Digital Signal Analysis And Processing

Surendra K.C.

7. Discrete Fourier Transform

7.1 Discrete Fourier transform (DFT) representation, Properties of DFT (Linearity, Time Shift, Frequency Shift, Duality, Convolution, Multiplication) Circular convolution

7.2 Fast Fourier Transform (FFT) algorithm (decimation in time algorithm, decimation in frequency alforithm)

7.3 Computational complexity of FFT algorithm.

Introduction

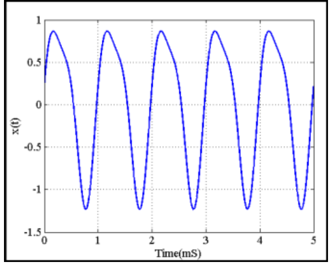
- The discrete time Fourier transform (DTFT) provides a method to represent a discrete time signal in frequency domain and to perform frequency analysis of discrete time signal.
- The drawback in DTFT is that the frequency domain representation of a discrete time signal obtained using DTFT will be a continuous function of ω and so it cannot be processed by digital system.
- The discrete Fourier transform (DTF) has been developed to convert a continuous function of ω to a discrete function of ω, so that frequency analysis of discrete time signals can be performed on a digital systm.
- Basically, the DFT of a discrete time signal is obtained by sampling DTFT of the signal at uniform frequency intervals and the number of samples should be sufficient to avoid aliasing of frequency spectrum.
- The drawback in DFT is that, the computation of each sample of DFT involves a large number of calculations.

Discrete Fourier Transform (DFT)

- The DFT is itself a sequence rather than a function of a continuous variable, and it corresponds to samples, equally spaced in frequency, of a Fourier representation of the signal.
- The DFT plays a central role in the implementation of a variety of digital signal processing algorithms.
- The sample of DTFT are represented as a function of integer k, and so the DFT is a sequence of complex numbers represented as $X(k)$ for $k = 0,1,2,3,\ldots$
- Since $X(k)$ is a sequence consisting complex numbers, the magintude and phase of each sample can be computed and listed as magnitude sequence and phase sequence respectively.
- The plot of magintude versus k is called magintude spectrum and the plot of phase versus k is called phase spectrum.
- in general these plots are called frequency spectrum.

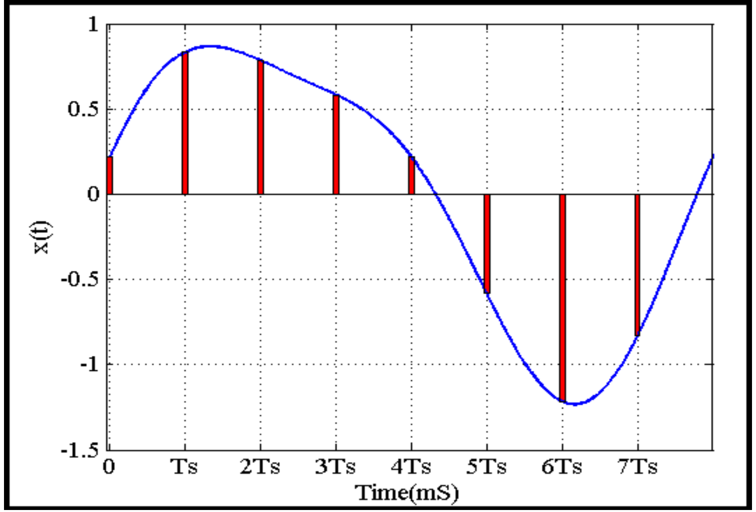
	Time	Frequency
Fourier transform (FT)	continuous	continuous
Fourier series (FS)	continuous periodic	discrete
Discrete-time Fourier transform (DTFT)	discrete	continuous periodic
Discrete Fourier series (DFS)	discrete periodic	discrete, periodic
Discrete Fourier transform (DFT)	discrete finite	discrete finite

Why Discrete Fourier Transform



- ✓ Assume that $x(t)$ as shown in figure, is a continuous-time signal that we need to analyze.
- ✓ Obviously, a digital computer cannot work with a contineous time signal and we need to take some samples of $x(t)$ and analyse these samples instead of the original signal.
- ➢ Moreover, the figure shows only the first 5 milisecond of the signal $x(t)$ that may continue for hours, months or more.
- ➢ Since our digital computer can process only a finite number of samples, we have to make an approximation and use a limited number of samples.
- ➢ Therefore, generally, a finite duration sequence is utilized to represent this analog continuous time signal which may extend to positive infintiy on the time axis.

- Assume that we sample $x(t)$ as in figure with a sampling rate of 8000 samples/second and take only L=8 sampes of the signal.
- We obtain the discrete sequence $x(n)$ as follows:



n	0	1	2	3	4	5	6	7
x(n)	0.21	0.83	0.78	0.58	0.21	-0.58	-1.2	-0.83

- Fourier analysis provides several mathemetical tools for detetmining the frequency content of a time-domain signal, but there are only two technique from Fourier analysis family which target discrete-time signals: Discrete time Fourier Transform (DTFT) and Discrete Fourier Transform (DFT).

- The DTFT of an input sequence, $x(n)$ is given by

$$X\left(e^{j\omega}\right) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$$

- We can use this equation to find the spectrum of finite duration signal $x(n)$; however, $X\left(e^{j\omega}\right)$ given by the above equation is a continuous function of ω.
- Hence, a digital computer cannot directly use this equation to analyse $x(n)$.
- However, we can use samples of $X\left(e^{j\omega}\right)$ to find an approximation of the spectrum of $x(n)$.
- The idea of smapling of $X\left(e^{j\omega}\right)$ at equally spaced frequency point is in fact the basis of the second Fourier technique i.e., the DFT.
- This sampling is taking place in the frequency domain i.e., $X\left(e^{j\omega}\right)$ is a function of frequency.

Discrete Fourier Transform (DFT) of Discrete Time Signal

- The frequency domain representation of a discrete time signal obtained using discrete time Fourier Transform (DTFT) will be a continuous and periodic function of ω, with periodicity of 2π.
- In order to obtain discrete function of ω, the DTFT can be sampled at sufficient number of frequency intervals.
- Let $X\left(e^{j\omega}\right)$ be discrete time Fourier transform of the discrete time signal $x(n)$. The discrete Fourier transform (DFT) of $x(n)$ is obtained by sampling one period of the discrete time Fourier transform $X\left(e^{j\omega}\right)$ at a finite number of frequency points.
- The frequency domain sampling is conventionally performed at N equal spaced frequency points in the period, $0 \leq \omega \leq 2\pi$. The sampling frequency points are denoted as ω_k and they are given by,

$$\omega_k = \frac{2\pi k}{N}; \quad for\ k = 0, 1, 2, \dots, N-1$$

- Now the DFT is a sequence consisting of N-samples of DTFT. Let the samples are denoted by $X(k)$ for $k = 0, 1, 2, \dots N-1$.

- Therefore, the sampling of $X\left(e^{j\omega}\right)$ is mathematically expressed as,

$$X(k) = \mathrm{X}\left(e^{j\omega}\right)\Big|_{\omega_k=\frac{2\pi k}{N}}; \quad \text{for k} = 0, 1, 2, \ldots, \mathrm{N} - 1$$

- The DFT sequence starts at $k = 0$, corresponding to $\omega = 0$ but does not include $k = N$, corresponding to $\omega = 2\pi$, (since the sample at $\omega = 0$ is same at $\omega = 2\pi$).
- Generally, the DFT is defined along the number of samples and is called N-point DFT.
- The number of samples N for a finite duration sequence $x(n)$ of length L should be such that, $N \geq L$, in order to avoid aliasing of frequency spectrum.
- To calculate DFT of a sequence it is not necessary to compute Fourier transform, since the DFT can be directly computed using the DFT definition.

