DISCRETE SIGNAL PROCESSING PRIMER

Prof. Domingo Rodriguez
domingo@ece.uprm.edu
http://www.ece.uprm.edu/~domingo

Revised by Edward Latorre - M.S. student

Spring 2003
Digital Signal Processing:

It is the treatment of signals using digital electronics techniques and computation, in order to extract information important to a user.

Discrete Signal:

It is a signal of the form \( \{x[n], n \in Z \} \), where the set \( Z \) is the set of integers or \( Z = \{... -3, -2, -1, 0, 1, 2, 3,...\} \).

We notice that the signal \( x[n] \) takes an infinite number of values.

Example:

\[ x = \{x[n] = 2^n, n \in Z\} = \{... -\frac{1}{8}, -\frac{1}{4}, -\frac{1}{2}, 1, 2, 4, 8,...\} \]

Another name for a discrete signal is the name sequence.

Causal Discrete Signal:

It is a sequence \( \{x[n]\} \) such that \( x[n] = 0 \) for \( n < 0 \).

Example: Unit step sequence

\[ u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \]
Example: Unit Sample Sequence

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

Discrete Finite Causal Signals:

Let \( Z_N = \{0, 1, 2, \ldots, N-1\} \). Example \( Z_5 = \{0, 1, 2, 3, 4\} \).

A sequence \( \{y[n]\} \) is causal and finite if \( \{y[n], n \in Z_N\} \). In this case we say that the signal has length \( N \).

Example:

\[
y[n] = 3^n, \ n \in Z_4
\]

\[
y[4] = \{y[0], y[1], y[2], y[3]\}
\]

\[
y[4] = \{3^0, 3^1, 3^2, 3^3\} = \{1, 3, 9, 27\}
\]

Discrete System:

A discrete system \( T \) takes as input a discrete signal, say \( \{x[n]\} \) and it produces as output another discrete signal, say \( y[n] \).

Block Diagram Representation of a Discrete System:

\[
x[n], \ n \in Z \xrightarrow{\text{Discrete System } T} y[n], \ n \in Z
\]

\[
x[n] = T(x[n])
\]
Discrete Linear System:

A discrete system \( T \) is linear if it satisfies the following condition:

\[
T = \{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}
\]

Problem 1: Determine if \( T \) is linear

\[
x[n] \quad n \in \mathbb{Z} \longmapsto T \longmapsto y[n] = e^{x[n]} \quad n \in \mathbb{Z}
\]

Solution:

1. Use the equation of the system,

\[
y[n] = T\{x[n]\} = e^{x[n]}, \text{ to obtain:}
\]

\[
T\{x_1[n]\} = e^{x_1[n]}
\]

\[
T\{x_2[n]\} = e^{x_2[n]}
\]

2. Use an intermediate step:

Let \( x_3[n] = ax_1[n] + bx_2[n] \)

Using induction, we obtain:

\[
T\{x_3[n]\} = e^{x_3[n]}
\]

Substituting for \( x_3[n] = ax_1[n] + bx_2[n] \) in the equation above, we obtain:

\[
T\{ax_1[n] + bx_2[n]\} = e^{ax_1[n] + bx_2[n]}
\]

3. Construct the expression

\[
aT\{x_1[n]\} + bT\{x_2[n]\}
\]

from the equation in 1:

\[
aT\{x_1[n]\} + bT\{x_2[n]\} = ae^{x_1[n]} + be^{x_2[n]}
\]

Finally, we check for the identity:

\[
T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}
\]

and determine whether or not the system \( T \) is linear:

\[
\therefore \text{The system } T \text{ is not Linear.}
\]
**Discrete Linear System**

The system \( T \) is linear if:

\[
T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}
\]

Simplified condition:

1. **Additivity or Superposition:** \( a = b = 1 \)
   
   \[
   T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\}
   \]

2. **Homogeneity:** \( b = 0 \)
   
   \[
   T\{ax_1[n]\} = aT\{x_1[n]\}
   \]

For the system to be linear it must satisfy, both, the additivity and homogeneity conditions.

**Example:**

\[
x[n], \ n \in \mathbb{Z} \rightarrow \text{System } T \rightarrow y[n] = T\{x[n]\} = x[n] \cdot x[n] = x^2[n]
\]

Check the homogeneity condition:

1. \( T\{x_1[n]\} = x_1^2[n] \)
   
   \[
aT\{x_1[n]\} = ax_1^2[n]
   \]

2. Let \( g[n] = ax_1[n] \)
   
   \[
   T\{g[n]\} = g^2[n]
   \]

Substituting for \( g[n] = ax_1[n] \), we obtain

\[
T\{ax_1[n]\} = (ax_1[n])^2 = a^2x_1^2[n]
\]

\( \therefore \) The system is not Linear.
Discrete Shift Invariant or Time Invariant System:

A system $T$ is shift invariant or time invariant if it satisfies the following condition: if $y[n] = T\{x[n]\}$, then $T$ is T.I. if $y[n - n_0] = T\{x[n - n_0]\}$.

Example:

\[
x[n], \quad n \in \mathbb{Z} \quad \xrightarrow{T.I. \quad \text{System}} \quad y[n] = T\{x[n]\}
\]

Discrete Filter:

A discrete filter $T$ is a system, which is, both, linear and time invariant.

Example 1:
\[ \delta[n] = -1 \times [n] \]

\[ T(\delta[n]) = h[n] \]

Example 2:

\[ x[n] = \cos \frac{2\pi n}{4}, \ n \in \mathbb{Z} \]

\[ y[n] = T(x[n]) \]

\[ x[0] = \cos \frac{2\pi(0)}{4} = 1 \]
\[ x[1] = \cos \frac{2\pi(1)}{4} = 0 \]
\[ x[2] = \cos \frac{2\pi(2)}{4} = -1 \]
\[ x[3] = \cos \frac{2\pi(3)}{4} = 0 \]
\[ x[4] = \cos \frac{2\pi(4)}{4} = 1 \]

Observation:

Any discrete signal can be expressed as a sum of delayed unit sample functions:

\[ x[n] = \sum_{k=-\infty}^{\infty} x[k] \delta[n-k] \]
Example 3:

\[ x[n] = \cos \frac{2\pi}{4} n, n \in \mathbb{Z} \]

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

\[ x[n] = \sum_{k = -\infty}^{\infty} \cos \frac{2\pi}{4} k \delta[n - k] \]

A finite representation:

\[ x[n] = \sum_{k = -2}^{6} \cos \frac{2\pi}{4} k \delta[n - k] \]

\[ x[n] = \cos \frac{2\pi}{4} \delta[n + 2] + \cos \frac{2\pi}{4} \delta[n + 1] + \cos \frac{2\pi}{4} \delta[n] + \cos \frac{2\pi}{4} \delta[n - 1] + \cos \frac{2\pi}{4} \delta[n - 2] \]

\[ + \cos \frac{2\pi}{4} \delta[n - 3] + \cos \frac{2\pi}{4} \delta[n - 4] + \cos \frac{2\pi}{4} \delta[n - 5] + \cos \frac{2\pi}{4} \delta[n - 6] \]

Problem 2: Obtain the output \( y[n] \)

\[ y[n] = T(x[n]) \]

\[ x[n] = \sum_{k = 0}^{3} x[k] \delta[n - k] \]

\[ x[n] = \sum_{k = 0}^{3} \cos \frac{2\pi}{4} k \delta[n - k] \]


\[ y[n] = T\{x[n]\} = T\{x[n]\} = \sum_{k = 0}^{3} x[k] \delta[n - k] \}


\[ x[n] = \sum_{k = 0}^{3} x[k] \delta[n - k] = T\{\delta[n]\} - T\{\delta[n - 2]\} = h[n] - h[n - 2] \]
Homework 1:

Prove that if the input to a discrete filter is the signal \( x[n] \) and the filter has the impulse response signal \( h[n] \), then the output is given by the convolution operation:

\[
y[n] = \sum_{k=-\infty}^{\infty} x[k] h[n-k], \quad n \in \mathbb{Z}
\]

Solution:

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

\[
y[n] = T\{ \sum_{k=-\infty}^{\infty} x[k] \delta[n-k] \}
\]
Apply Superposition: $T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}$

So, $y[n] = \sum_{k=-\infty}^{\infty} T\{x[k]\delta[n-k]\}$

Apply Homogeneity: $T\{ax_1[n]\} = aT\{x_1[n]\}$

So, $y[n] = \sum_{k=-\infty}^{\infty} x[k]T\{\delta[n-k]\}$

Since, $T\{\delta[n-k]\} = h[n-k]$ then, $y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$

Problem 3:

For Laboratory One

Finite Impulse Response Filter:

It is any filter whose impulse response signal is of final duration, that is, it has length equal to $N_h$.

Example: Average Filter

$h[n] = \begin{cases} 
\frac{1}{N_h}, & 0 \leq n < N_h \\
0, & \text{otherwise} 
\end{cases}$
If N = 4

\[ h[n] = \begin{cases} h[n], & n \geq 0 \\ 0, & n < 0 \end{cases} \]

**Causal Filter:**

A filter \( T \) is called causal if the impulse response signal of the filter is a causal signal.

