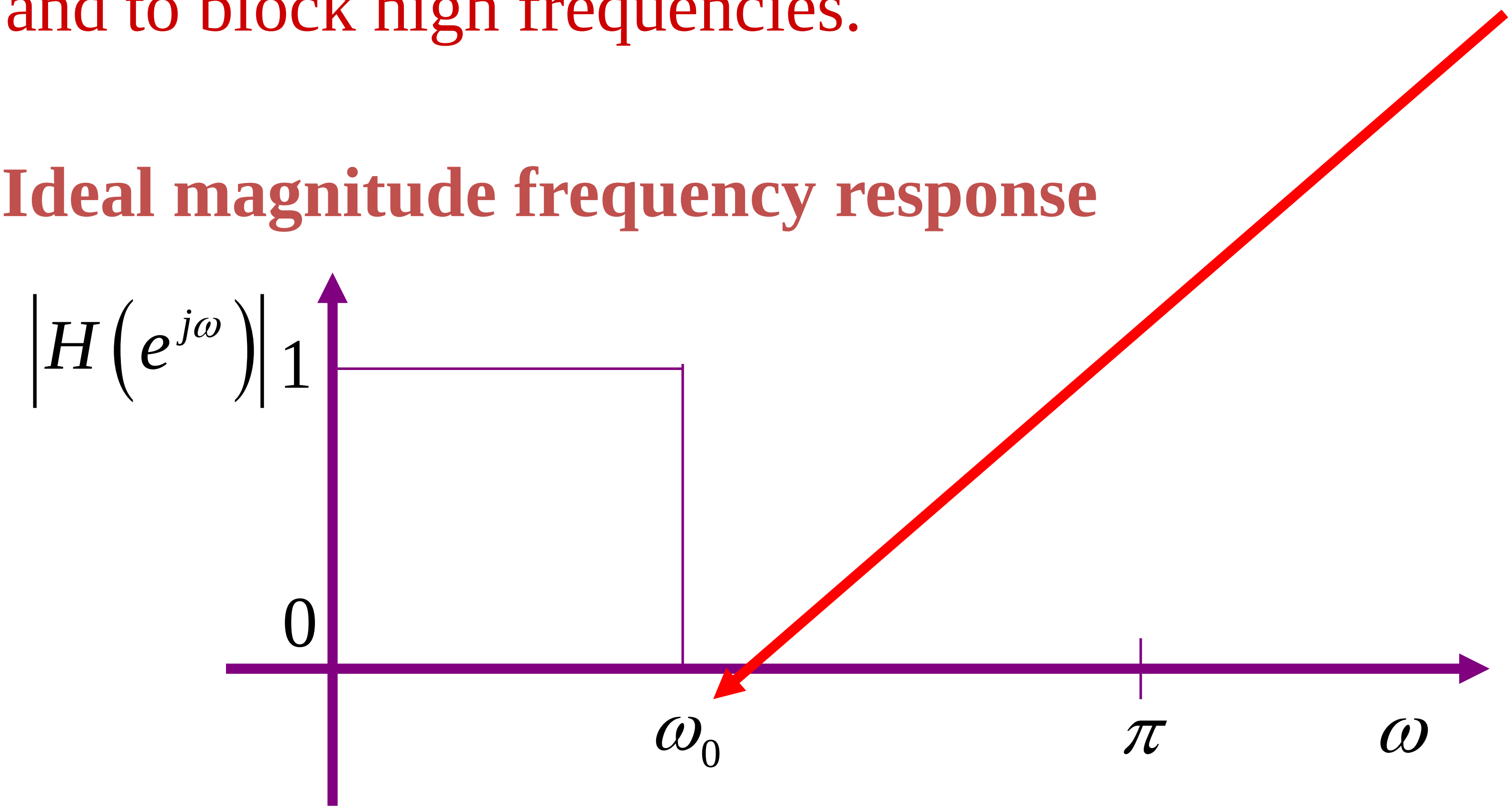
- Received radio signals.
- Signals received by imaging sensors, such as television cameras or infrared imaging devices.
- Electrical signals measured from the human body (such as brain, heart or neurological signals).

# Filter Specifications

## Ideal Filters

**Low-Pass Filters:** Low-pass filters are designed to pass low frequencies, from zero to a certain cut off frequency and to block high frequencies.