Frequency Domain Sampling: The Discrete Fourier Transform

We consider the sampling of the Fourier transform of an aperiodic discrete time sequence and establish the relationship between the sample Fourier transform and the DFT.

Frequency-Domain Sampling and Reconstruction of Discrete-Time Signals

- Aperiodic finite-energy signals have continuous spectra.
- Let us consider an aperiodic discrete time signal $x(n)$ with Fourier transform

$$X\left(e^{j\omega}\right) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}$$

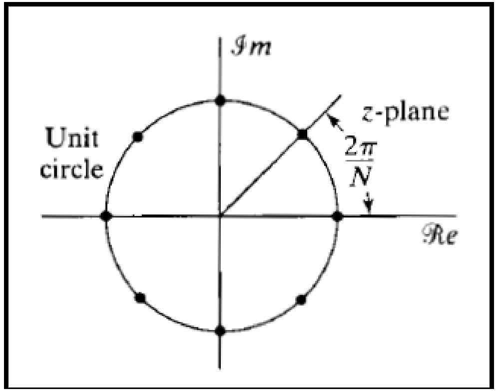
Now assume $X(k)$ is obtained by sampling $X(e^{j\omega})$ at frequencies $\omega_k = \frac{2\pi}{N}k$; i.e.,

$$\tilde{X}(k) = \mathrm{X}(e^{j\omega})\Big|_{\omega=\frac{2\pi k}{N}} = \mathrm{X}(e^{j\frac{2\pi k}{N}})$$

Since the Fourier transform is periodic in ω with period 2π, the resulting sequence is periodic in k with period N.

Also, since the Fourier transform is equal to the z-transform evaluated on the unit circle, it follows that $X(k)$ can also be obtained by sampling $X(z)$ at N equally spaced points on the unit circle. Thus,

$$\tilde{X}(k) = \mathrm{X}(z)\Big|_{z=\frac{2\pi k}{N}} = \mathrm{X}(e^{j\frac{2\pi k}{N}})$$



These sampling points are shown in figure for $N = 8$.

This figure shows that the sequence of samples is periodic, since the N points are equally spaced starting with zero angle. Therefore, the same sequence repeats as k varies outside the range $0 \leq k \leq N-1$.

The sequence of samples $X(k)$, being periodic with period N, could be the sequence of discrete Fourier series coefficients of a sequence $x(n)$. To obtain that sequence, we can simply substitute $X(k)$ obtained as

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

And the Fourier transform,

$$X\left(e^{j\omega}\right) = \sum_{m=-\infty}^{\infty} x(m)\, e^{-j\omega n}$$

Thus,

$$\tilde{x}(n) = \frac{1}{N}\sum_{k=0}^{N-1}[\sum_{m=-\infty}^{\infty} x(m)e^{-j2\pi k^{m}/_{N}}]e^{j2\pi k^{n}/_{N}}$$

Interchanging the order of summation

$$\tilde{x}(n) = \sum_{n=-\infty}^{\infty} x(m)[\frac{1}{N}\sum_{k=0}^{N-1} e^{j2\pi k^{(n-m)}/_{N}}]$$

$$= \sum_{m=-\infty}^{\infty} x(m)\tilde{p}(n-m)$$

Where,

$$\tilde{p}(n-m) = \frac{1}{N}\sum_{k=0}^{N-1} e^{j2\pi k^{(n-m)}/_{N}} = \sum_{r=-\infty}^{\infty} \delta[n-m-rN]$$

$$\tilde{x}(n) = \sum_{m=-\infty}^{\infty} x(m) \sum_{r=-\infty}^{\infty} \delta[n-m-rN]$$

$$= \sum_{m=-\infty}^{\infty} x(m) \sum_{r=-\infty}^{\infty} \delta[(n-rN)-m]$$

And therefore

$$\tilde{x}(n) = x(n) * \sum_{r=-\infty}^{\infty} \delta[n - rN] = \sum_{r=-\infty}^{\infty} x[n - rN]$$

$$\tilde{x}(n) = \sum_{r=-\infty}^{\infty} x[n - rN]$$

The finite length sequence $x(n)$ can be recovered from $\tilde{x}(n)$ as

$$x(n) = \begin{cases} \tilde{x}(n), & 0 \leq n \leq N-1 \\ 0, & otherwise \end{cases}$$

This can alternatively written as

$$\tilde{x}(n) = x[(n\ modulo\ N)] = x[\big((n)\big)_N]$$

Defination of Discrete Fourier Transform (DFT)

Let,$x(n)$ = Discrete time signal of length L

$X(n)$ = DFT of $x(n)$

Now, the N-point DFT of $x(n)$, where $\mathrm{N} \geq L$, is defined as,

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

For convenience, let

$$W_N = e^{-j\frac{2\pi}{N}}$$

Then,

$$X(k) = \sum_{n=0}^{N-1} x(n), W_N^{kn}\ , \qquad for\ k = 0,1,2,\ldots, N-1$$

Since $X(k)$ is a sequence consisting of N-complex numbers for $k = 0,1,2,\ldots N-1$, the DFT of $x(n)$ can be expressed as a sequence as shown below.

$$X(k) = \{X(0), X(1), X(2), \ldots \ldots, X(N-1)\}$$

Frequency Response Using DFT

The $X(k)$ is a discrete function of discrete time frequency ω, and so it is also called discrete frequency spectrum (or signal spectrum) of the discrete time signal $x(n)$.

The $X(k)$ is a complex valued function of k and so it can be expressed in rectangular form as,

$$X(k) = X_r(k) + jX_i(k)$$

Where, $X_r(k)$ is the real part of $X(k)$ and $X_i(k)$ is the imaginary part of $X(k)$.

Now the magintude function (or magnitude spectrum) $|X(k)|$ is defined as,

$$|X(k)|^2 = X(k)X^*(k) = X_r^2(k) + X_i^2(k)$$

The phase function (or phase spectrum) $\measuredangle X(k) = \tan^{-1} \frac{X_i(k)}{X_r(k)}$

Since $X(k)$ is a sequence consisting of N-complex numbers for $k = 0, 1, 2, \dots N - 1$, the magnitude and phase spectrum of $X(k)$ can be expresses as a sequence as shown below.

Magintude sequence, $|X(k)| = \{|X(0)|, |X(1)|, |X(2)|, \dots |X(N-1)|\}$

Phase sequece, $\measuredangle X(k) = \{\measuredangle X(0), \measuredangle X(1), \measuredangle X(2), \dots \measuredangle X(N-1)\}$

The magnitude and phase sequence can be sketched graphically as a function of k.

The magnitude sequence versus k is called magnitude spectrum and the plot of samples of phase sequence verus k is called phase spectrum. In general these plots are called frequency spectrum.