**RC – Filter:**

The matrix representation of the linear convolution operation is:

\[ y(t) = x(t) * h(t) \]

\[ y(t) = \int_{-\infty}^{\infty} x(\tau) h(t - \tau) d\tau \]

**Matrix Representation of the Linear Convolution Operation:**

Example:

Compute \( y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k] h[n-k] \) for \( x = [1, -2, 3] \) and \( h = [1, -1] \).

Solution:

\[ y[n] = x[n] * h[n] = \sum_{k=0}^{2} x[k] h[n-k] \quad n \in \mathbb{Z} \]

Notice that \( x = \{1, -2, 3\} = \{x[0], x[1], x[2]\} \). This signal is not defined for any other values and it is assumed equal to zero.
Expanding the sum, we get
\[
y[n] = x(0)h[n] + x[1]h[n-1] + x[2]h[n-2], \quad n \in \mathbb{Z}
\]
\[
\]
\[
y[0] = x(0)h[0] + x[1]h[-1] + x[2]h[-2] = 1
\]
\[
\]
\[
\]
\[
\]
\[
\]
\[
\begin{bmatrix}
y[0] \\
y[1] \\
y[2] \\
y[3]
\end{bmatrix}
= \begin{bmatrix}
h[0] & 0 & 0 \\
h[1] & h[0] & 0 \\
0 & h[1] & h[0] \\
0 & 0 & h[1]
\end{bmatrix}
\begin{bmatrix}
x[0] \\
x[1] \\
x[2]
\end{bmatrix}
\]
\[
4 \times 3
\]

**Continuous Filters:**

A continuous filter is usually represented in terms of a differential equation of the form:
\[
a_M \frac{d^M}{dt^M}y(t) + a_{M-1} \frac{d^{M-1}}{dt^{M-1}}y(t) + \ldots + a_0 y(t) = b_N \frac{d^N}{dt^N} x(t) + b_{N-1} \frac{d^{N-1}}{dt^{N-1}} x(t) + \ldots + b_0 x(t)
\]

This can also be expressed as follows:
\[
\sum_{m=0}^{M} a_m \frac{d^m}{dt^m}y(t) = \sum_{n=0}^{N} b_n \frac{d^n}{dt^n} x(t)
\]

We call \(y(t)\) the output of the filter and \(x(t)\) the input or forcing function of the filter.
Example:

Using KVL:

\[ x(t) = R i_c(t) + y(t) \]

\[ i_c(t) = C \frac{dy(t)}{dt} \]

Substituting, we get

\[ x(t) = RC \frac{dy(t)}{dt} + y(t) \]

\[ \therefore \sum_{m=0}^{1} a_m \frac{d^m}{dt^m} y(t) = x(t) \]

\[ a_0 = 1, a_1 = RC, b_0 = 1 \]

Discrete Filters:

Discrete filters are represented using difference equations

\[ \sum_{k=0}^{N} d_k y[n-k] = \sum_{k=0}^{M} P_k x[n-k] \]

Homework 2:

Show that a difference equation can be obtained from a differential equation using the following approximation

\[ \frac{dy(t)}{dt} \approx \Delta y \frac{Y[nTs] - Y[(n-1)Ts]}{Ts} \]

1. Use the RC circuit given.
2. Set \( T_s = 1 \).
Solution:
For an RC-filter we have the following differential equation

\[ y(t) = x(t) - RC \frac{dy(t)}{dt} \]

Using the proposed approximate, we get

\[ y[nTs] = x[nTs] - RC \left[ \frac{y[nTs] - y[(n-1)Ts]}{Ts} \right] \]

Simplifying by normalizing the equation (setting \( Ts = 1 \)), we get

\[
\begin{align*}
(1 + RC)y[n] &= x[n] + RCy[n-1] \\
y[n] &= \frac{1}{1 + RC} x[n] + \frac{RC}{1 + RC} y[n-1] \\
\alpha &= \frac{1}{1 + RC}, \beta = \frac{RC}{1 + RC}
\end{align*}
\]

Discrete Signal

A discrete signal has as its domain a discrete set. A discrete signal can be obtained from a continuous signal by making the time axis a discrete set:

Example:

\[ x : \mathbb{R} \to \mathbb{C} \]

\[ t \to x(t) = e^{j2\pi f_0 t} \]

To make \( x \) a discrete signal, we proceed in the following manner

\[ t \bigg|_{t=nTs, n \in \mathbb{Z}} \]

\[ x(t) \bigg|_{t=nTs, n \in \mathbb{Z}} = x[nTs] \]
Digital Signal

A digital signal has as its co-domain a finite discrete set.

Example:

\[ x : \mathbb{R} \rightarrow B \]

\[ t \rightarrow x(t) = \begin{cases} 1 & \text{if } t < 0 \\ 0 & \text{if } t \geq 0 \end{cases} \]

A discrete signal can be converted into a digital signal by using a quantizer:

\[ x[n], n \in \mathbb{Z} \rightarrow Q \rightarrow x_q[n] \in A \]

The set \( A \) is a finite discrete set.

FIR Filter used in Cardiology:

\[ h[n] = c[n] \]

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

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

Let \( m = n - k \), thus \( k = n - m \).

Substituting:

\[ y[n] = \sum_{m=n+n}^{m=\infty} x[n-m] h[n-(n-m)] \]

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

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

If \( h[k] = \{h[-N],...,h[0],...,h[N]\} \)

**Discrete Filter Implementation:**

A large class of discrete filters can be expressed in terms of a difference equation of the form:

\[
\sum_{k=0}^{M} d_k y[n-k] = \sum_{k=0}^{N} b_k x[n-k]
\]

This is the only type of filters that we will study in this primer!

**Operations:**

\[
\begin{align*}
\text{SUM} & \quad \bullet \quad \text{D delay} \quad \text{S shift} \quad \text{Z-1 delay} \\
\end{align*}
\]

Example:

\[
d_0 = 1, \quad b_0 = 1, \quad b_1 = -1, \quad b_2 = 3.7
\]

\[ y[n] = \sum_{k=0}^{2} b_k x[n-k] \]

Expanding

\[ y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] \]

if \( x[n] = \delta[n] \), then \( y[n] = h[n] \)
If I know $h[n]$ I can tell you the output of the system for any input $x[n]$.

**Discrete Time Fourier Transform:**

Let $x[n]$ be a discrete signal. Its discrete-time Fourier transforms is defined as follows

$$F\{x[n]\} = DTFT\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}, \quad \omega \in \mathbb{R}, \quad j = \sqrt{-1}$$

Remember that $e^{-j\omega n} = \cos \omega n - j\sin \omega n$. This implies that the DTFT of the signal $x[n]$ is a complex function signal.
Property of the DTFT:

- Always Periodic with period equal to $2\pi$.

Proof:

A signal $X(\omega)$ is periodic with period $\omega_p$ if the following condition is satisfied:

$$X(\omega + \omega_p) = X(\omega).$$

Define $X(\omega) = \mathcal{F}\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}; \omega \in \mathbb{R}$

If we let $\omega$ go to $\omega + \omega_p$ by changing the argument of $X(\omega)$, we get

$$X(\omega + \omega_p) = \sum_{n=-\infty}^{\infty} x[n]e^{-j(\omega + \omega_p)n} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}e^{-j\omega_p n}$$

Allow $\omega_p = 2\pi$

Then, $e^{-j\omega_p n} = e^{-j2\pi n} = \cos(2\pi n) - j \sin(2\pi n)$, $n \in \mathbb{Z}$

We then have the following result:

$$X(\omega + \omega_p)|_{\omega_p = 2\pi} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = X(\omega)$$

Example:

**Discrete Fourier Transform:**

This is only defined for finite discrete signals, say of length $N$.

Let $x[n]$ be a discrete signal of length $N$. Its DFT is given by the following equation:

$$X(\omega)|_{\omega = \omega_k = \frac{2\pi k}{N}} = X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\omega_k n} = \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi kn}{N}}, k \in \mathbb{Z}_N$$
The DFT can be represented in matrix form:

\[ X = F_N x \]

When \( x \) is a column vector and is the input signal, \( X \) is a column vector and it is the output signal or transformed signal and \( F_N \) is a matrix of order \( N \) (\( N \) rows by \( N \) Columns) called the Fourier matrix.

**Homework 3:**

Write in matrix form the DFT of an arbitrary signal of length \( N=8 \) and check the MATLAB instruction “dftmtx”.