## Ideal magnitude frequency response



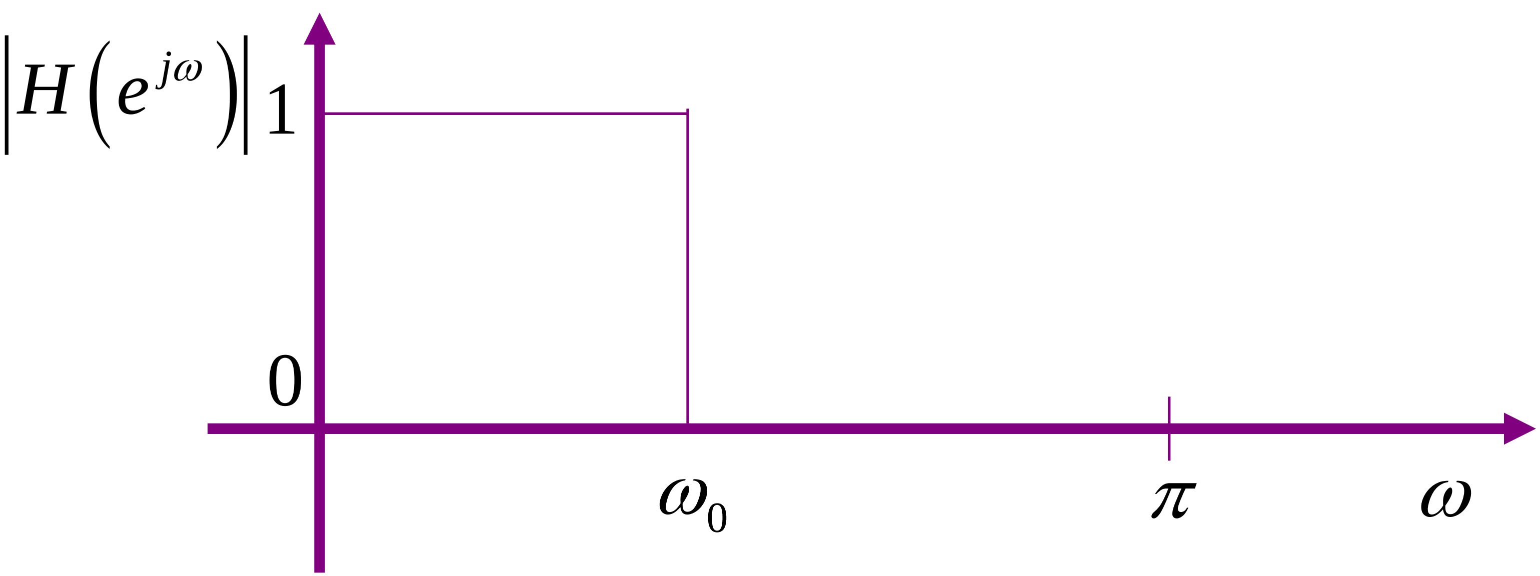
# Filter Specifications

## Ideal Filters

## Low-Pass Filters:

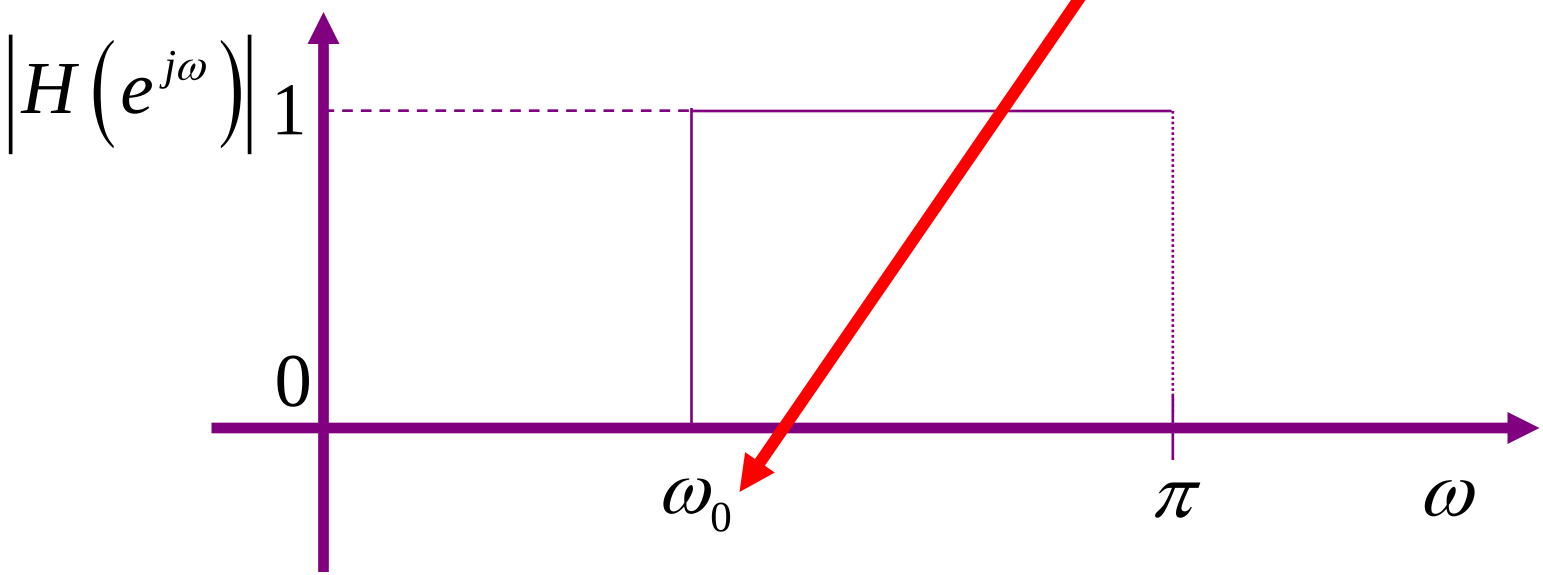
$$\left|H\left(e^{j\omega}\right)\right| = \begin{cases} 1 & for\ \ \omega \in < 0, \omega_0 > \ \ i.e.\ \ \omega \in pass-band \\ 0 & for\ \ \omega \in (\omega_0, \pi > \ \ i.e.\ \ \omega \in stop-band \end{cases}$$

## Ideal magnitude frequency response



**High-Pass Filters:** High-pass filters are designed to pass high frequencies, from a certain cut off frequency to $\pi$, and to block low frequencies.