Inverse DFT

Let, $x(n) =$ Discrete time signal

$X(k) = N$ $-$point DFT of $x(n)$

The inverse disctete Fourier transform of the sequence $X(k)$ of length N is defined as,

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

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k), W_N^{-kn}, \qquad for\, n = 0,1,2, \dots, N-1$$

Example 1: Compute 4-point DFT and 8-point DFT of causal three sample sequence given by,

$$x(n) = \begin{cases} \frac{1}{3}, & 0 \le n \le 2 \\ 0, & else \end{cases}$$

Also find DTFT of the sequence and compare it.

Solution: By definition of DFT, we have

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

For 4-point DFT (N=4)

$$X(k) = \sum_{n=0}^{3} x(n) e^{-j\frac{2\pi}{4}kn} == \sum_{n=0}^{2} x(n) e^{-j\frac{\pi}{2}kn} = x(0)e^{0} + x(1)e^{-j\frac{\pi}{2}k} + x(2)e^{-j\pi k}$$

$$= \frac{1}{3} + \frac{1}{3}e^{-j\frac{\pi}{2}k} + \frac{1}{3}e^{-j\pi k} = \frac{1}{3}\left[1 + \cos\frac{\pi k}{2} - j\sin\frac{\pi k}{2} + \cos\pi k - j\sin\pi k\right]$$

For 4-point, $X(k)$ has to be evaluated for k=0,1,2,3

When $\mathrm{k} = 0;\ X(0) = \frac{1}{3}[1 + \cos 0 - j\sin 0 + \cos 0 - j\sin 0] = 1 = 1\angle 0$

When $k = 1; X(1) = \frac{1}{3}\left[1 + \cos\frac{\pi}{2} - j\sin\frac{\pi}{2} + \cos\pi - j\sin\pi\right] = \frac{1}{3}[1 + 0 - j - 1 - j \times 0]$

$$= -j\frac{1}{3} = 0.333\angle - 0.5\pi$$

When $k = 2; X(2) = \frac{1}{3}[1 + \cos\pi - j\sin\pi + \cos 2\pi - j\sin 2\pi] = \frac{1}{3}[1 + 1 - j0 + 1 - j0]$

$$= \frac{1}{3} = 0.333\angle 0$$

When $k = 3; X(3) = \frac{1}{3}\left[1 + \cos\frac{3\pi}{2} - j\sin\frac{3\pi}{2} + \cos 3\pi - j\sin 3\pi\right] = \frac{1}{3}[1 + 0 + j - 1 - j \times 0]$

$$= j\frac{1}{3} = 0.333\angle 0.5\pi$$

The 4-point DFT sequence is given by

$$X(k) = \left(1, -j\frac{1}{3}, \frac{1}{3}, j\frac{1}{3}\right)$$

Magnitude Function, $|X(k)| = [1, 0.333.\, 0.333.\, 0.333]$

Phase Function, $\angle X(k) = [0, -0.5\pi, 0, 0.5\pi]$

For 8-point DFT (N=8)

$$X(k) = \sum_{n=0}^{7} x(n) e^{-j\frac{2\pi}{8}kn} == \sum_{n=0}^{2} x(n) e^{-j\frac{\pi}{4}kn} = x(0)e^{0} + x(1)e^{-j\frac{\pi}{4}k} + x(2)e^{-j\frac{\pi}{2}k}$$

$$= \frac{1}{3} + \frac{1}{3} e^{-j\frac{\pi}{4}k} + \frac{1}{3} e^{-j\frac{\pi}{2}k} = \frac{1}{3}\left[1 + \cos\frac{\pi}{4}k - j\sin\frac{\pi}{4}k + \cos\frac{\pi}{2}k - j\sin\frac{\pi}{2}k\right]$$

For 8-point, $X(k)$ has to be evaluated for k=0,1,2,3,4,5,6,7

When $\mathrm{k} = 0;\ X(0) = \frac{1}{3}[1 + \cos 0 - j\sin 0 + \cos 0 - j\sin 0] = 1 = 1\angle 0$

When $k = 1; X(1) = \frac{1}{3}[1 + \cos\frac{\pi}{4} - j\sin\frac{\pi}{4} + \cos\frac{\pi}{2} - j\sin\frac{\pi}{2}]$

$$= \frac{1}{3}[1 + 0.707 - j0.707 + 0 - j] = 0.568 - \mathrm{j}0.568 = 0.803\angle - 0.25\pi$$

When $k = 2; X(2) = \frac{1}{3}[1 + \cos\frac{2\pi}{4} - j\sin\frac{2\pi}{4} + \cos\frac{2\pi}{2} - j\sin\frac{2\pi}{2}]$

$$= \frac{1}{3}[1 + 0 - j - 1 - j0] = -\mathrm{j}0.333 = 0.333\angle - 0.5\pi$$

$$\text{When } k = 3;\ X(3) = \frac{1}{3}\left[1 + \cos\frac{3\pi}{4} - j\sin\frac{3\pi}{4} + \cos\frac{3\pi}{2} - j\sin\frac{3\pi}{2}\right]$$

$$= 0.333\,(1 - 0.707 - j0.707 + 0 + j1)$$

$$= 0.098 + j0.098 = 0.139\angle -0.785 = 0.139\angle -0.25\pi$$

$$\text{When } k = 4;\ X(4) = \frac{1}{3}\left[1 + \cos\frac{4\pi}{4} - j\sin\frac{4\pi}{4} + \cos\frac{4\pi}{2} - j\sin\frac{4\pi}{2}\right]$$

$$= 0.333\,(1 - 1 - j0 + 1 - j0) = 0.333 = 0.333\angle 0$$

$$\text{When } k = 5;\ X(5) = \frac{1}{3}\left[1 + \cos\frac{5\pi}{4} - j\sin\frac{5\pi}{4} + \cos\frac{5\pi}{2} - j\sin\frac{5\pi}{2}\right]$$

$$= 0.333\,(1 - 0.707 + j0.707 + 0 - j1)$$

$$= 0.098 - j0.098 = 0.139\angle -0.785 = 0.139\angle -0.25\pi$$

$$\text{When } k = 6;\ X(6) = \frac{1}{3}\left[1 + \cos\frac{6\pi}{4} - j\sin\frac{6\pi}{4} + \cos\frac{6\pi}{2} - j\sin\frac{6\pi}{2}\right]$$

$$= 0.333\,(1 + 0 + j1 - 1 - j0)$$

$$= j0.333 = 0.333\angle\pi/2 = 0.333\angle 0.5\pi$$

$$\text{When } k = 7;\ X(7) = \frac{1}{3}\left[1 + \cos\frac{7\pi}{4} - j\sin\frac{7\pi}{4} + \cos\frac{7\pi}{2} - j\sin\frac{7\pi}{2}\right]$$

$$= 0.333\,(1 + 0.707 + j0.707 + 0 + j1)$$

$$= 0.568 + j0.568 = 0.803\angle 0.785 = 0.803\angle 0.25\pi$$

Phase angles are in radians

$\therefore$ The 8-point DFT sequence X(k) is given by,

$$X(k) = \{1\angle 0,\ 0.803\angle -0.25\pi,\ 0.333\angle -0.5\pi,\ 0.139\angle -0.25\pi,\ 0.333\angle 0,\ 0.139\angle 0.25\pi,\ 0.333\angle 0.5\pi,\ 0.803\angle 0.25\pi\}$$

$\therefore$ Magnitude Function, $|X(k)| = \{1,\ 0.803,\ 0.333,\ 0.139,\ 0.333,\ 0.139,\ 0.333,\ 0.803\}$

Phase Function, $\angle X(k) = \{0,\ -0.25\pi,\ -0.5\pi,\ -0.25\pi,\ 0,\ 0.25\pi,\ 0.5\pi,\ 0.25\pi\}$

Let, $X(e^{j\omega})$ be Fourier transform of $x(n)$.