**Problem 4: Spectral Resolution:**

Let \( \{1, 1, 1, 1\} = \{x[0], x[1], x[2], x[3]\} \)

Let \( x_m = \{x[0], x[1], x[2], x[3], z_m\}, m = 1, 2, 3, ..., 8 \)

\[ z_1 = \{0, 0, 0, 0\} \quad x_1 = \{x[0], x[1], x[2], x[3], z_1\} = \{x[0], x[1], x[2], x[3], 0, 0, 0, 0\} \]

\[ z_2 = \{z_1 z_1\} \]

\[ z_3 = \{z_2 z_1\} \]

\[ \vdots \]

\[ z_8 = \{z_7 z_1\} \]

\[ X[k] = \text{dft}\{x[n]\} = \sum_{n=0}^{N-1} x[n]e^{-j2\pi nk/N} \]

Let \( W_N = e^{-j2\pi/N} \)

\[ X[k] = \text{dft}\{x[n]\} = \sum_{n=0}^{N-1} x[n]W_N^{kn}; \ N = 4 \]

\[ X[k] = \sum_{n=0}^{3} x[n]W_4^{kn} = x[0] + x[1]W_4^k + x[2]W_4^{2k} + x[3]W_4^{3k}, \ k \in Z_4 \]


\[ X[k] = 1 + W_4^k + W_4^{2k} + W_4^{3k}, \ k \in Z_4 \]

\[ X_1[k] = \text{DFT}\{x_1[n]\} = \sum_{n=0}^{N-1} x_1[n]W_N^{kn}; \ N = 8 \]
\[
X_i(k) = \sum_{n=0}^{7} x_i[n] W_8^{kn} = x_i[0] + x_i[1] W_8^k + x_i[2] W_8^{2k} + x_i[3] W_8^{3k} + ... + x_i[7] W_8^{7k}, \quad k \in \mathbb{Z}_8
\]
\[
\]
\[
X_i[k] = x_i[0] + x_i[1] W_8^k + x_i[2] W_8^{2k} + x_i[3] W_8^{3k}; \quad k \in \mathbb{Z}_8
\]
\[
X_i[k] = 1 + W_8^k + W_8^{2k} + W_8^{3k}, \quad k \in \mathbb{Z}_8
\]
\[
X(\omega) = DTFT\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}; \quad \omega \in \mathbb{R}
\]
\[
X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = x[0] + x[1]e^{-j\omega} + x[2]e^{-j2\omega} + x[3]e^{-j3\omega}; \quad \omega \in \mathbb{R}
\]
\[
X_i(\omega) = DTFT\{x[n]\} = \sum_{n=-\infty}^{\infty} x_i[n]e^{-j\omega n}
\]
\[
X_i(\omega) = \sum_{n=0}^{\infty} x_i[n]e^{-j\omega n} = x_i[0] + x_i[1]e^{-j\omega} + x_i[2]e^{-j2\omega} + x_i[3]e^{-j3\omega}
\]
\[
X_i(\omega) = x_i[0] + x_i[1]e^{-j\omega} + x_i[2]e^{-j2\omega} + x_i[3]e^{-j3\omega}, \quad \omega \in \mathbb{R}
\]
Spectral Resolution:

It is defined as the smallest distance between two samples in the spectrum of a signal.
Problem 5:

A radar signal is normally processed in order to extract information regarding the distance and uniform speed of an object in space. After a signal has been demodulated and sampled, it is expressed as \( y[n] = x[n - n_0]e^{j\omega_0 n} \). This signal, known as the received signal \( y[n] \) has the parameter \( n_0 \), from where we can determine the range of the object from the radar, and the parameter \( \omega_0 \), known as the Doppler frequency, from where we can obtain the DTFT of the signal \( y[n] \).

Solution:

\[
Y(\omega)\big|_{\omega = \omega_0} = \sum_{n=\infty}^{\infty} x[n - n_0]e^{j\omega_0 n}e^{-j\omega n}; \omega \in \mathbb{R}
\]

\[
= \sum_{n=\infty}^{\infty} x[n - n_0]e^{-j(\omega - \omega_0)n}; \omega \in \mathbb{R}
\]

Let \( k = n - n_0; \ n = k + n_0 \)

\[
Y(\omega) = \sum_{k=\infty}^{\infty} x[k]e^{-j(\omega - \omega_0)(k+n_0)}
\]

\[
Y(\omega) = \sum_{k=\infty}^{\infty} x[k]e^{-j(\omega - \omega_0)k}e^{-j(\omega - \omega_0)n_0}
\]

\[
Y(\omega) = e^{-j(\omega - \omega_0)n_0}\sum_{k=\infty}^{\infty} x[k]e^{-j(\omega - \omega_0)k}
\]
Let $\lambda = \omega - \omega_0$

$$Y(\omega) = e^{-j(\omega - \omega_0)\lambda} \sum_{k=-\infty}^{\infty} x[k] e^{-j\lambda k}$$

$\therefore \quad Y(\omega) = e^{-j(\omega - \omega_0)\lambda} X(\lambda) = e^{-j(\omega - \omega_0)\lambda} X(\omega - \omega_0)$

**Periodic discrete signals:**

A signal $x[n]$ is said to be periodic, with fundamental period $N$, if the following condition is satisfied:

$$x[n + qN] = x[n], \text{ for } q \in \mathbb{Z}$$

Example:

![Graph of a periodic signal](image)

The signal $x[n]$ has a fundamental period equal to $N$. In this case $N = 4$:

Let $q = 1$

$$x[n + 4] = x[n]$$

For $n = -3$

$$x[-3 + 4] = x[-3]$$

$\therefore \quad x[-3] = x[1]$

Observation:

Any periodic signal $x[n]$ with fundamental period $N$, can uniquely be represented by a causal signal $x[n]$, of length equal to $N$, whose values are equal to the $N$ values of the periodic signal in its fundamental period.
Example:

The periodic signal $x[n]$, with fundamental period $N = 4$, can be represented uniquely by the signal $x[n]$, of length $N = 4$. Remember that $x[n]$ is causal with $x[n] = \{x[0], x[1], x[2], x[3]\} = \{0, 1, 2, 3\}$.

**Cyclic or Circular Convolution of Periodic Signals:**

Given two periodic signals, say $x[n]$ and $h[n]$, with the same fundamental period $N$, the cyclic or circular convolution of $x[n]$ and $h[n]$ is a new periodic signal

$$y[n] = x[n] \ast_c h[n],$$

with fundamental period also equal to $N$ and which is defined by the following equation

$$y[n] = \sum_{k=0}^{N-1} x[k]h[n-k]; n \in Z_N.$$

**Circular or Cyclic Convolution of Periodic signals using Causal Representations:**

Let $x[n]$ and $h[n]$ be two periodic signals with fundamental period $N$. Let $x[n]$ and $h[n]$ be their causal representations, respectively. The circular or cyclic convolution of the causal representation is a new causal signal, of length $N$, and denoted by $y[n]$. The signal $y[n]$ is given by

$$y[n] = \sum_{k=0}^{N-1} x[k]h[<n-k>_N]; n \in Z_N$$

The symbol $<p>_N$ denotes the remainder of $p$ after being divided by $N$. This is sometimes called “$p$ modulo $N$”. The periodic signal $y[n]$ is obtained from its causal representation $y[n]$ by repeating the causal signal $y[n]$, starting at the fundamental period.

Observation:

Remainder $\left( \frac{P}{N} \right) = \text{Remainder} \left( \frac{P + qN}{N} \right) = \text{Remainder} \left( \frac{P}{N} \right) + \text{Remainder} \left( \frac{qN}{N} \right)$

1. $<5>_4 = \text{Remainder} \left( \frac{5}{4} \right) = 1$

2. $<-1>_4 = < -1 + 4 >_4 = < 3 >_4 = 3$
Problem 6: Compute \( x[n]O_N h[n] = y[n] \)

Solution:
\[
\]

**Homework 4:**

Write the cyclic convolution in matrix form.

Relating the cyclic convolution and the DFT:

The cyclic convolution \( x[n]O_N h[n] = y[n] \) can be expressed in matrix form as follows:

\[
y = H_N x
\]

Here, the matrix \( H_N \) is called the filter matrix of the cyclic convolution and it has the property that all its diagonals have a constant parameter. This matrix is also called a circulant matrix since the whole matrix can be generated from the first column (or first row) of the matrix. The first column of the matrix contains the impulse response values of the filter.

Example: \( N = 4 \)

\[
y[n] = x[n]O_N h[n] = \sum_{k=0}^{N-1} x[k]h[< n - k >]; n \in Z_4
\]

Expanding, we get
\[
\]

\[
\]

\[
\begin{bmatrix}
y[0] \\
y[1] \\
y[2] \\
y[3]
\end{bmatrix} = \begin{bmatrix}
\end{bmatrix} \begin{bmatrix}
x[0] \\
x[1] \\
x[2] \\
x[3]
\end{bmatrix}
\]
Example:
\[ x[n + q \cdot 4] = x[n] \]
\[ q = 1 \]
\[ x[n + 4] = x[n] \]

Observation:
1. The efficiency of computing a cyclic convolution operation can be improved using a Fast Fourier Transform (FFT) algorithm. An FFT algorithm is an efficient method for computing the DFT.
2. Any linear convolution can be computed using a cyclic convolution operation. Remember that the filters only do linear convolution.
3. The Discrete Time Domain Convolution Theorem states that the DFT of the cyclic convolution of two discrete signals is equal to the product of the DFT of each of the individual signals.