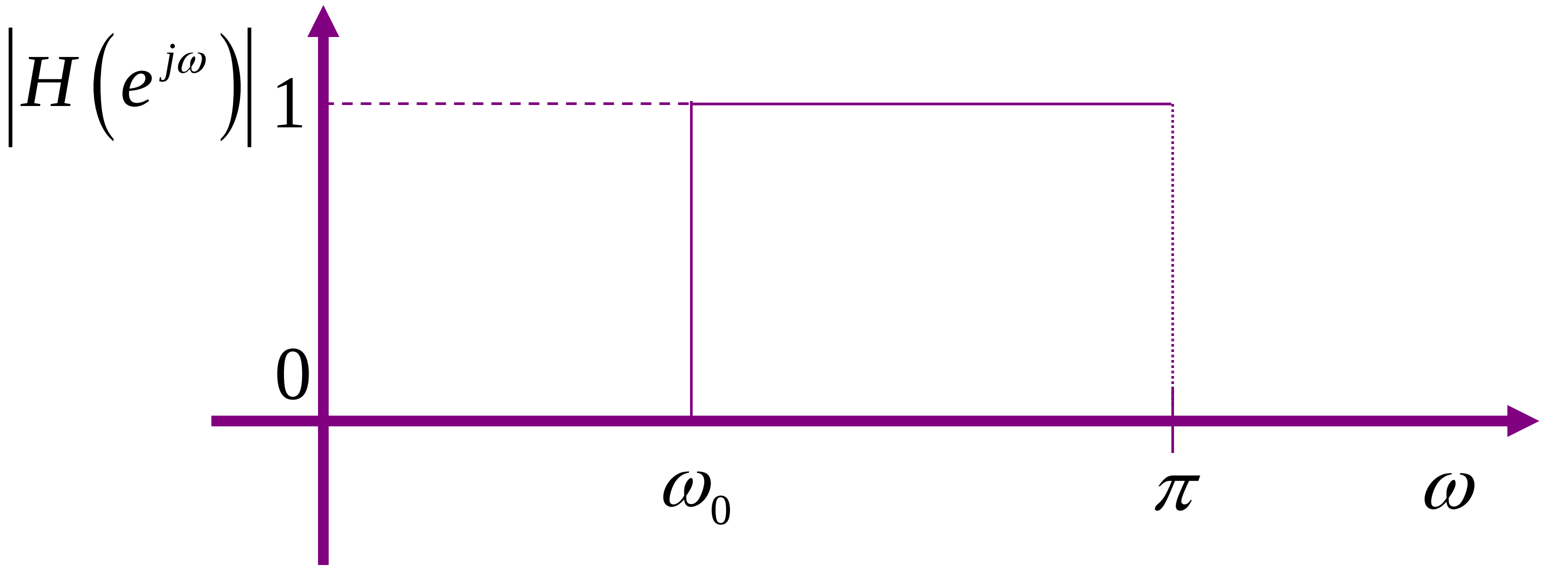
## Ideal magnitude frequency response



# High-Pass Filters:

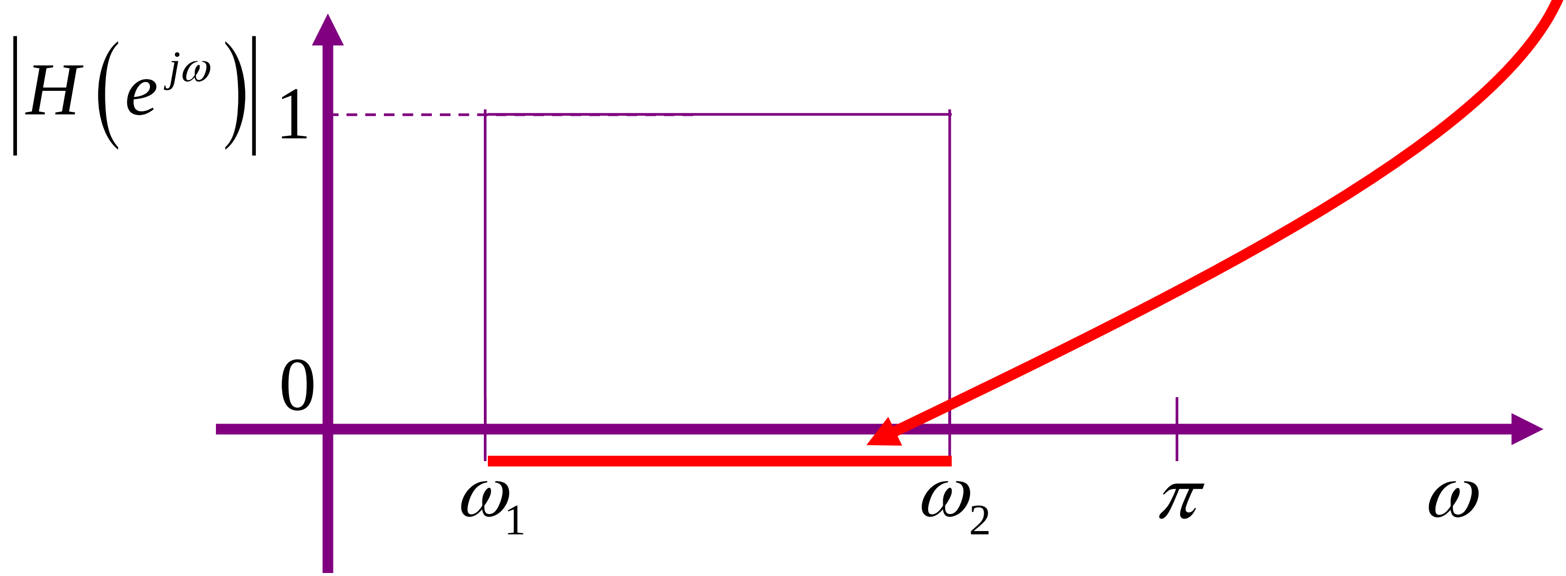
$$\left|H\left(e^{j\omega}\right)\right| = \begin{cases} 0 & for \;\; \omega \in <0, \omega_0) \quad i.e. \;\; \omega \in stop-band \\ 1 & for \;\; \omega \in <\omega_0, \pi> \quad i.e. \;\; \omega \in pass-band \end{cases}$$

## Ideal magnitude frequency response



**Band-Pass Filters:** Band-pass filters are designed to pass a certain frequency range, which does not include zero, and to block other frequencies.

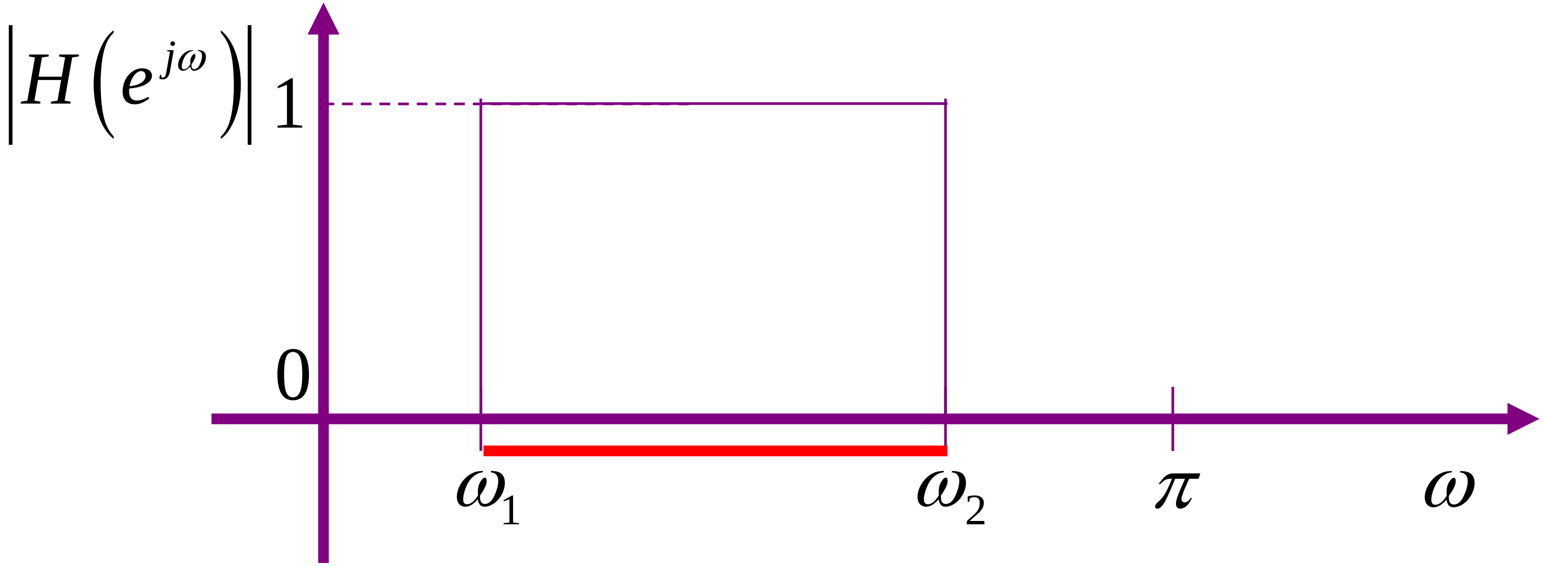
**Ideal magnitude frequency response**



# Band-Pass Filters:

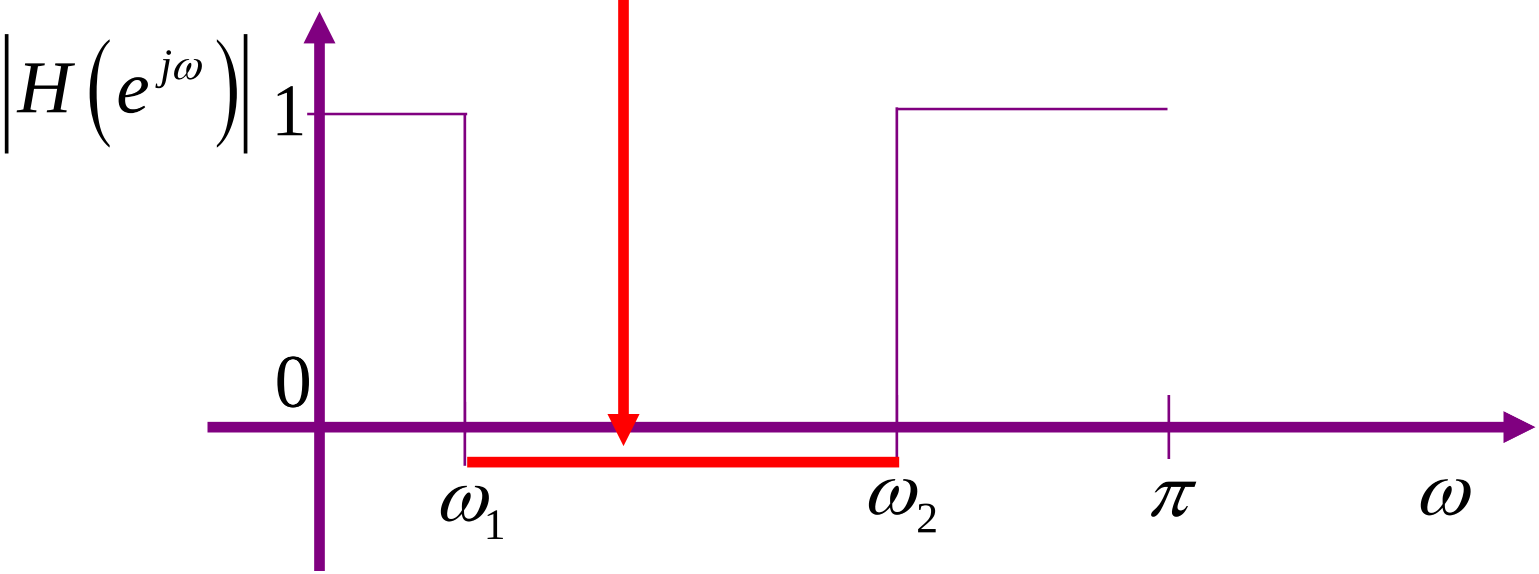
$$\left|H\left(e^{j\omega}\right)\right| = \begin{cases} 0 & for\ \ \omega \in <0, \omega_1) \cup (\omega_2, \pi)\ \ i.e.\ \omega \in stop-band \\ 1 & for\ \ \omega \in <\omega_1, \omega_2> \qquad i.e.\ \omega \in pass-band \end{cases}$$

## Ideal magnitude frequency response



**Band-Stop Filters:** Band-stop filters are designed to block a certain frequency range, which does not include zero, and to pass other frequencies.