Now by definition of Fourier transform,

$$X(e^{j\omega}) = \sum_{n=-\infty}^{+\infty} x(n)\, e^{-j\omega n} = \sum_{n=0}^{2} x(n)\, e^{-j\omega n}$$

$$= x(0)\, e^{0} + x(1)\, e^{-j\omega} + x(2)\, e^{-j2\omega} = \frac{1}{3} + \frac{1}{3} e^{-j\omega} + \frac{1}{3} e^{-j2\omega}$$

$$= \frac{1}{3} + \frac{1}{3}(\cos\omega - j\sin\omega) + \frac{1}{3}(\cos 2\omega - j\sin 2\omega)$$

$$= \frac{1}{3}(1 + \cos\omega + \cos 2\omega) - j\frac{1}{3}(\sin\omega + \sin 2\omega)$$

- The magnitude spectrum of $X(k)$ are shown as in figure for N=4, 8 and 16 respectively.
- The curve in dotted line is the sketch of Magnitude (left) and Phase (right) function of $X\left(e^{j\omega}\right)$ for ω in the range of 0 to 2π.
- Here's it is observed that the Magnitude and Phase of DFT coefficients are sampled of Magnitude and Phase function of $X\left(e^{j\omega}\right)$.

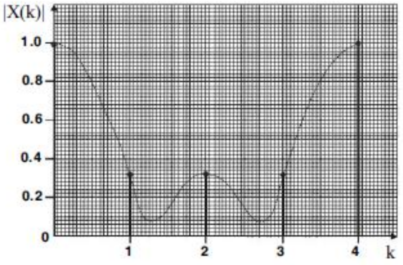


Fig 1 : Magnitude spectrum of X(k) for N=4.

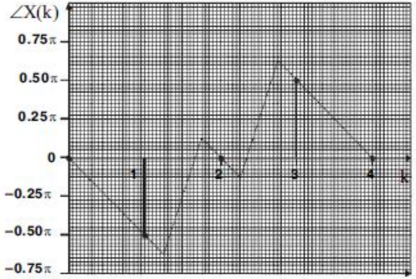


Fig 4 : Phase spectrum of X(k) for N=4.

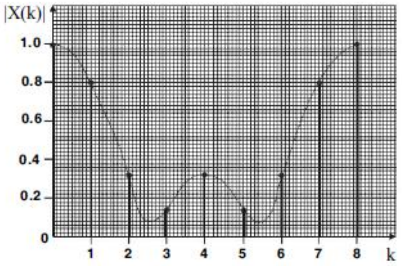


Fig 2 : Magnitude spectrum of X(k) for N=8.

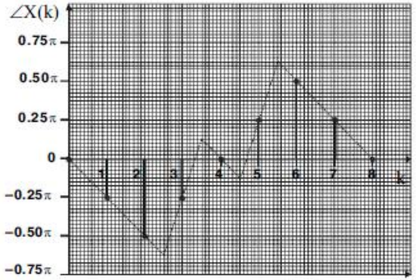


Fig 5 : Phase spectrum of X(k) for N=8.

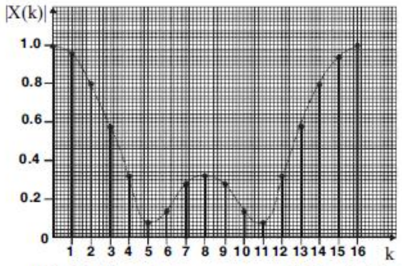


Fig 3 : Magnitude spectrum of X(k) for N=16.

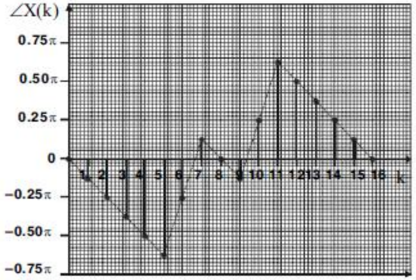


Fig 6 : Phase spectrum of X(k) for N=16.

The DFT as a Linear Transformation

The formulas for the DFT and IDFT can be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}\,, \qquad for\ k = 0,1,2,\ldots,N-1$$

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) W_N^{-kn}, \qquad for\ n = 0,1,2,\ldots,N-1$$

Where

$$W_N = e^{-j\frac{2\pi}{N}}$$

Which is an N^{th} root of unity.

The computation of each point of the DFT can be accomplished by N complex multiplication and $(N-1)$ complex additions.

Hence the N-point DFT values can be computed in a total of N^2 complex multiplcation and $N(N-1)$ complex additions.

Let us define an N-point vector $\boldsymbol{x_N}$ of the signal sequence $x(n), n = 0, 1, 2, \dots, N-1$, an N-point vector $\boldsymbol{X_N}$ of frequency samples, and an $N \times N$ matrix $\boldsymbol{W_N}$ as

$$\boldsymbol{x_N} = \begin{bmatrix} x(0) \\ x(1) \\ . \\ . \\ . \\ x(N-1) \end{bmatrix}, \qquad \boldsymbol{X_N} = \begin{bmatrix} X(0) \\ X(1) \\ . \\ . \\ . \\ X(N-1) \end{bmatrix},$$

$$\boldsymbol{W_N} = \begin{bmatrix} 1 & 1 & 1 & . & 1 \\ 1 & W_N & W_N^2 & . & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & . & W_N^{2(N-1)} \\ . & . & . & . & . \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & . & W_N^{(N-1)(N-1)} \end{bmatrix}$$

With this definitions, the N-point DFT may be expressed in matrix form as:

$$\boldsymbol{X_N} = \boldsymbol{W_N x_N}$$

Where $\boldsymbol{W_N}$ is the martix of the linear transformation and $\boldsymbol{W_N}$ is a symmetric matrix. If we assume that the inverse of $\boldsymbol{W_N}$ exists, then

$$\boldsymbol{x_N} = \boldsymbol{W_N^{-1} X_N}$$

But this is an expression for IDFT. In fact, the IDFT can be expressed in matrix form as

$$\boldsymbol{x_N} = \frac{1}{N}\boldsymbol{W_N^{*} X_N}$$

Where $\boldsymbol{W_N^{*}}$ is the complex conjugate of the martix $\boldsymbol{W_N}$. Thus comparing we get

$$\boldsymbol{W_N} = \frac{1}{N}\boldsymbol{W_N^{*}}$$

Which, in turn, implies

$$\boldsymbol{W_N W_N^{*}} = N\boldsymbol{I_N}$$

Where $\boldsymbol{I_N}$ is an $N \times N$ identity martix.

Example: Compute the DFT of the four-point sequence, $x(n) = (0, 1, 2, 3)$

Solution: The first step is to determine the matrix $\boldsymbol{W_4}$.