Discrete Time Domain Convolution Theorem or DTDCT:
Let \( x[n] \) and \( h[n] \) be two causal representations, each of length \( N \).
Let \( y[n] = x[n] \circ h[n] \) be the cyclic convolution operation between these two signals.
Let \( Y[k] = DFT\{y[n]\} \); \( X[k] = DFT\{x[n]\} \); \( H[k] = DFT\{h[n]\} \)
\( DFT\{x[n] \circ h[n]\} = DFT\{x[n]\} \bullet DFT\{h[n]\} \)
\[ \therefore Y[k] = X[k] \bullet H[k] \]

In MATLAB, the Hadamard product is expressed as ‘ . * ’.

Expressing a Hadamard product as a Matrix-Vector Multiplication:
Example:
\[
A = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}; \quad B = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix}; \quad C = A \bullet B
\]
\[
C = a_0 \bullet b_0 = a_1 \bullet b_1 = a_2 \bullet b_2
\]
In matrix-vector product form, we get:

1. Write one of the vectors as a diagonal matrix.
2. Perform a matrix-vector multiplication operation with the other vector.

\[
\begin{bmatrix}
    a_0 & 0 & 0 \\
    0 & a_1 & 0 \\
    0 & 0 & a_2
\end{bmatrix}
\begin{bmatrix}
    b_0 \\
    b_1 \\
    b_2
\end{bmatrix}
= \begin{bmatrix}
    a_0 b_0 \\
    a_1 b_1 \\
    a_2 b_2
\end{bmatrix}
\]

\[
\begin{bmatrix}
    b_0 & 0 & 0 \\
    0 & b_1 & 0 \\
    0 & 0 & b_2
\end{bmatrix}
\begin{bmatrix}
    a_0 \\
    a_1 \\
    a_2
\end{bmatrix}
= \begin{bmatrix}
    a_0 b_0 \\
    a_1 b_1 \\
    a_2 b_2
\end{bmatrix}
\]

**Applying the DFT Fourier Matrix to the DTDCT:**

\[Y = F_N \cdot y \quad X = F_N \cdot x \quad H = F_N \cdot h\]

\[Y[k] = X[k] \cdot H[k]\]

\[F_N \cdot y = (F_N \cdot x) \cdot (F_N \cdot h)\]

\[
F_N \cdot y = D_h X; \\
D_h = \begin{bmatrix}
    H[0] & 0 & 0 & \ldots & 0 \\
    0 & H[1] & 0 & \ldots & 0 \\
    0 & 0 & H[2] & \ldots & 0 \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    0 & \ldots & H[n-1]
\end{bmatrix}
\]

\[F_N y = D_h \cdot (F_N x)\]

\[F_N y = D_h \cdot F_N x\]

\[F_N^{-1}F_N y = F_N^{-1}D_h \cdot F_N x\]

\[I_N y = F_N^{-1}D_h \cdot F_N x\]

\[y = (F_N^{-1}D_h \cdot F_N)x\]

\[y = H_N x\]

\[\therefore H_N = F_N^{-1}D_h F_N\]

\[F_N H_N F_N^{-1} = F_N (F_N^{-1}D_h F_N) F_N^{-1}\]

\[\therefore D_h = F_N H_N F_N^{-1}\]

Therefore, the Discrete Fourier Matrix diagonalizes any circulant matrix \( H_N \).
Review:

Time Invariance

A discrete system $T$ is said to be time invariant (T.I.) if it commutes with the discrete system $D_{n_0}$. The system $D_{n_0}$ acts on a discrete signal in the following manner:

$$D_{n_0}\{x[n]\} = x[n-n_0]$$

The system $T$ is T.I. if

$$T(D_{n_0}\{x[n]\}) = D_{n_0}(T\{x[n]\})$$

The system $D_{n_0}$ is called an $n_0 −$ delay system. It is also a filter.

Example: Show if $T\{x[n]\} = x[n]cos\frac{2\pi n}{N}$ is a T.I. system.

Solution:

Is $T$ T.I.?

1. $(TD_{n_0})\{x[n]\} = T(D_{n_0}\{x[n]\})$

Let $g[n] = x[n-n_0]$

$$T\{g[n]\} = g[n]cos\frac{2\pi n}{N} = x[n-n_0]cos\frac{2\pi n}{N}$$

2. $D_{n_0} T\{x[n]\} = D_{n_0}\{x[n]cos\frac{2\pi n}{N}\}$

Let $s[n] = x[n]cos\frac{2\pi n}{N}$

$$D_{n_0}\{s[n]\} = s[n - n_0] = x[n - n_0]cos\frac{2\pi [n - n_0]}{N}$$

∴ The system $T$ is not T.I.!
Observation:

\[ T(D_{n_0}\{x[n]\}) = D_{n_0} T\{x[n]\} \]

\[ T\{x[n-n_0]\} = y[n-n_0] \]

**Block Diagram Representation of T.I.:**

**Z – Transform:**

Let \( x[n], n \in Z \) be an arbitrary signal. Its Z – transform, if it exists, is given by

\[ X(Z) = \sum_{n=-\infty}^{\infty} x[n]Z^{-n}; \quad Z \in C \] where \( C \) is the set of complex numbers.

**Z – Plane**

\[ Z_0 = (Z_{OR}, Z_{OI}) \]

\[ Z_{OR} = \text{Re}(Z_0) \]

\[ Z_{OI} = \text{Im}(Z_0) \]

\[ Z_{OR} = |Z_0| \cos \theta \]

\[ Z_{OI} = |Z_0| \sin \theta \]

\[ Z_0 = |Z_0|e^{j\theta}; \quad \theta \text{ is the argument of } Z_0. \]

We say that a Z – transform function, say \( X(Z) \), exists at a point, say \( Z_0 \), if

\[ X(Z_0) < \infty. \]

Remember \[ X(Z_0) = \sum_{n=-\infty}^{\infty} x[n]Z_0^{-n} \]
Region of Convergence of a Z – transform Function:

Let \( X(Z) \) be an arbitrary Z – transform function.

The set of all values of \( Z \) for which \( X(Z) \) exists is called its region of convergence or R.O.C:

\[
\text{R.O.C of } X(Z) = \{ Z : X(Z) = \sum_{n=-\infty}^{\infty} x[n] Z^{-n} < \infty; \ Z \in C \}
\]

Relating the Z – transform and the Discrete Time Fourier Transform:

Observation:

The equation \( Z = e^{j\theta}; 0 \leq \theta \leq 2\pi \)

\[
Z = e^{j\theta} = \cos \theta + j \sin \theta
\]

\[
|Z| = (\cos^2 \theta + \sin^2 \theta)^{\frac{1}{2}} = 1
\]

Let \( X(Z) \) be the Z – transform of the signal \( x[n] \). The discrete time Fourier transform (DTFT) of the signal \( x[n] \) can be obtained from its Z – transform \( X(Z) \) by evaluating \( X(Z) \) on the unit circle:

\[
X(\omega) = DTFT\{x[n]\} = X(Z)|_{Z = e^{j\omega}} = X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] (e^{j\omega})^{-n}; \ 0 \leq \omega \leq 2\pi
\]

Discrete Complex Signal:

\[
x : Z \rightarrow C
\]

\[
n \rightarrow x[n] = e^{j \frac{2\pi kn}{N}}
\]

A complex signal can be represented in the following manner:

\[
x[n] = x_r[n] + jx_i[n], \ n \in Z.
\]

If \( n \) is a fixed number or value \( n_o \), we get a complex number.

\[
x[n_0] = x_r[n_0] + jx_i[n_0]
\]
A complex function can also be represented in polar notation:

\[ x[n] = |x[n]| e^{j\theta[n]} \]

\[ |x[n]| = \left( X_R^2[n] + X_I^2[n] \right)^{\frac{1}{2}} \]; This is the magnitude or absolute value of \( x[n] \).

\[ \theta[n] = \text{Arg}\{x[n]\} = \tan^{-1} \frac{X_I[n]}{X_R[n]} = \theta[n] \]; This is the phase, angle, or argument of \( x[n] \).