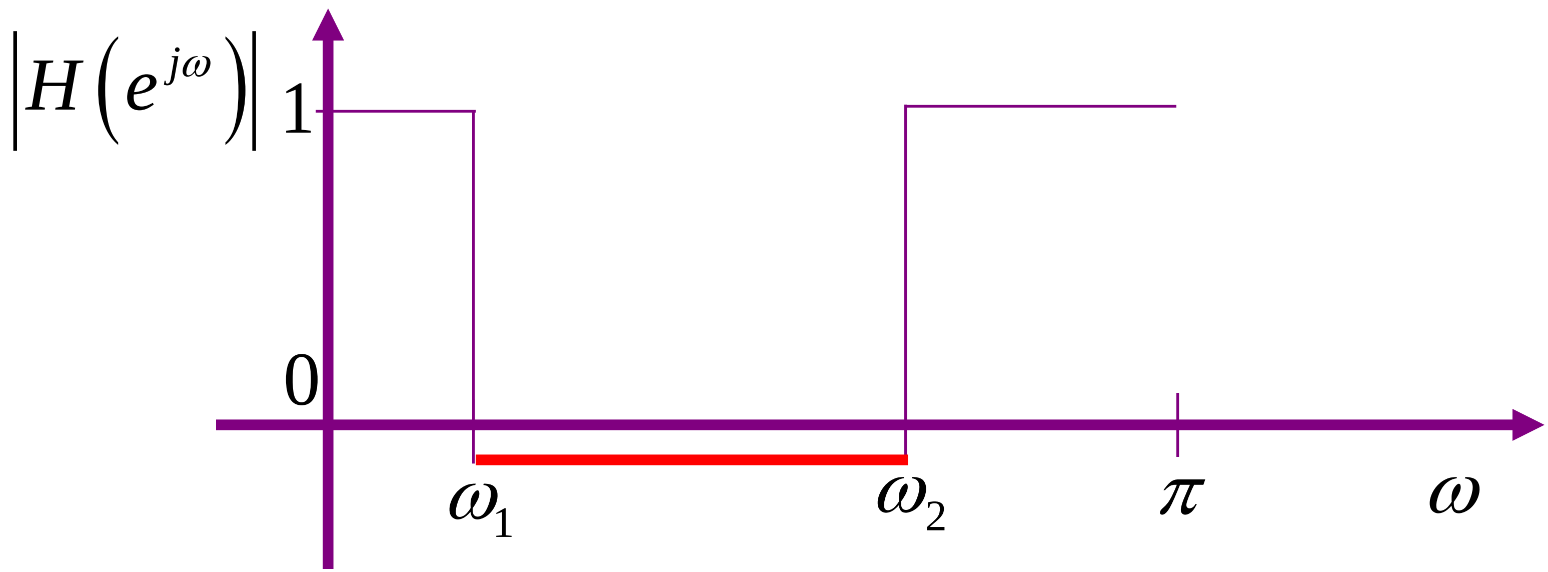
## Ideal magnitude frequency response



# Band-Stop Filters:

$$\left|H\left(e^{j\omega}\right)\right| = \begin{cases} 1 & for \;\; \omega \in <0, \omega_1> \cup <\omega_2, \pi) \;\; i.e. \; \omega \in pass-band \\ 0 & for \;\; \omega \in (\omega_1, \omega_2) \;\; i.e. \; \omega \in stop-band \end{cases}$$

## Ideal magnitude frequency response



**A. Comments on phase response:** The phase response of ideal filters is **linear**:

$$\phi(\omega) = -\omega t_0$$

**B. Comments on group delay function:** Group delay function of ideal filters is **constant**:

$$\tau(\omega) = -\frac{d\phi(\omega)}{d\omega} = -\frac{d}{d\omega}\left[-\omega t_0\right] = t_0 = const.$$

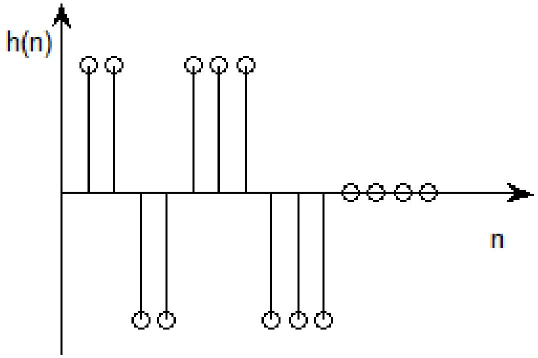
**C. Note:** It will be proved for linear phase FIR filters:

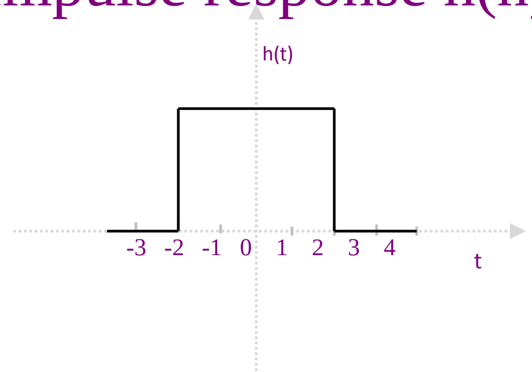
$$t_0 = \frac{M-1}{2}$$

# Classification of filters by a nature of transfer function

- Finite Impulse response (FIR):

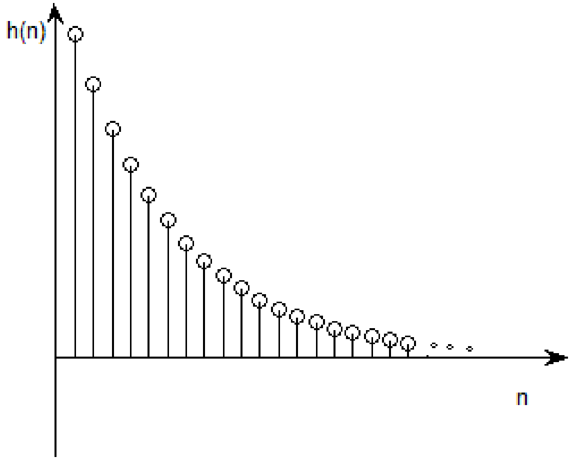
Time domain existence of Impulse response h(n)/h(t) is finite.

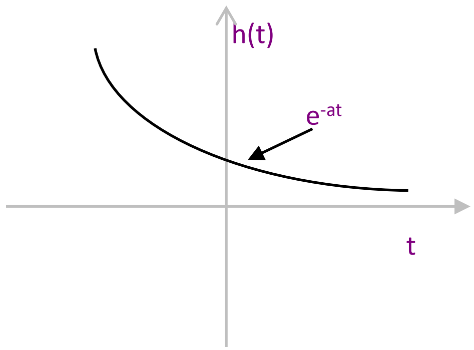



- Infinite Impulse response(IIR)

Time domain existence of Impulse response h(n)/h(t) is infinite.