It is easy to prove that the $\boldsymbol{W_N}$ has a periodic propery $W_N^{N+k} = W_N^k$

$$W_4^0 == e^{-j\frac{2\pi}{4}\times 0} = e^0 = 1$$

$$W_4^1 == e^{-j\frac{2\pi}{4}\times 1} = e^{-j\frac{\pi}{2}} = \cos\frac{\pi}{2} - j\sin\frac{\pi}{2} = -j$$

$$W_4^2 == e^{-j\frac{2\pi}{4}\times 2} = e^{-j\pi} = \cos\pi - j\sin\pi = -1$$

$$W_4^3 == e^{-j\frac{2\pi}{4}\times 3} = e^{-j\frac{3\pi}{2}} = \cos\frac{3\pi}{2} - j\sin\frac{3\pi}{2} = j$$

We also see that $W_N^{N/2+k} = -W_N^k$

$$[\boldsymbol{W}_4] = W_4^{kn} = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^2 \\ W_4^0 & W_4^3 & W_4^2 & W_4^1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

Now using, $\boldsymbol{X_N} = \boldsymbol{W_N x_N}$

$$\boldsymbol{X_N} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 6 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix}$$

Let see the inverse of $\boldsymbol{X_N}$ using $\boldsymbol{x_N} = \frac{1}{N}\boldsymbol{W_N^*}\boldsymbol{X_N}$, Thus

$$\boldsymbol{W_N^*} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}$$

$$\boldsymbol{x_N} = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 6 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 6-2+2j-2-2-2j \\ 6-2j+2j^2-2+2j+2j^2 \\ 6+2-2j-2+2+2j \\ 6+2j-2j^2+2-2j-2j^2 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 0 \\ 4 \\ 8 \\ 12 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix}$$

- The DFT and IDFT are computational tools that play a very important role in many digital signal processing applications, such as frequency analysis (spectrum analysis) of signal, power spectrum estimation, and linear filtering.
- The importance of the DFT and IDFT in such practical applications is due to a large extent on the existance of computationally efficient algorithms, known collectively as fast Fourier transform (FFT) algorithems, for computing the DFT and IDFT.
- The importance of DFT in engineering can never be understated. It is key in
  - Signal and image processing, specially in areas of filtering, and spectrum analysis.
  - Data compression techniques, such as those used in JPEG image compression.
  - Solving partial differential equations.
  - Digital signal processing, for instance in telecommunications and in biomedical engineering.
- Through its wide applications and significant impact, DFT continues to be a cornerstone in understanding and shaping the world of engineering.

Properties of the Discrete Fourier Transform

1. Linearity

If two finite-duration sequences $x_1(n)$ and $x_2(n)$ are linearly combined, i.e., if

$$x_3(n) = ax_1(n) + bx_2(n)$$

Then the DFT of $x_3(n)$ is

$$X_3(k) = aX_1(k) + bX_2(k)$$

Where the lengths of the sequences and their discrete Fourier transforms are all equal to the maximum of the lengths of $x_1(n)$ and $x_2(n)$. Of course, DTFs of greater length can be computed by augmenting both sequences with zero-valued samples.

2. Symmetry

For complex valued sequence $x(n)$ and its DFT $X(k)$, i.e.,

$$x(n) = x_R(n) + jx_I(n), \qquad for\ 0 \le n \le N-1$$

$$X(k) = X_R(k) + jX_I(k), \qquad for\ 0 \le k \le N-1$$

Now,

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

$$X_R(k) + jX_I(k) = \sum_{n=0}^{N-1} [x_R(n) + jx_I(n)] e^{-j\frac{2\pi}{N}kn}$$

$$= \sum_{n=0}^{N-1} [x_R(n) + jx_I(n)][\cos(\frac{2\pi}{N}kn) - j\sin(\frac{2\pi}{N}kn)]$$

$$= \sum_{n=0}^{N-1} [x_R(n)\cos(\frac{2\pi}{N}kn) + x_I(n)\sin(\frac{2\pi}{N}kn)]$$

$$-j\sum_{n=0}^{N-1} [x_R(n)\sin\left(\frac{2\pi}{N}kn\right) - x_I(n)\cos(\frac{2\pi}{N}kn)]$$

Similarly of IDFT,

$$x_R(n) + jx_I(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi}{N}kn}, \qquad for\ n = 0, 1, \dots, N-1$$

$$= \frac{1}{N}\sum_{k=0}^{N-1}[X_R(k)\cos(\frac{2\pi}{N}kn) + X_I(k)\sin(\frac{2\pi}{N}kn)]$$

$$-j\frac{1}{N}\sum_{k=0}^{N-1}[X_R(k)\sin\left(\frac{2\pi}{N}kn\right) - X_I(k)\cos(\frac{2\pi}{N}kn)]$$

a. Real-Valued sequences

If $x(n)$ is real-valued sequences:

$$X(N-k) = X^*(k) = X(-k)$$

b. Real and even sequences

If $x(n)$ is real and even sequences, that is,

$$x(n) = x(N-n), \qquad 0 \le n \le N-1$$

Then $X_I(k) = 0$. Hence the DFT reduces to

$$X(k) = \sum_{n=0}^{N-1} x(n)\cos\left(\frac{2\pi}{N}kn\right), \qquad 0 \le k \le N-1$$

Which is itself real-valued and even. Furthermore, since $X_I(k) = 0$, the IDFT reduces to

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\cos\left(\frac{2\pi}{N}kn\right), \qquad 0 \le n \le N-1$$

c. Real and Odd sequences

If $x(n)$ is real and odd, that is,

$$x(n) = -x(N-n), \qquad 0 \le n \le N-1$$

Then $X_R(k) = 0$, hence

$$X(k) = -j\sum_{n=0}^{N-1} x(n)\sin\left(\frac{2\pi}{N}kn\right), \qquad 0 \le k \le N-1$$

Which is purely imaginary and odd. Since $X_R(k) = 0$, then IDFT reduces to

$$x(n) = j\frac{1}{N}\sum_{k=0}^{N-1} X(k)\sin\left(\frac{2\pi}{N}kn\right), \qquad 0 \le n \le N-1$$

d. Purely Imaginary sequences

In this case, $x(n) = jx_I(n)$. Then

$$X_R(k) = \sum_{n=0}^{N-1} x_I(n) \sin(\frac{2\pi}{N} kn), \qquad 0 \le k \le N-1$$

$$X_I(k) = \sum_{n=0}^{N-1} x_I(n) \cos(\frac{2\pi}{N} kn), \qquad 0 \le k \le N-1$$

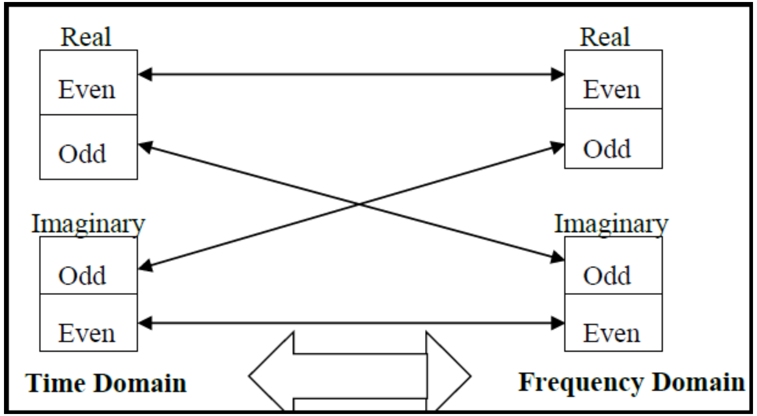
We observe that $X_R(k)$ is odd and $X_I(k)$ is even.