**Properties of Complex Numbers:**

Let \( a \) and \( b \) be two complex numbers. Let the \( a^* \) denote the complex conjugate of \( a \). Then:

1. \((a + b)^* = a^* + b^*\)
   
   \[ a = a_R + j a_I; \quad b = b_R + j b_I \]
   \[ a^* = a_R - j a_I; \quad b^* = b_R - j b_I \]
   \[ a + b = (a_R + b_R) + j(a_I + b_I) \]
   \[ a^* + b^* = (a_R + b_R) - j(a_I + b_I) \]
   \[ (a + b)^* = (a_R + b_R) - j(a_I + b_I) \]

2. \((ab)^* = a^* \cdot b^*\)

3. \( c = \frac{a}{b} \)
   
   \[ |c| = \left| \frac{a}{b} \right| = \frac{|a|}{|b|} \]
Problem 7: Obtain the $DTFT\{x^*[n]\}$

Solution:

$$DTFT\{x^*[n]\} = \left(\sum_{n=-\infty}^{\infty} x^*[n]e^{-j\omega n}\right)^*$$

$$(\sum_{n=-\infty}^{\infty} x^*[n]e^{-j\omega n})^* = \sum_{n=-\infty}^{\infty} (x^*[n]e^{-j\omega n})^*$$

$$= \sum_{n=-\infty}^{\infty} x[n](e^{-j\omega n})^* = \sum_{n=-\infty}^{\infty} x[n]e^{j\omega n}$$

$$= \sum_{n=-\infty}^{\infty} x[n]e^{-j(-\omega)n}$$

Let $\lambda = -\omega$:

$$= \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = X(\lambda) = X(-\omega)$$

$$\therefore DTFT\{x^*[n]\} = [X(-\omega)]^* = X^*(-\omega)$$

Problem 8: Obtain the $DTFT\{x[-n]\}$

Solution:

$$DTFT\{x[-n]\} = \sum_{n=-\infty}^{\infty} x[-n]e^{-j\omega n}$$

Let $m = -n$:

$$\sum_{n=-\infty}^{\infty} x[-n]e^{-j\omega n} = \sum_{m=-\infty}^{\infty} x[m]e^{-j\omega (-m)} = \sum_{m=-\infty}^{\infty} x[m]e^{-j(-\omega)m}$$

Let $\lambda = -\omega$:

$$= X(\lambda) = X(-\omega)$$

A signal $x[n]$ is real if $x^*[n] = x[n]$.

A real signal $x[n]$ is called even if $x[-n] = x[n]$.

Example of a real even signal:

$$x[n] = \cos(\omega_0 n)$$
Fourier Transform (DTFT) of a real even signal.

Let \( x[n] \) be real and even:

\[
x[n] = x^*[n]
\]

\[
x[n] = x[-n]
\]

\[
X(\omega) = X^*(-\omega) = X(-\omega)
\]

\(\text{Real Condition}\)

\(\text{Symmetry Condition}\)

Let \( x[n] \) be a real signal:

\[
x^*[n] = x[n]
\]

\[
X^*(-\omega) = X(\omega)
\]

The magnitude of every real function is symmetric

\[
|X^*(-\omega)| = |X(\omega)|
\]

\(\text{Symmetry Condition}\)

**Inverse DTFT:**

Let \( X(\omega) \) be the DTFT of the signal \( x[n] \). We can recover the signal \( x[n] \) from its Fourier transform by using the formula (IDTFT):

\[
x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega.
\]

**Problem 9:**

Obtain the DTFT of \( x[n] = \alpha^n u[n], |\alpha| < 1 \).

**Solution:**

\[
X(\omega) = DTFT\{x[n]\} = \sum_{n=0}^{\infty} \alpha^n \mu[n] e^{-j\omega n} = \sum_{n=0}^{\infty} \alpha^n e^{-j\omega n}
\]
Expanding, we get

\[ X(\omega) = 1 + \alpha e^{-j\omega} + \alpha^2 e^{-j2\omega} + \alpha^3 e^{-j3\omega} + \cdots \]

\[ X(\omega) = \sum_{n=0}^{\infty} (\alpha e^{-j\omega})^n \]

Let \( b = \alpha e^{-j\omega} \)

\[ X(\omega) = \sum_{n=0}^{\infty} b^n = 1 + b + b^2 + b^3 + \cdots \]

\[ X(\omega) - bX(\omega) = 1 \]

\[ (1 - b)X(\omega) = 1 \]

\[ \therefore X(\omega) = \frac{1}{1 - b} = \frac{1}{1 - \alpha e^{-j\omega}} \]

**BIBO Stability:**

A system \( T \) is said to be BIBO (Bounded Input Bounded Output) Stable if its impulse response satisfies the following condition:

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

Transfer function of a system \( T \):

A system \( T \) has a transfer function \( H(z) \) if the system is a filter and \( h[n] \) is its impulse response. Thus, we have \( H(z) = Z\{h[n]\} = \sum_{n=-\infty}^{\infty} h[n]z^{-n} \).

The frequency response of a filter is defined as the DTFT of its impulse response. Thus,

\[ H(\omega) = DTFT\{h[n]\} = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n} \]

We have \( H(z) \cdot X(z) = Y(z) \) or \( Z\{x[n]*h[n]\} = Z\{y[n]\} = X(z) \cdot H(z) \)

\[ H(z) = \frac{Y(z)}{X(z)} \]

**Poles of a Transfer Function \( H(z) \):**

The poles of a transfer function are the values of \( z \) for which \( H(z) \) does not exist or tends toward infinite.
Zeros of a Transfer Function $H(z)$:

The Zeros of a transfer function $H(z)$ are the values of $z$ for which the transfer function goes to zero.

Difference Method for filter Design Starting from an Analog Passive Filter:

Analog System:

This method consists of turning a differential equation, representing an analog passive filter, into a difference equation representing a discrete time filter.

Example: First-order RC Filter

1. Differential equations

\[ x(t) = Ri_e(t) + y(t) \]

\[ i_e(t) = C \frac{dy(t)}{dt} \]

\[ x(t) = RC \frac{dy(t)}{dt} + y(t) \]

\[ y(t) = x(t) - RC \frac{dy(t)}{dt} \]

We want

\[ \frac{dy(t)}{dt} \approx \frac{y[nTs] - y[(n-1)Ts]}{Ts} \]

We proceed with this substitution in the differential equation in order to obtain the difference equation desired:

\[ x[nTs] = RC \left( \frac{y[nTs] - y[(n-1)Ts]}{Ts} \right) + y[nTs] \]
We proceed to normalize the sampling time:

\[ T_s = 1 \]

We then have

\[
x[n] = RC(y[n] - y[n-1]) + y[n] \\
= RCy[n] - RCy[n-1] + y[n]
\]

\[
x[n] + RCy[n-1] = (RC + 1)y[n]
\]

\[
\therefore y[n] = \frac{x[n] + RCy[n-1]}{RC + 1}
\]

\[
1 \cdot y[n] = \frac{1}{RC + 1} x[n] + \frac{RC}{RC + 1} y[n-1]
\]

**Discrete System**: First-order filter simulating an RC Filter.

To obtain the impulse response, we proceed as follows:

Let \( x[n] = \delta[n] \), then \( y[n] = h[n] \).

\[
h[n] = b_0\delta[n] + a_1h[n-1]
\]

\[
h[0] = b_0\delta[0] + a_1h[-1] = b_0
\]

\[
h[1] = b_0\delta[1] + a_1h[0] = a_1b_0
\]

\[
h[2] = b_0\delta[2] + a_1h[1] = a_1^2b_0
\]

\[
h[3] = a_1\delta[2] + a_1h[3] = a_1^3b_0
\]

\[
h[m] = a_1^m b_0
\]

\[
\therefore h[n] = a_1^n b_0 u[n]
\]
The transfer function of the filter is given by

\[ H(z) = Z\{h[n]\} = \sum_{n=-\infty}^{\infty} h[n]z^{-n} \]

\[ H(z) = \sum_{n=0}^{\infty} a_n b_0 z^{-n} = b_0 \sum_{n=0}^{\infty} \left( \frac{a_1}{z} \right)^n \]

Let \( c = \frac{a_1}{z} \),

\[ H(z) = b_0 \sum_{n=0}^{\infty} (c)^n \]

\[ H(z) = b_0 (1 + c + c^2 + ...) \]

\[ cH(z) = b_0 (c + c^2 + c^3 + ...) \]

\[ H(z) - cH(z) = b_0 (1) \]

\[ H(z) = \frac{b_0}{(1-c)} \quad |c| < 1 \]

The transfer function of the filter exists only for values of \( z \) which satisfy the condition \(|c| < 1\), hence the Region of Convergence is \( z > a_1 \).

Problem 10:
Obtain the poles and zeros of the previous transfer function and draw a pole-zero plot.

Solution:

\[ H(z) = \frac{b_0}{(1-a_1/z)} = \frac{zb_0}{z-a_1} \]

We have one pole at \( z-a_1 = 0 \) or \( z = a_1 \).

We have one zero at \( z = 0 \).

A filter \( T \) is said to be BIBO stable if

\[ \sum_{n=-\infty}^{\infty} |h[n]| < \infty. \]

We know that if we have \( H(z) \), we can get \( H(\omega) \) by evaluating \( H(z) \) on the unit circle:

\[ H(\omega) = H(z)|_{z=e^{j\omega}} = \sum_{n=-\infty}^{\infty} h[n]z^{-n} \bigg|_{z=e^{j\omega}} = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n} \]
Observation:

1) A right-sided signal is a causal signal if it starts at a point \( n \geq 0 \).
2) A left-sided signal is anti-causal if it starts at a point \( n \leq -1 \).
3) The poles are the values of \( z \) for which \( H(z) \) tends to infinity. The region of convergence or R.O.C. of \( H(z) \) are the values of \( z \) for which \( H(z) \) exists. The transfer function \( H(z) \) exists if it is less than infinity. This implies that the R.O.C. cannot have poles.