# Time domain/Complex domain representation of transfer function

**Time domain** | **Complex domain** | **Frequency domain**

**$h(t)$** —Laplace transform→ **$H(S)$** —$s=\sigma+j\Omega$→ **$H(j\Omega)$**

**$h(n)$** —Z transform→ **$H(Z)$** —$z = re^{j\omega}$→ **$H(j\omega)$**

# Transfer function: CT Signal

## FIR

$$y(t)=\sum_{i=0}^{N} b_i \frac{d^i x(t)}{dt^i} \xleftrightarrow{Laplace-transform} Y(s)=\sum_{i=0}^{N} b_i s^i X(s)$$

$$\therefore H(s)=\sum_{i=0}^{N} b_i s^i$$

## IIR

$$y(t)=\sum_{i=0}^{N} b_i \frac{d^i x(t)}{dt^i}-\sum_{i=0}^{N} a_i \frac{d^i y(t)}{dt^i} \xleftrightarrow{L-transform} Y(s)=\sum_{i=0}^{N} b_i s^i X(s)-\sum_{i=0}^{M} a_i s^i Y(s)$$

$$\therefore H(s)=\frac{\sum_{i=0}^{N} b_i s^i}{\sum_{i=0}^{M} a_i s^i}$$

# Transfer function: DT Signal

## FIR

$$y(n)=\sum_{i=0}^{N} b_i x(n-i) \xleftrightarrow{ztransform} Y(z)=\sum_{i=0}^{N} b_i z^{-i} X(z)$$

$$\therefore H(Z)=\sum_{i=0}^{N} b_i z^{-i}$$

## IIR

$$y(n)=\sum_{i=0}^{N} b_i x(n-i)-\sum_{i=0}^{N} a_i y(n-i) \xleftrightarrow{ztransform} Y(z)=\sum_{i=0}^{N} b_i z^{-i} X(z)-\sum_{i=0}^{N} a_i z^{-i} Y(z)$$

$$\therefore H(Z)=\frac{\sum_{i=0}^{N} b_i z^{-i}}{\sum_{i=0}^{N} a_i z^{-i}}$$

# Use of DFT in Linear filtering

- Filtering the longer sequences in time domain is very complicated.
- So filtering of longer sequence can be divided in to filtering of smaller sequence and then combining together their responses for the overall response.
- It is easy to perform filtering in frequency domain compared to time domain.
- Fourier representation of DT finite duration signal is continuous and can not be processed with digital computers.
- When there is need to process these using computers data is required to be stored digitally and then processed processed.
- So DFT is obtained by discretising the DTFT by changing $\omega=2\pi k/N$, i.e. dividing the 0 to $2\pi$ into N equally spaced points.

# Use of DFT in Linear filtering

- Suppose X(k) and H(k) the DFT of I/P and impulse response respectively. Then computing Y(k)= X(k).H(k) and inverse transforming Y(k) to obtain the response (O/P) of the filter.
- Fast DFT computing methods are also available for reducing the number of computations.
- Sectioned filtering can be performed using sectioned convolutions of
  1. Overlap and add method
  2. Overlap and save method

# Sectioned convolution in time domain first

- Need: Performing convolution over a longer duration input signal is a tedious task.
- Alternative: Dividing I/P into smaller blocks and smaller blocks are convolved to generate the combined result finally.
- Types

1. Overlap and add method
2. Overlap and save method

# Overlap Add method

- Consider the length of h(n)= M
- The I/P sequence x(n) is partitioned into smaller blocks of size L, where N=L+M-1
- N = length of the convolution & is $2^K$.
- To each i/p data block of size L append M-1 zeros.
- Thus the data blocks may be represented as

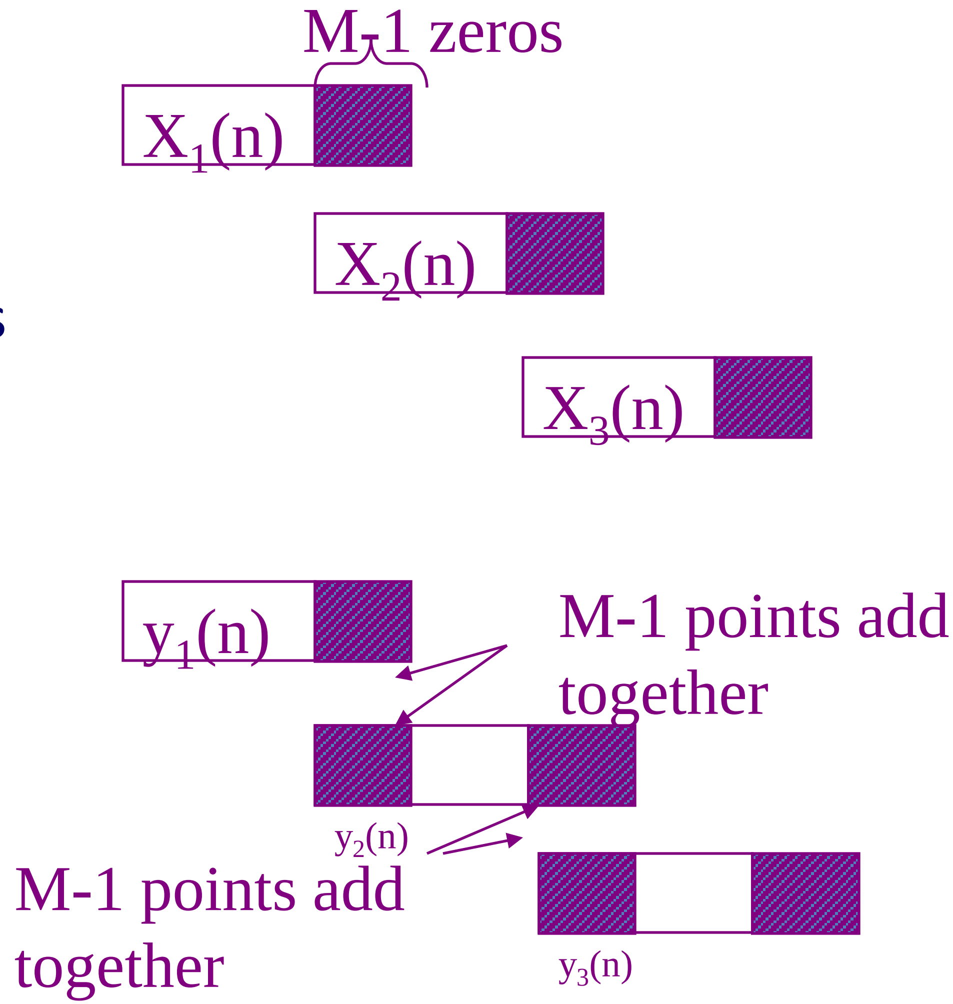
$x_1(n)=\{x(0), x(1), x(2)...........x(L-1),\underbrace{0,0...0}_{\text{M-1 zeros}}\}$

$x_2(n)=\{x(L), x(L+1), x(L+2)...........x(2L-1),\underbrace{0,0...0}_{\text{M-1 zeros}}\}$

Then each $x_k(n)$ circularly convolved with h(n) to obtain $y_k(n)$

# Graphical presentation Overlap Add method

The last M-1 points from each o/p data block must be overlapped and added to first M-1 points of the succeeding data block




$$Y(n)=\{y_1(0),y_1(1)\ldots\ldots ,y_1(L-1),y_1(L)+y_2(0),$$
$$y_1(L+1)+y_2(1),\ldots\ldots y_1(N-1)+y_2(M-1)\ldots\ldots$$