Case I: If $x_I(n)$ is odd, then $X_I(k) = 0$ and hence $X(k)$ is purely real.

Case II: If $x_I(n)$ is even, then $X_R(k) = 0$ and hence $X(k)$ is purely imaginary.

The symmetry properties is summarized in block diagram as

$x[n]$	$X[k]$	$\mathrm{Re}(X[k])$	$\mathrm{Im}(X[k])$
real	$X[-k] = X[k]^*$	even	odd
real & even	real & even	even	zero
real & odd	imaginary & odd	zero	odd
imaginary	$X[-k] = -X[k]^*$	odd	even
imaginary & even	imaginary & even	zero	even
imaginary & odd	imaginary & odd	odd	zero



Circular Symmetries of a Sequence

The N –point DFT of a finite duration sequence, $x(n)$ of length $L \leq N$ is equivalent ot the N –point DFT of a periodic sequence $\tilde{x}(n)$, of period N, which is obtained by periodically extending $x(n)$, that is,

$$\tilde{x}(n) = \sum_{r=-\infty}^{\infty} x[n - rN]$$

Now suppose that we shift the periodic sequene $\tilde{x}(n)$ by k units to the right. Thus we obtain another periodic sequence

$$\widetilde{x'}(n) = \tilde{x}(n-k) = \sum_{r=-\infty}^{\infty} x[n - k - rN]$$

The finite duration sequence

$$\tilde{x}(n) = \begin{cases} \widetilde{x'}(n), & 0 \leq n \leq N-1 \\ 0, & otherwise \end{cases}$$

Is related to the original sequence $x(n)$ by a circular shift. In general, the circular shift of the sequence can be represented as the index modulo N. Thus we can write

$$x'(n) = x(n-k, modulo\ N) = x((n-k))_N$$

Example : Let $x(n) = (\mathbf{1}, 2, 3, 4)$ as in figure (a). The periodic sequence $x_p(n)$ is shown if figure (b)

$$x_p(n) = (\ldots 1, 2, 3, 4, \mathbf{1}, 2, 3, 4, 1, 2, 3, 4, \ldots)$$

Now we delay the periodic sequence $x_p(n)$ by two samples as in figure (c) and this sequence is denoted by $x_p(n-2)$.

$$x_p(n-2) = (\ldots 1, 2, 3, 4, 1, 2, \mathbf{3}, 4, 1, 2, 3, 4, \ldots)$$

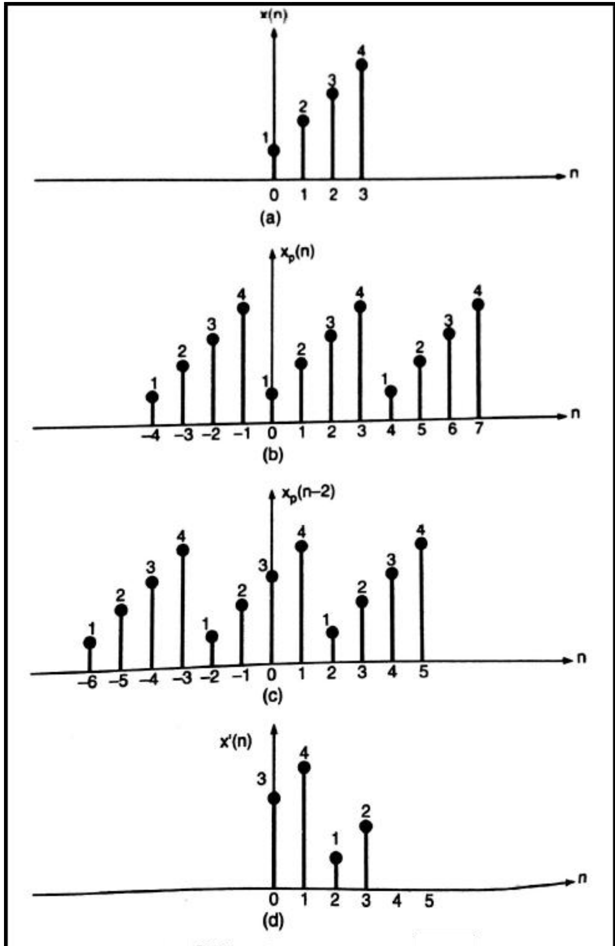
Since the original sequence is present in the range $n = 0$ *to* $n = 3$ so the shifted signal should be in the same range as in figure (d) and

$$x'(n) = (\mathbf{3}, 4, 1, 2)$$

Thus $x'(n)$ is obtained by circular shifting sequence $x(n)$, by two samples.

This means that $x'(n)$ is related to $x(n)$ by circular shift.

$$x'(n) = x(n - 2\ modulo\ 4) = x((n-2))_4$$



Shifting of sequence x(n)

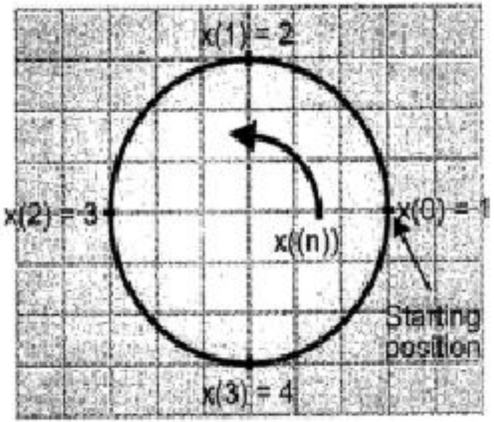
Graphical Representation

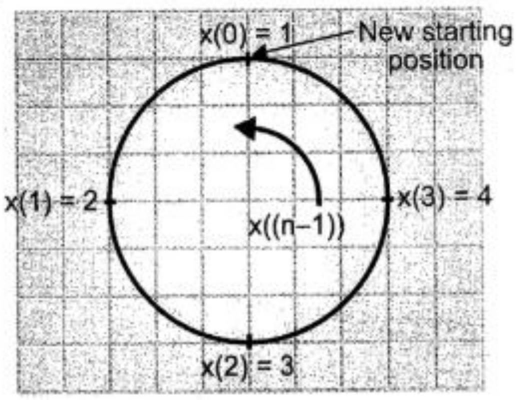
The circular shifting of the sequence can be plotted graphically as

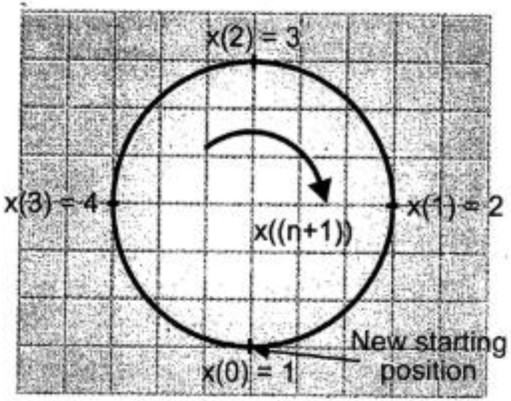
i. Circular Plot of Sequence $x(n) = (\mathbf{1}, 2, 3, 4)$ this plot is obtained by writing the samples of $x(n)$ circularly anti-clockwise as in first figure.

ii. Circular Delay: To delay sequence $x(n)$ circularly by one sample i.e., $x(n-1)$, shift every sample circularly in anti-clockwise direction by 1 as in second figure. Note: Delay by ‘k’ samples means shift the sequence circularly in anticlockwise dierection by k.

iii. Circular Advance : To advance sequence $x(n)$ circularly by one sample i.e., $x(n+1)$, shift every sample circularly in clockwise direction by 1 as in third figure. Note: Advance by k samples means the sequence circularly in clockwise direction by k.