Problem 11:

Explain why a causal FIR filter is always stable.

BIBO Stability:

\[
\sum_{k=-\infty}^{\infty} h[k] = \sum_{k=-\infty}^{\infty} h[k] x[n-k], \quad n \in \mathbb{Z}
\]

Bounded input implies that \( |x[n]| \leq M_x, \ n \in \mathbb{Z} \), where \( M_x \) is a positive number no matter how large.
We want \( |y[n]| \leq M_y, \ n \in \mathbb{Z} \), where \( M_y \) is a positive number.

\[
|y[n]| = \left| \sum_{k=-\infty}^{\infty} h[k] x[n-k] \right| \leq \sum_{k=-\infty}^{\infty} |h[k]| |x[n-k]| \leq M_x \sum_{k=-\infty}^{\infty} |h[k]|
\]

Given that the input \( x[n] \) is bounded, the output \( y[n] \) will always be bounded provided that \( \sum_{k=-\infty}^{\infty} |h[k]| < \infty \)

We know that

\[
H(\omega) = H(z) \bigg|_{z = e^{j\omega}} \rightarrow (\text{This evaluation on the unit circle})
\]
A necessary and sufficient condition for the Z-transform of an arbitrary signal, say $x[n]$, to exist is that

$$\sum_{n=-\infty}^{\infty} |x[n]z^{-n}| < \infty$$

If $\sum_{n=-\infty}^{\infty} |h[n]z^{-n}| < \infty$ then $H(z)$ exists at $z$.

Let $z = e^{+j\omega}$,

$$\sum_{n=-\infty}^{\infty} |h[n]z^{-n}| < \infty, \quad \sum_{n=-\infty}^{\infty} |h[n]|e^{-|j\omega|} = \sum_{n=-\infty}^{\infty} |h[z]| < \infty$$

**Observation:**

If a filter is causal, in order to be BIBO stable, all its poles must be inside the unit circle.

**Analog to Digital Conversion Techniques:**

**ROC is** $|z| > |a|$
Time - Frequency System Analysis:

The signal $x_m(t)$ is a bandlimited signal with maximum frequency content equal to $B = f_{\text{max}}$. $B$ is called the bandwidth of the signal $x_m(t)$:

For the ideal sampler:

Since $s(t)$ is a periodic signal, with fundamental period of duration $T_s$, we can represent this signal in terms of complex exponential Fourier series:

$$s(t) = \sum_{n=-\infty}^{\infty} C_n e^{j2\pi nf_{\text{max}}t}$$

where $f_0 = \frac{1}{T_s} = F_s$; $n \in \mathbb{Z}$

$$C_n = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} s(t) e^{-j2\pi nf_{\text{max}}t} dt$$

$$C_n = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} \delta(t) e^{-j2\pi nf_{\text{max}}t} dt$$
\[
\begin{align*}
    s(t) &= \sum_{n=-\infty}^{\infty} F_s e^{j2\pi n F_s t} \\
    s(f) &= \Im\left\{ \sum_{n=-\infty}^{\infty} F_s e^{j2\pi n F_s t} \right\} = \sum_{n=-\infty}^{\infty} F_s \Im\left\{ e^{j2\pi n F_s t} \right\} \\
    \therefore \ s(f) &= \sum_{n=-\infty}^{\infty} F_s \delta(f-nF_s) \\
\end{align*}
\]

Let

\[ X_m(f) = \Im\{x_m(t)\} \]

So we have,

\[ \Im\{x(t)\} = X_s(f) = \Im\{x_m(t) \cdot s(t)\} = \Im\{x_m(t)\} \cdot \Im\{s(t)\} = X_m(f) \cdot S(f) \]

\[ X_s(f) = X_m(f) \cdot S(f) = X_m(f) \cdot \left( \sum_{n=-\infty}^{\infty} F_s \delta(f-nF_s) \right) = \sum_{n=-\infty}^{\infty} F_s X_m(f) \cdot \delta(f-nF_s) \]

We know that

\[ X_m(f) \cdot \delta(f-nF_s) = \int_{-\infty}^{\infty} X_m(\lambda) \delta(f-nF_s-\lambda)d\lambda = X_m(f-nF_s) \]

\[ \therefore X_s(f) = \sum_{n=-\infty}^{\infty} F_s X_m(f-nF_s) \]
Example:

Where \((F_s - f_{max}) \geq f_{max}\).

Nyquist Theorem:

A signal \(x_m(t)\) can be recovered from its samples, \(x_m(nT_s)\), \(n \in \mathbb{Z}\), if the signal \(x_m(t)\) is bandlimited, with bandwidth \(B = f_{max}\), and the sampling frequency satisfies the condition: \(F_s \geq 2B\) or \(F_s \geq 2f_{max}\):

We want \(F_s - f_{max} \geq f_{max}\) or \(F_s \geq 2f_{max}\).

Analog to Digital Conversion:

**PAM Transmitter**

**Example**:

\[ x_m(t) \xrightarrow{\text{PAM Transmitter}} x_g(t) \xrightarrow{\text{Uniform Quantizer}} x_q(t) \xrightarrow{\text{Coder}} x_b(t) \]

PAM Transmitter
\[ h(t) = g_\tau(t) = \begin{cases} 1, & 0 \leq t \leq \tau \\ 0, & \text{otherwise} \end{cases} \]

\[
x_s(t) = x_m(t) \left( \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \right) = \sum_{n=-\infty}^{\infty} x_m(nT_s) \delta(t - nT_s) \]

\[
x_g(t) = x_s(t) \ast g_\tau(t) = \int_{-\infty}^{\infty} x_s(\lambda) g_\tau(t - \lambda) d\lambda = \int_{-\infty}^{\infty} \left( \sum_{n=-\infty}^{\infty} x_m(nT_s) \delta(\lambda - nT_s) \right) g_\tau(t - \lambda) d\lambda 
\]

\[
x_g(t) = \sum_{n=-\infty}^{\infty} x_m(nT_s) \int_{T_s(t - nT_s)}^{\infty} \delta(\lambda - nT_s) g_\tau(t - \lambda) d\lambda
\]

\[
x_g(t) = \sum_{n=-\infty}^{\infty} x_m(nT_s) g_\tau(t - nT_s)
\]
\[ x_g(t) = \sum_{n=-\infty}^{\infty} x_m(nT_s) g_\tau(t - nT_s) \]

\[ x_g(t) = \ldots x_m(0) g_\tau(t) + x_m(T_s) g_\tau(t - T_s) + x_m(2T_s) g_\tau(t - 2T_s) + \ldots \]

**Zero Order Hold Filter:**

We get the zero-order-hold filter when \( \tau = T_s \).

**Rules for Designing a Uniform Quantizer:**

1) Obtain the maximum values among the samples of \( x_g(t) (x_g(nT_s), \ n \in \mathbb{Z}) \)

We call this maximum \( \max \{x_g(t)\} \).

2) Obtain the minimum value among the samples of \( x_g(t) (x_g(nT_s), \ n \in \mathbb{Z}). \)

We call this minimum \( \min \{x_g(t)\} \).

3) Determine the number of levels or possible outputs permitted to the quantizer. We call this number of levels \( L \).
4) Compute the quantization step $\Delta$

$$\Delta = \frac{\max\{x_g(t)\} - \min\{x_g(t)\}}{L}.$$ 

5) Determine or identify the “quantization levels” starting always at $L_1 = \min\{x_g(t)\}$.

Each quantization level is obtained using the recurrence formula $L_{n+1} = L_n + \Delta$;

$$n = 1:1:L, \quad L_1 = \min\{x_g(t)\}.$$ 

6) Determine or identify the “quantization values”, known here as the values $v_n$, as the mid-points between the quantization levels. We get the quantization values using the recurrence formula

$$v_n = L_n + \frac{\Delta}{2}; \quad n = 1:1:L.$$ 

**Filter Design: First-order**

$$h[n] = b_0 a^n u[n]$$

**FIR**

$$h_D[n] = \begin{cases} h[n], & n \in Z_N \\ 0, & \text{otherwise} \end{cases}$$

**FIR Filter Design: Windowing Technique**

Given the DTFT $X(\omega)$ of an arbitrary signal $x[n]$, the signal can be recovered from its spectrum using the following formula for inverse DTFT:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{jn\omega} d\omega; \quad n \in Z$$
If the signal $X(\omega)$ is the frequency response of a filter, then $X(\omega) = H(\omega)$.

The impulse response is then obtained from the frequency response as follows:

$$h[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(\omega) e^{j\omega n} d\omega, \quad n \in \mathbb{Z}$$

$$h : \mathbb{Z} \rightarrow \mathbb{C}$$

Low-pass FIR Filter Design:

1. Select an ideal filter with a prescribed frequency response.
2. Take the inverse DTFT to obtain an infinite response.
3. Multiply in the time domain by a window with the desired order or length. Allow this first window to be rectangular.
4. Multiply the result of part 3 by a new window to improve the desired frequency response.