**P1:Compute the convolution using overlap add method for the following sequence** assume N=8

$x(n) = \{1,2,3,3,2,1,-1,-2,-3,5,6,-1,2,1\}$ and $h(n) = \{3,2,1,1\}$

ANS: If N=8 and M=4, then L=5

$$\therefore x_1 = \{1,2,3,3,2,0,0,0\} \quad \therefore x_2 = \{1,-1,-2,-3,5,0,0,0\}$$

$$\therefore x_3 = \{6,-1,2,1,0,0,0,0\} \quad \therefore h(n) = \{3,2,1,1,0,0,0,0\}$$

## Matrix approach for Circular Convolution

$$\therefore y[n] = \begin{bmatrix} h(0) & h(N-1) & .. & h(1) \\ h(1) & h(0) & .. & h(2) \\ \vdots & \vdots & .. & \vdots \\ h(N-1) & h(N-2) & .. & h(0) \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix}$$

**Using Matrix approach for Circular Convolution let us compute each output(Here linear convolution using circular convolution.**

$$\therefore y_1 = x_1 \circledast h = \{3,8,14,18,17,10,5,2\}$$

$$\therefore y_1[n] = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 3 \\ 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 8 \\ 14 \\ 18 \\ 17 \\ 10 \\ 5 \\ 2 \end{bmatrix}$$

$$\therefore y_2 = x_2 \circledast h = \{3,-1,-7,-13,6,5,2,5\}$$

$$\therefore y_2[n] = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ -2 \\ -3 \\ 5 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ -7 \\ -13 \\ 6 \\ 5 \\ 2 \\ 5 \end{bmatrix}$$

$$\therefore y_3 = x_3 \circledast h \quad = \{18,9,10,12,3,3,1,0\}$$

$$\therefore y_3[n] = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 6 \\ -1 \\ 2 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 18 \\ 9 \\ 10 \\ 12 \\ 3 \\ 3 \\ 1 \\ 0 \end{bmatrix}$$

- Adding last 3 points of previous o/p with first 3-points of succeeding O/P

$y_1(n) = \{3,8,14,18,17,10, 5, 2\}$

$y_2(n) = + \{3,-1,-7,-13,6, 5, 2, 5\}$

$y_3(n) = + \{18,9,10,12,3,3,1,0\}$

$$\therefore y(n) = \{3,8,14,18,17,13,4,-5,-13,6, 23,11,15,12,3,3,1,0\}$$

# Overlap Save method

- Consider the length of h(n)= M
- Each data block of i/p subsection overlaps with the previous section by M-1 points.
- Each data block consists of last M-1 points of the previous block of i/p followed by L new data points resulting in a sequence of length L+M-1=N
- Thus the data blocks may be represented as

$x_1(n)=\{\underbrace{0,0\ldots0,}_{\text{M-1 zeros}}\ x(0),\ x(1),\ x(2)\ldots\ldots\ldots\ldots x(L-1)\}$

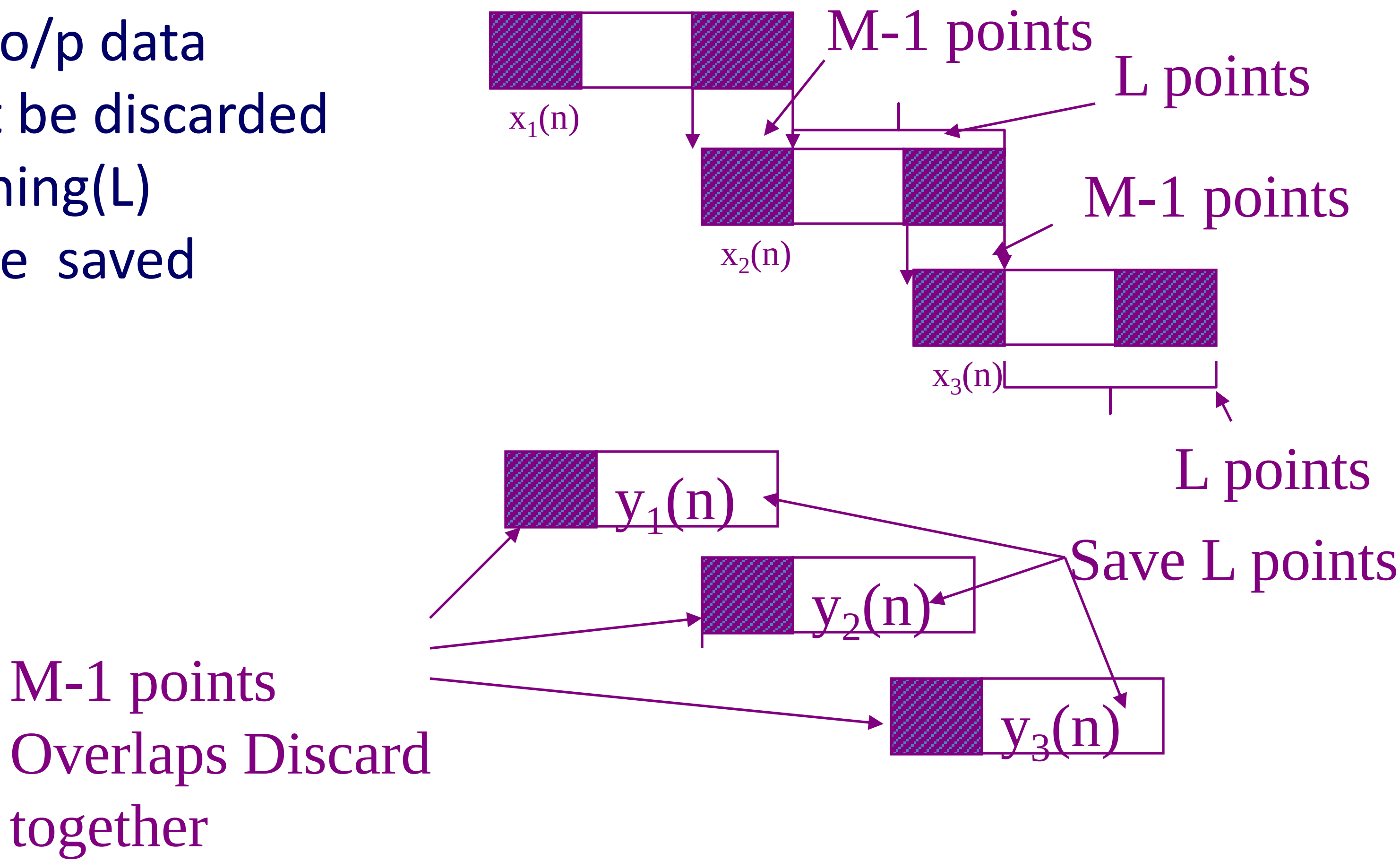
$x_2(n)=\{\underbrace{x(L-M+1),\ldots.\ x(L-1),}_{\text{M-1 points from } x_1(n)}\underbrace{x(L),\ x(L+1)\ldots\ldots\ldots\ldots x(2L-1)}_{\text{New L Data points}}\}$

Then each $x_k(n)$ circularly convolved with

h(n) to obtain $y_k(n)$

# Graphical presentation Overlap Save method

The First M-1 points from each o/p data block must be discarded and remaining(L) samples are saved