- The inherent periodicity resulting from the arrangement of the N −point sequence on the circumference of a circle dictates a different defination of even and odd symmetry, and time reversal of a sequence.
- An N −point sequence is called circulary even if it is symmetric about the point zero on the cirlce. This implies that

$$x(N-n) = x(n), \qquad 1 \le n \le N-1$$

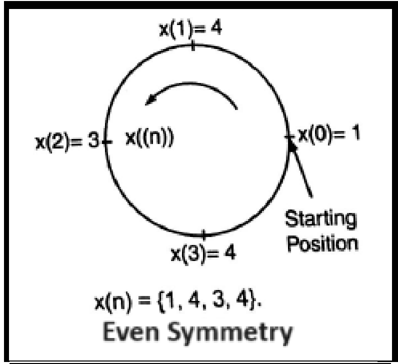
- An N −point sequence is called circulary odd if it is antisymmetric about the point zero on the cirlce. This implies that

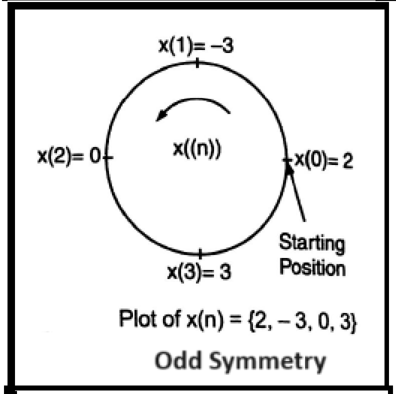
$$x(N-n) = -x(n), \qquad 1 \le n \le N-1$$

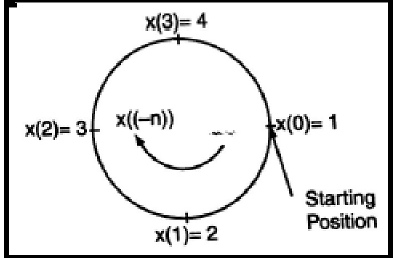
- The time reversal of an N −point sequence is attained by reversing its samples about the point zero on the circle. Thus the sequence $x((-n))_N$ is simply given as

$$x((-n))_N = \mathrm{x(N-n)}, \qquad 0 \le n \le N-1$$

- This time reversal is equivalent to plotting $x(n)$ in a clockwise direction on a circle. Example $x(n) = (1,2,3,4)$ & $x(-n) = (1,4,3,2)$



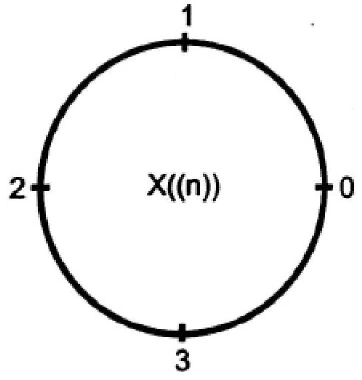


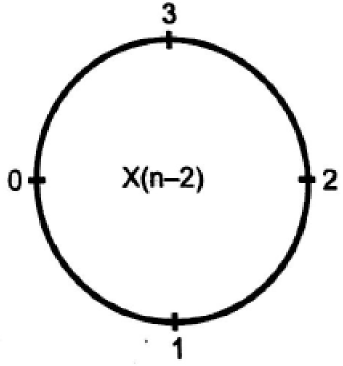


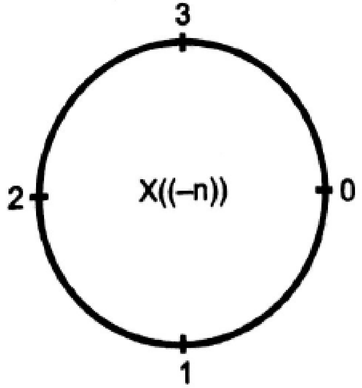
Questions: For the Sequence $x(n) = (0, 1, 2, 3)$, find

1. $x((n-2))_4$
2. $x((-n))_4$

$x(n) = (0, 1, 2, 3)$ $\quad$ $x((n-2))_4 = (2, 3, 0, 1)$ $\quad$ $x((-n))_4 = (0, 3, 2, 1)$







3. Multiplication of Two DFTs and Circular Convolution

Suppose two finite duration sequences of length N, $x_1(n)$ and $x_2(n)$ and their respective N-point DFTs as

$$X_1(k) = \sum_{n=0}^{N-1} x_1(n) e^{-j\frac{2\pi}{N}kn}, \qquad k = 0, 1, \dots, N-1$$

$$X_2(k) = \sum_{n=0}^{N-1} x_2(n) e^{-j\frac{2\pi}{N}kn}, \qquad k = 0, 1, \dots, N-1$$

If we multiply the two DFTs together, the result is a DFT, say $X_3(k)$, of a sequence $x_3(n)$ of length N.

Let us determine the relationship between $x_3(n)$ and the sequences $x_1(n)$ and $x_2(n)$.

We have

$$X_3(k) = X_1(k) \times X_2(k), \qquad k = 0, 1, \dots, N-1$$

The IDFT of $X_3(k)$ is

$$x_3(m) = \frac{1}{N}\sum_{k=0}^{N-1} X_3(k)e^{j\frac{2\pi}{N}km}, \qquad 0 \le m \le N-1$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} X_1(k)X_2(k)e^{j\frac{2\pi}{N}km}, \qquad 0 \le m \le N-1$$

$$= \frac{1}{N}\sum_{k=0}^{N-1}[\sum_{n=0}^{N-1} x_1(n)e^{-j\frac{2\pi}{N}kn}\sum_{l=0}^{N-1} x_1(l)e^{-j\frac{2\pi}{N}kl}]e^{j\frac{2\pi}{N}km}$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} x_1(n)\sum_{l=0}^{N-1} x_2(l)\,[\sum_{k=0}^{N-1} e^{j\frac{2\pi}{N}k(m-n-l)}]$$

Let define $a = e^{j\frac{2\pi}{N}(m-n-l)}$, then

$$\sum_{k=0}^{N-1} e^{j\frac{2\pi}{N}k(m-n-l)} = \sum_{k=0}^{N-1} a^k = \begin{cases} N, & a = 1 \\ \dfrac{1-a^N}{1-a}, & a \neq 1 \end{cases}$$

$a = 1$ when $m-n-l$ is a multiple of N i.e., $m-n-l = \pm pN$.

On the other hand, $a^N = e^{j2\pi(m-n-l)} = 1$ for any value of $a \neq 0$.