Problem 12:

Design an FIR low-pass filter of length or order equal to $N$ and frequency response (digital frequency) with cut-off $\omega_c$.

Solution:

1. 

2. 

$$h_L[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_L(\omega) e^{j\omega n} d\omega$$

$$h_L[n] = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega$$

$$h_L[n] = \frac{1}{2\pi} e^{j\omega n} \bigg|_{-\omega_c}^{\omega_c}$$

$$h_L[n] = \frac{1}{\pi n} \left[ \frac{e^{j\omega_c n} - e^{-j\omega_c n}}{2j} \right]$$

$$h_L[n] = \frac{1}{\pi n} \left[ \frac{e^{j\omega n} - e^{-j\omega n}}{2j} \right]$$
\[ h_L[n] = \frac{\omega}{\pi} \sin c[\omega, n]; \quad \sin c_\theta = \frac{\sin \theta}{\theta} \]

3. To obtain an FIR filter \( h_D[n] \), we proceed as follows

\[ h_D[n] = h_L[n] \cdot v_R[n] \]

**Circular Continuous-Frequency Convolution Theorem**

\[
DTFT\{h_D[n]\} = H_D(\omega) = DTFT\{h_L[n] \cdot v_R[n]\}
\]

\[
H_D(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_L(\lambda) V_R(\omega - \lambda) d\lambda
\]

\[
H_D(\omega) = H_L(\omega) \otimes V_R(\omega)
\]
Infinite Impulse Response (IIR) Filter Design:

There are many techniques for IIR filter design. We will concentrate on two very important techniques: Impulse Response Invariant and Analog Filter to Digital Filter transformation.

Properties of the Z - transform.

\[ Z\{x[n]\} = X(z) \]

Let \( g[n] = x[n - n_0] \)

Let \( x[n] = \delta[n] \)

\[ Z\{g[n]\} = G(z) \]

\[ G(z) = \sum_{n=-\infty}^{\infty} g[n]z^{-n} \]

\[ G(z) = \sum_{n=-\infty}^{\infty} x[n-n_0]z^{-n} \]

Let \( m = n - n_0;\quad n = m + n_0 \)

\[ G(z) = \sum_{m=-\infty}^{\infty} x[m]z^{-(m+n_0)} \]

\[ G(z) = z^{-n_0} \sum_{m=-\infty}^{\infty} x[m]z^{-m} \]

\[ G(z) = z^{-n_0}X(z) \]
Problem 13:

Obtain a Block Diagram Representation of the system below using Z-Transform

Solution:

\[ y[n] = b_0 x[n] + a_1 y[n-1] \]
\[ Z\{y[n]\} = Z\{b_0 x[n] + a_1 y[n-1]\} \]

\[ Y(z) = b_0 Z\{x[n]\} + a_1 Z\{y[n-1]\} \]
\[ Y(z) = b_0 X(z) + a_1 Z\{y[n-1]\} \]

We know in general that

\[ Z\{y[n-n_0]\} = z^{-n_0} Y(z) \]

Let \( n_0 = 1 \)

\[ Z\{y[n-1]\} = z^{-1} Y(z) \]

Then

\[ Y(z) = b_0 X(z) + a_1 z^{-1} Y(z) \]

We know:

\[ Y(z) = X(z) \cdot H(z) \]

or

\[ H(z) = \frac{Y(z)}{X(z)} \]

\[ Y(z) - a_1 z^{-1} Y(z) = b_0 X(z) \]
\[ (1 - a_1 z^{-1}) Y(z) = b_0 X(z) \]

\[ H(z) = \frac{b_0}{1 - a_1 z^{-1}} = \frac{Y(z)}{X(z)} \]
Block Diagram Representation:

\[ Y(z) = b_0 X(z) + a_1 z^{-1} Y(z) \]

IIR Filter Design:

Second order Filter Analysis:

General filter of the form

\[ y[n] + a_1 y[n-1] + a_2 y[n-2] = b_1 x[n] \]

We want to obtain

\[ H(z) = \frac{Y(z)}{X(z)} \]

\[ y[n] = b_1 x[n] - a_1 y[n-1] - a_2 y[n-2] \]

\[ Z\{y[n]\} = Z\{b_1 x[n] - a_1 y[n-1] - a_2 y[n-2]\} \]

\[ = b_1 Z\{x[n]\} - a_1 Z\{y[n-1]\} - a_2 Z\{y[n-2]\} \]

\[ Z\{y[n] + a_1 y[n-1] + a_2 y[n-2]\} = b_1 X(z) \]

\[ H(z) = \frac{b_1}{1 + a_1 z^{-1} + a_2 z^{-2}} \]

\[ H(z) = \frac{z^2 b_1}{z^2 + a_1 z + a_2} \]

Poles of \( H(z) \) \( \Rightarrow P_{1,2} = -\frac{1}{2} a_1 \pm \frac{1}{2} \sqrt{a_1^2 - 4 a_2} \)
1) If \( a_i^2 \geq 4a_2 \), the poles are on the real line.

2) If \( a_i^2 < 4a_2 \), \( P_{1,2} = -\frac{1}{2}a_i + j\frac{1}{2}\sqrt{4a_2 - a_i^2} \);
\[ a_i^2 - 4a_2 \rightarrow -(4a_2 - a_i^2) \]

\[ |H(s)|^2 = H(s) \cdot H^*(s) \]

\[ L^{-1}\{H(s)\} = h(t) \]

**FIR Filter Implementation using the Z-transform**

The FIR filters are also called non-recursive filters or transversal filters.

\[ y[n] = \sum_{k=0}^{M-1} x[k]h[n-k] \]

\[ y[n] = \sum_{k=0}^{M-1} h[k]x[n-k] \]

The filter in this case is of order M!

Let \( b[k] = h[k] \)

\[ y[n] = \sum_{k=0}^{M-1} b[k]x[n-k] \]

Taking the Z – transform of this equation, we get

\[ Y(z) = Z\{\sum_{k=0}^{M-1} b[k]x[n-k]\} = \sum_{k=0}^{M-1} b[k]Z\{x[n-k]\} = \left(\sum_{k=0}^{M-1} b[k]z^{-k}\right)X(z) \]

\[ H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{M-1} b[k]z^{-k} \]
Fast Fourier Transform:
It is an algorithm to compute the discrete Fourier transform in an efficient manner. There are many fast Fourier transform algorithms. We will concentrate on the algorithms designed by John Tukey and James Cooley in 1965 and are commonly known as Cooley – Tukey FFT algorithms.

Cooley – Tukey FFT algorithms:
The objective is to develop an efficient algorithm to compute the matrix-vector operation:
\[ X = f_n x \]
The direct computation of this matrix-vector operation required \( N^2 \) multiplications and \( N(N-1) \) additions.

Example: \( N = 4 \)

\[
X = f_4 x = \begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & w_4 & w_4^2 & w_4^3 \\
1 & w_4^2 & 1 & w_4^3 \\
1 & w_4^3 & w_4^2 & w_4 \\
\end{bmatrix}
\begin{bmatrix}
x[0] \\
x[1] \\
x[2] \\
x[3] \\
\end{bmatrix} = \begin{bmatrix}
x[0] \\
x[1] \\
x[2] \\
x[3] \\
\end{bmatrix}
\]

\( w_4 = e^{-\frac{2\pi}{4}} \)

\( w_4^6 = e^{-\frac{2\pi}{4}} \cdot e^{-\frac{2\pi}{4}} = e^{-\frac{2\pi}{4}} \cdot e^{-\frac{2\pi}{4}} = e^{-\frac{2\pi}{2}} = 1 \)
For $N = 2^M$, a power of 2, the Cooley-Tukey algorithm reduces the number of multiplications to $N \log_2 N$.