**$Y(n)=\{y_1(n), y_2(n), y_3(n)\ldots\ldots.\}$ Where each $y_k(n)$=Saved L points**

**P2:Compute the convolution using overlap save method for the following sequence** assume N=8

$x(n) = \{1,2,3,3,2,1,-1,-2,-3,5,6,-1,2,1\}$ and

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

ANS: If N=8 and M=4, then L=5

$\therefore x_1 = \{0,0,0,1,2,3,3,2\} \quad \therefore x_2 = \{3,3,2,1,-1,-2,-3,5\}$

$\therefore x_3 = \{-2,-3,5,6,-1,2,1,0\} \quad \therefore h(n) = \{3,2,1,1,0,0,0,0\}$

**Using Matrix approach for Circular Convolution to compute each O/P**

$$\therefore y_1[n] = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 2 \\ 3 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 10 \\ 5 \\ 2 \\ 3 \\ 8 \\ 14 \\ 18 \\ 17 \end{bmatrix}$$

$$\therefore y_1 = x_1 \circledast h = \{10,5,2,3,8,14,18,17\}$$

$$\therefore y_2[n] = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 3 \\ 2 \\ 1 \\ -1 \\ -2 \\ -3 \\ 5 \end{bmatrix} = \begin{bmatrix} 19 \\ 17 \\ 15 \\ 13 \\ 4 \\ -5 \\ -13 \\ 6 \end{bmatrix}$$

$$\therefore y_2 = x_2 \circledast h = \{19,17,15,13,4,-5,-13,6\}$$

$$\therefore y_3[n] = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \\ 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} -2 \\ -3 \\ 5 \\ 6 \\ -1 \\ 2 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -3 \\ -12 \\ 7 \\ 23 \\ 17 \\ 15 \\ 12 \\ 3 \end{bmatrix}$$

$$\therefore y_3 = x_3 \circledast h = \{-3,-12,7,23,17,15,12,3\}$$

# Discarding the first three points and saving the last 5 points of each output.

$$y_1(n) = \{10,\ 5,\ 2,\ 3,8,14,18,17\}$$

$$y_2(n) = \{19,17,\ 15,13,4,-5,-13,6\}$$

$$y_3(n) = \{-3,-12,7,23,11,15,\ 12,\ 3\}$$

$$\therefore y(n) = \{3,8,14,18,17,13,4,-5,-13,6,\ 23,11,15,12,3\}$$

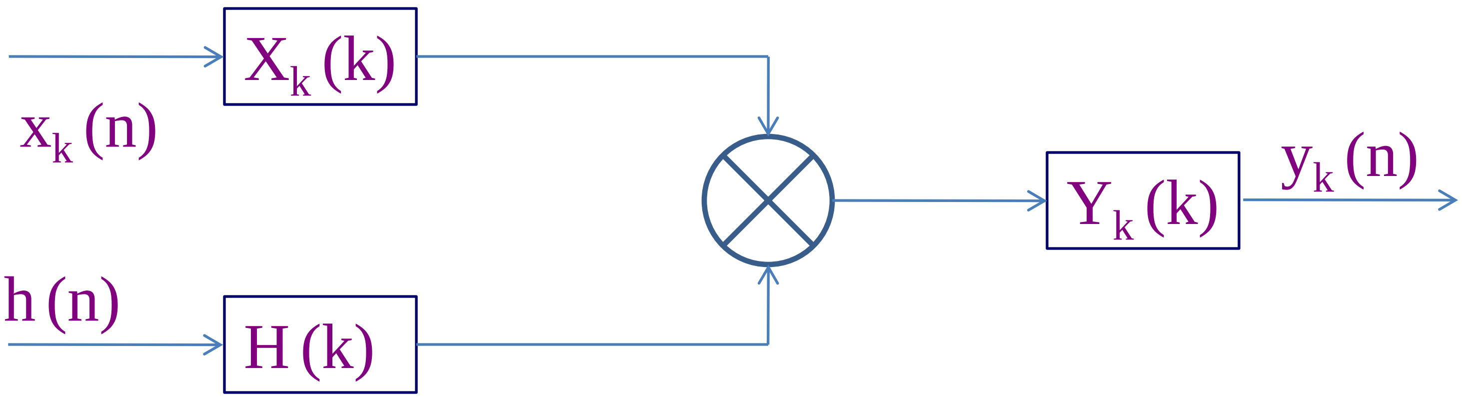
# Difference between Overlap add method and Overlap save method of convolution

**Suppose length of input subsection is L and length of impulse response is M.**

1. M-1 zeros are added at end of each input $x_k(n)$ in overlap and add method whereas in overlap save method only at the start of first subsection the M-1 zeros are added. The last M-1 samples of any section are repeated in the succeeding subsection in case of overlap save method.
2. Though both convolutions are computed for length N, the overlap add method performs linear convolution where as overlap save method performs circular convolution.
3. In Overlap add method outputs of two succeeding sections overlap each other and in overlap save method inputs of two succeeding sections overlap each other.
4. In overlap add method the last M-1 samples of each section are added to the first M-1 samples of succeeding section and rest samples are appended to obtain the overall output. In overlap and save method of convolution first M-1 samples of all outputs are discarded and last L samples are saved to append each other to generate the overall response.

# Use of DFT in Linear filtering

- For the response of each section fast DFT computing algorithms can be used



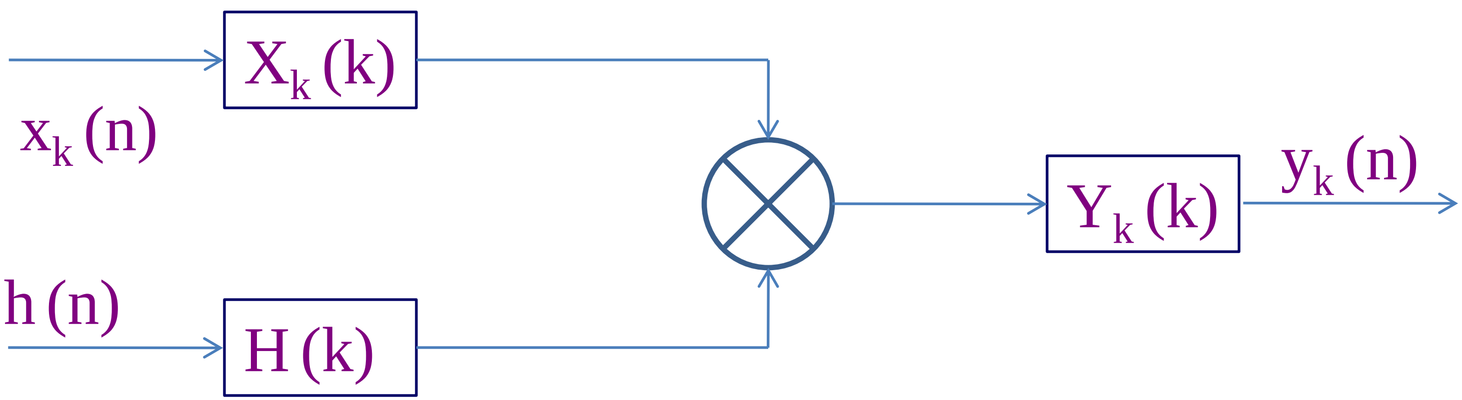

- Then combining response of each section using above two methods the overall response can be obtained.

P3. Determine the response of a LTI system with h(n) as {1,-1} for an input x(n) as{1,0,1,-2,1,2,3,-1,2} using overlap Add method and 4 point circular convolution.

⟹Given N=4, M=2, therefore L=3. Now dividing the I/P sequence into subsections of size 3 and then adding one zero at end of each subsection.

$\therefore x_1(n)=\{1,0,1,0\}$, $x_2(n)=\{-2,1,2,0\}$ and $x_3(n)=\{3,-1,2,0\}$.

- Circular convolution of each subsection with impulse response can be obtained using DFT



$$\therefore X_1[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$\therefore H[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1+j \\ 2 \\ 1-j \end{bmatrix}$$

$$\therefore Y_1[k] = \{X_1[k]\ H[k]\} = \{0,0,4,0\}$$

$$\therefore y_1[n] = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & +j \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 4 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$$

$$\therefore y_1(n) = \{1,-1,1,-1\}$$

$$\therefore X_2[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} -2 \\ 1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ -4-j \\ -1 \\ -4+j \end{bmatrix}$$

$$\therefore H[k] = \{0, 1+j, 2, 1-j\}$$

$$\therefore Y_2[k] = \{X_2[k]\ H[k]\} = \{0, -3-5j, -2, -3+5j\}$$

$$\therefore y_2[n] = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & +j \end{bmatrix} \begin{bmatrix} 0 \\ -3-5j \\ -2 \\ -3+5j \end{bmatrix} = \begin{bmatrix} -2 \\ 3 \\ 1 \\ -2 \end{bmatrix}$$

$$\therefore y_2(n) = \{-2, 3, 1, -2\}$$

$$\therefore X_3[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 3 \\ -1 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ 1+j \\ 6 \\ 1-j \end{bmatrix}$$

$$\therefore H[k] = \{0, 1+j, 2, 1-j\}$$

$$\therefore Y_3[k] = \{X_3[k]\, H[k]\} = \{0, 2j, 12, -2j\}$$

$$\therefore y_3[n] = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & +j \end{bmatrix} \begin{bmatrix} 0 \\ 2j \\ 12 \\ -2j \end{bmatrix} = \begin{bmatrix} 3 \\ -4 \\ 3 \\ 2 \end{bmatrix}$$

$$\therefore y_3(n) = \{3, -4, 3, -2\}$$

$$\therefore y_1(n) = \{1, -1, 1, -1\}$$

$$\therefore y_2(n) = \{-2, 3, 1, -2\}$$

$$\therefore y(n) = \{1, -1, 1, -3, 3, 1, 1, -4, 3, -2\}$$

P4. Determine the response of a LTI system with h(n) as {1,-1} for an input x(n) as{1,0,1,-2,1,2,3,-1,2} using overlap save method and 4 point circular convolution.

⟹Given N=4, M=2, therefore L=3. Now dividing the I/P sequence into subsections of size 3, H(k) is already computed in previous problem.

$\therefore x_1(n)=\{0,1,0,1\}$, $x_2(n)=\{1,-2,1,2\}$ and $x_3(n)=\{2,3,-1,2\}$.

$$\therefore H[k]=\{0,1+j,2,1-j\}$$

$$\therefore X_1[k]=\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & -j & -1 & j\\ 1 & -1 & 1 & -1\\ 1 & j & -1 & -j\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\end{bmatrix}=\begin{bmatrix}2\\0\\-2\\0\end{bmatrix}$$

$$\therefore Y_1[k]=\{X_1[k]\,H[k]\}=\{0,0,-4,0\}$$

$$\therefore y_1[n]=\frac{1}{4}\begin{bmatrix}1 & 1 & 1 & 1\\ 1 & j & -1 & -j\\ 1 & -1 & 1 & -1\\ 1 & -j & -1 & +j\end{bmatrix}\begin{bmatrix}0\\0\\-4\\0\end{bmatrix}=\begin{bmatrix}-1\\1\\-1\\1\end{bmatrix}$$

Discard first and save last three

$$\therefore y_1(n)=\{1,-1,1\}$$

$$\therefore X_2[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ -2 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 2 \\ 4j \\ 2 \\ -4j \end{bmatrix}$$

$$\therefore H[k] = \{0, 1+j, 2, 1-j\}$$

$$\therefore Y_2[k] = \{X_2[k]\ H[k]\} = \{0, -4+4j, 4, -4-4j\}$$

$$\therefore y_2[n] = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & +j \end{bmatrix} \begin{bmatrix} 0 \\ -4+4j \\ 4 \\ -4-4j \end{bmatrix} = \begin{bmatrix} -1 \\ -3 \\ 3 \\ 1 \end{bmatrix}$$

Discard first and save last three

$$\therefore y_2(n) = \{-3, 3, 1\}$$

$$\therefore X_3[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 6 \\ 3-j \\ -4 \\ 3+j \end{bmatrix}$$

$$\therefore H[k] = \{0, 1+j, 2, 1-j\}$$

$$\therefore Y_3[k] = \{X_3[k]\, H[k]\} = \{0, 4+2j, -8, 4-2j\}$$

$$\therefore y_3[n] = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & +j \end{bmatrix} \begin{bmatrix} 0 \\ 4+2j \\ -8 \\ 4-2j \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ -4 \\ 3 \end{bmatrix}$$

$$\therefore y_3(n) = \{1, -4, 3\}$$

$$\therefore y_1(n) = \{1, -1, 1\} \qquad \therefore y_2(n) = \{-3, 3, 1\}$$

Appending all three in sequence

$$\therefore y(n) = \{1, -1, 1, -3, 3, 1, 1, -4, 3\}$$

# Discrete Fourier Transform: Computation Complexity

- The DFT pair was given as

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} \qquad x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j(2\pi/N)kn}$$

- **Computational complexity of DFT :**
  - Each DFT coefficient requires
    - N complex multiplications
    - N-1 complex additions
  - All N DFT coefficients require
    - $N^2$ complex multiplications
    - N(N-1) complex additions

# Discrete Fourier Transform: Computation Complexity

- **Complexity in terms of real operations**
  - **$4N^2$ real multiplications**
  - **2N(N-1) real additions**
- **Most fast methods are based on symmetry properties**
  - **Conjugate symmetry**

$$e^{-j(2\pi/N)k(N-n)} = e^{-j(2\pi/N)kN} e^{-j(2\pi/N)k(-n)} = e^{j(2\pi/N)kn}$$

  - **Periodicity in n and k**

$$e^{-j(2\pi/N)kn} = e^{-j(2\pi/N)k(n+N)} = e^{j(2\pi/N)(k+N)n}$$

# Discrete Fourier Transform: Computation Complexity

- **Baseline for computational complexity N=8:**
  - **Each DFT coefficient requires**
    - **8 complex multiplications**
    - **7 complex additions**
  - **All 8 DFT coefficients require**
    - **64 complex multiplications**
    - **56 complex additions**
- **Complexity in terms of real operations**
    - **256 real multiplications**
    - **112 real additions**
- **To reduce Complexity in terms of operations there is a need to develop the fast computing algorithms for DFT.**

P5. Consider the finite length sequence $x(n)=\delta(n)+2\delta(n-5)$ find

i) 10-point DFT X(k) of x(n)

ii) The sequence that has DFT $Y(k)=e^{j\frac{4\pi}{10}k}X(k)$.

iii) Find the sequence s(n) that has DFT S(k)=X(k)W(k) where W(k) the 10-point DFT of u(n)-u(n-7).

P6. Find N point DFT of $x(n)=4+\cos^2\left(\frac{2\pi}{N}n\right) \quad 0\leq n\leq N-1$

P7. Determine the circular convolution of the sequence x(n)={2,1,2,1} and h(n)={1,2,3,4} using DFT and IDFT.

# REFERENCES

1)DTSP, II edition, authored by Oppenheim and Schafer, PHI.
2)DSP- authored by Deffata, John Wiley
3)DSP- authored by Rabiner Gold
4) DSP: Principles, algo and app, authored by PROAKIS, PE.
5)DSP – Sanjit K Mitra PHI