Example:

<table>
<thead>
<tr>
<th>$N$</th>
<th>Direct Method</th>
<th>Cooley-Tukey Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>$(1024)^2$</td>
<td>$1024 \log_2 1024 = (10)1024$</td>
</tr>
</tbody>
</table>

**Cooley-Tukey Algorithm Technique:**

Additive property of the DFT:

Example: $N = 4$

$$X = F_4 x = F_4 x_e + F_4 x_0$$

1. We will represent $x$ as a sum of two vectors: $x[n] = x_e[n] + x_0[n]$, $n \in \mathbb{Z}_4$

$$x = \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} = \begin{bmatrix} 0 \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} + \begin{bmatrix} x[0] \\ 0 \\ 0 \\ x[3] \end{bmatrix} = x_e + x_0$$

2. We will use the linearity property of the DFT $F_4 x = F_4 (x_e + x_0) = F_4 x_e + F_4 x_0$

$$F_4 x_e = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w_4 & w_4^2 & w_4^3 \\ 1 & w_4^2 & 1 & w_4^2 \\ 1 & w_4^3 & w_4^2 & w_4 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} = \begin{bmatrix} x[0] + x[2] \\ x[0] + w_4 x[2] \\ x[0] + x[2] \\ x[0] + w_4^2 x[2] \end{bmatrix}$$
\[ w_4^2 = e^{-\frac{2\pi}{4}} = e^{-j\pi} = \cos \pi - j \sin \pi = -1 \]

\[
\begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix} \begin{bmatrix} x[0] \\ x[2] \end{bmatrix} = \begin{bmatrix} x[0] + x[2] \\ x[0] - x[2] \end{bmatrix}
\]

\[
F_2 = \left[ w_2^{Kn} \right]_{x \in \mathbb{Z}_2} = \begin{bmatrix} 1 & 1 \\ 1 & w_2 \end{bmatrix}
\]

\[ w_2 = e^{-\frac{2\pi}{2}} = -e^{-j\pi} = -1 \]

**Butterfly Block Diagram (Flow Diagram)**

Representation of the FFT:

\[
\begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix} \begin{bmatrix} x[0] \\ x[2] \end{bmatrix} = \begin{bmatrix} x[0] + x[2] \\ x[0] - x[2] \end{bmatrix}
\]

\[
F_4x_e = \begin{bmatrix} F_2 \\ F_2 \end{bmatrix} \begin{bmatrix} x[0] \\ x[2] \end{bmatrix}
\]

We want to compute

\[
F_4x = F_4x_e + F_4x_0
\]

16 multiplications
12 summations

1. \[ F_4x_e = \begin{bmatrix} F_2 \\ F_2 \end{bmatrix} \begin{bmatrix} x[0] \\ x[2] \end{bmatrix} = F_4 \begin{bmatrix} x[0] \\ 0 \\ x[2] \\ 0 \end{bmatrix} \]
2. \[ F_4 x_0 = F_4 \begin{bmatrix} 0 \\ x[1] \\ 0 \\ x[3] \end{bmatrix} \]

In general, we want to know

\[
\text{DFT} \left\{ \frac{x[n - n_0]}{g[n]} \right\} = ?
\]

\[
\text{DFT} \left\{ x[n - n_0] \right\} = \sum_{n=0}^{N-1} x[n-n_0] w_n^{Kn} \\
= w_n^{Kn} \sum_{m=n_0}^{m=N-n_0} x[m] w_n^{Km}
\]

**Example:**

- \( \tilde{x}[n] \)
- Fundamental Period
- \( n = N - 1 \)
- \( x[n] \)
- \( -n_0 \)
- \( N - 1 \)
- \( \cdots \)

- \( \tilde{x}[n] \)
- \( 1 \)
- \( 2 \)
- \( 3 \)
- \( -1 \)
- \( -3 \)
- \( 8 \)
- \( \cdots \)
Remainder \( \left( \frac{P}{N} \right) = \langle p \rangle_N = \langle p + qN \rangle_N \)

\( \langle -3 \rangle_4 = \langle -3 + 4 \rangle_4 = \langle -3 + 4 \rangle_4 = \langle 1 \rangle_4 = 1 \)

\( x[-3] \leftrightarrow x[1] \)

\[ G[K] = W_N^{Kn_0} \left( \sum_{m=0}^{N-1} x[n] \cdot W_N^{Km} \right) \]

Hadamard product

DFT \( \{x[n-n_0]\} = W_N^{Kn_0} \cdot X[K] \)

\[ W_N^{Kn_0} = e^{-j2\pi Kn_0/N} \]

Homework:

Express \( F_4x_0 \) in matrix form.

\[ G[k] \rightarrow \text{long. } N \]

\[ G[k] = W_N^{Kn_0} \cdot X[k] \]

\[ G[k] = W_N^{Kn_0} \cdot (F_4x) \]

\[ F_4x_0 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w_4 & w_4^2 & w_4^3 \\ 1 & w_4^2 & 1 & w_4^3 \\ 1 & w_4^3 & w_4^2 & w_4 \end{bmatrix} \begin{bmatrix} 0 \\ x[3] \\ 0 \\ x[3] \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & w_4 & 0 & w_4^3 \\ 0 & w_4^2 & 0 & w_4^2 \\ 0 & w_4^3 & 0 & w_4 \end{bmatrix} \begin{bmatrix} 0 \\ x[3] \\ 0 \\ x[3] \end{bmatrix} \]
Compacting, we get

\[
F_4 x_0 = \begin{bmatrix}
1 & 1 \\
w_4 & w_4^3 \\
w_4^2 & w_4^2 \\
w_4^3 & w_4
\end{bmatrix}
\begin{bmatrix}
x[1] \\
x[3]
\end{bmatrix}
= \begin{bmatrix}
w_4 x[1] + w_4^3 x[3] \\
w_4^2 x[1] + w_4^2 x[3] \\
w_4^3 x[1] + w_4 x[3]
\end{bmatrix}
\]

We know that

\[
\text{DFT}_N \{x[n \mod N]\} = W_N^{kn_0} \cdot X[K]
\]

Example: \(N = 4\), \(x[n] = \{x[0], x[1], x[2], x[3]\}\)

\[
y[n] = x[n \mod N] \;
\]
\[
y[n] = x[n - 2 \mod 4] \;
\]
\[
y[0] = x[0 - 2 \mod 4] = x[2]
\]
\[
\]
\[
y[2] = x[2 - 2 \mod 4] = x[0]
\]
\[
\]

\[
<p >_N = <p + qN >_N
\]

\[
<p + qN >_N = \text{Remainder} \left( \frac{P + qN}{N} \right) =
\]

\[
\text{Remainder} \left( \frac{P}{N} \right) + \text{Remainder} \left( \frac{qN}{N} \right)
\]

\[
<1>_4 = 1
\]
\[
<5>_4 = <1 + 4>_4 = <1>_4 + <4>_4
\]
\[
<9>_4 = <1 + 2\cdot4>_4 = <1>_4 + <8>_4
\]
\[
<21>_4 = <1 + 5\cdot4>_4 = <1>_4 + <20>_4
\]
\[ < -21 >_{n_0} = < -21 + 2 \cdot 11 > = 1 \]

\[ y[n] = \{x[2], x[3], x[0], x[1]\} \]

\[
\begin{bmatrix}
  x[0] \\
  x[1] \\
  x[2] \\
  x[3]
\end{bmatrix} \rightarrow \begin{bmatrix}
  x[3] \\
  x[0] \\
  x[1] \\
  x[2]
\end{bmatrix} \rightarrow \begin{bmatrix}
  x[2] \\
  x[3] \\
  x[0] \\
  x[1]
\end{bmatrix} \rightarrow \begin{bmatrix}
  x[1] \\
  x[2] \\
  x[3] \\
  x[0]
\end{bmatrix}
\]

\[ F_4 \begin{bmatrix}
  x[1] \\
  0 \\
  x[3] \\
  0
\end{bmatrix} = \begin{bmatrix} F_2 \\ F_2 \end{bmatrix} \cdot \begin{bmatrix}
  x[1] \\
  x[3]
\end{bmatrix} = DFT_4 \{s[n]\} = S[K] \]

We want

\[ F_4 \begin{bmatrix}
  0 \\
  x[1] \\
  0 \\
  x[3]
\end{bmatrix} = DFT_4 \{s[n - n_0]\}; n_0 = 1 \]

\[ F_4 \begin{bmatrix}
  0 \\
  x[1] \\
  0 \\
  x[3]
\end{bmatrix} = W_{4}^{k_{n_0}} \cdot S[k]; k \in Z_4 \]

If \( n_0 = 1 \)

\[ w_{4}^{k_{n_0}} = \begin{bmatrix}
  1 \\
  w_4 \\
  w_4^2 \\
  w_4^3
\end{bmatrix} \]
\[ F_4 \begin{bmatrix} 0 \\ x[1] \\ 0 \\ x[3] \end{bmatrix} = \begin{bmatrix} 1 \\ w_4 \\ w_4^2 \\ w_4^3 \end{bmatrix} \cdot \begin{bmatrix} s[0] \\ S[1] \\ s[2] \\ s[3] \end{bmatrix} \]

\[ \therefore F_4 \begin{bmatrix} 0 \\ x[1] \\ 0 \\ x[3] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & w_4 & 0 & 0 \\ 0 & 0 & w_4^2 & 0 \\ 0 & 0 & 0 & w_4^3 \end{bmatrix} \cdot F_2 \begin{bmatrix} x[1] \\ x[3] \end{bmatrix} \]

Remember:

\[ F_2 = \begin{bmatrix} 1 & 1 \\ 1 & w_4^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \]

\[ \begin{bmatrix} 1 \\ w_4 \\ w_4^3 \\ w_4^2 \\ w_4^3 \end{bmatrix} \begin{bmatrix} x[1] \\ x[3] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & w_4 & 0 & 0 & w_4^2 \\ 0 & 0 & w_4^2 & 0 & 1 \\ 0 & 0 & 0 & w_4^3 & 1 \end{bmatrix} \begin{bmatrix} x[1] \\ x[3] \end{bmatrix} \]

\[ F_4 x = F_4 x_e + F_4 x_0 \]