Thus,

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & l = m-n+pN = \left((m-n)\right)_N, \quad p \; an \; integer \\ 0, & otherwise \end{cases}$$

Therefore,

$$x_3(m) = \sum_{n=0}^{N-1} x_1(n)\, x_2\left((m-n)\right)_N, \qquad \mathrm{m} = 0,1,\ldots,\mathrm{N}-1$$

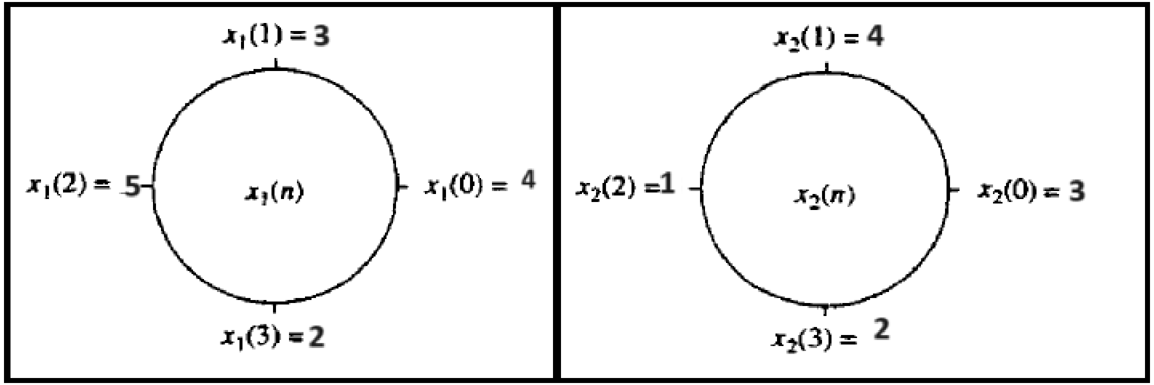
$$x_3(m) = \sum_{n=0}^{N-1} x_1(n)\, x_2\big((m-n)\big)_N, \qquad \mathrm{m} = 0,1,\ldots,\mathrm{N}-1$$

This expression has the form of a convolution sum. However, it is not the ordinary linear convolution which relates the output sequence $y(n)$ of a linear system to the input sequence $x(n)$ and the impulse response $h(n)$. Instead, the convolution sum in the above equation involves the index $\big((m-n)\big)_N$ and is called circular convolution.

Thus we conclude that multiplication of the DFTs of two sequences is equivalent to the circular convolution of the two sequences in the time domain.

$$x_3(n) = x_1(n) \textcircled{N} x_2(n)$$

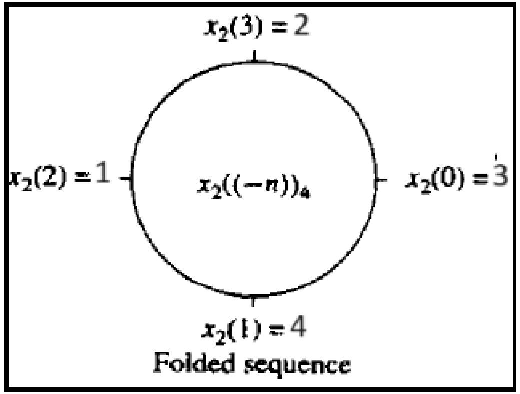
Example 1 : Perform the circular convolution of the following two sequences:
$x_1(n) = (\mathbf{4}, 3, 5, 2)$ and $x_2(n) = (\mathbf{3}, 4, 1, 2)$
Solution: The sequence $x_1(n)$ and $x_2(n)$ are graphed in a counter clockwise direction on a circle as shown in figure below.



Now $x_3(n)$ is obtained by circularly convolving $x_1(n)$ and $x_2(n)$ as

$$x_3(m) = \sum_{n=0}^{3} x_1(n)\, x_2\big((m-n)\big)_4, \qquad \mathrm{m} = 0, 1, 2, 3$$

Beginning with $m = 0$, we have



$$x_3(0) = \sum_{n=0}^{3} x_1(n)\, x_2\big((-n)\big)_4$$

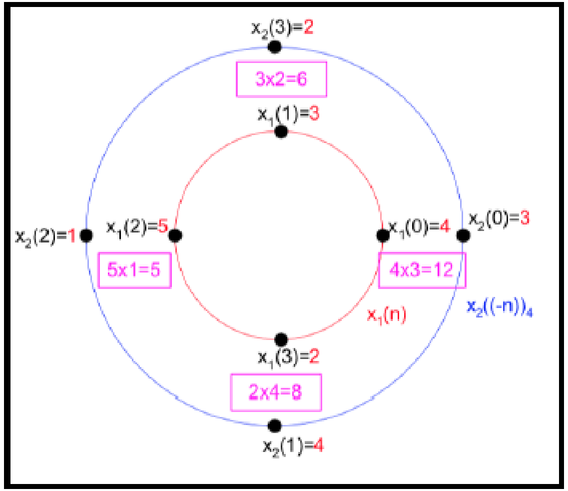
$x_2\big((-n)\big)_4$ is simply the sequence $x_2(n)$ folded and graphed on a circle as in figure.

The product sequence is obtained by multiplying $x_1(n)$ with $x_2\big((-n)\big)_4$, point by point.

This sequence is illustrated as in figure below.

Finally the sum the value in the product sequence to obtain

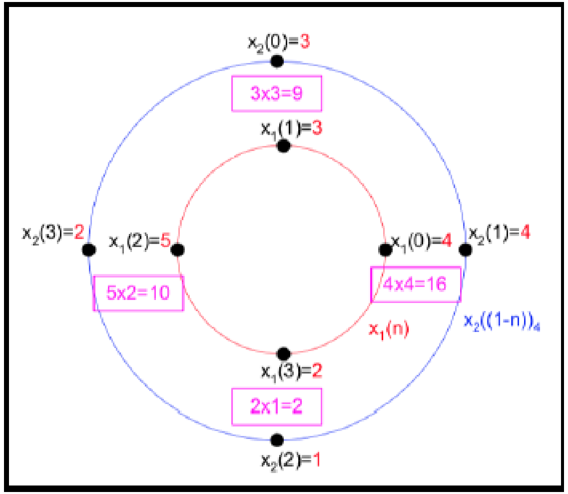
$$x_3(0) = (12 + 6 + 5 + 8) = (31)$$



For $m = 1$, we have

$$x_3(1) = \sum_{n=0}^{3} x_1(n)\, x_2\big((1-n)\big)_4$$

$x_2\big((1-n)\big)_4$ is same as $x_2\big((-(n-1))\big)_4$ is simply the sequence $x_2\big((-n)\big)_4$ rotated counterclockwise by one unit in time. This rotated sequence multiples $x_1(n)$ to yield the product sequence as illustrated in figure.



Finally, we sum the values in the product sequence to obtain $x_3(1) = (16 + 9 + 10 + 2) = (37)$

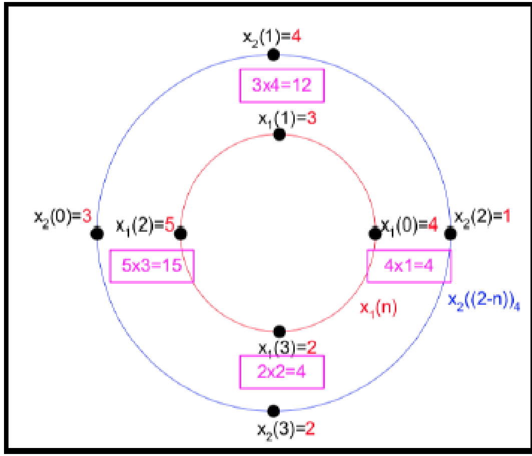
For $m = 2$, we have

$$x_3(2) = \sum_{n=0}^{3} x_1(n)\, x_2\big((2-n)\big)_4$$

$x_2\big((2-n)\big)_4$ is simply the sequence $x_2\big((-n)\big)_4$ rotated counterclockwise by two unit in time.

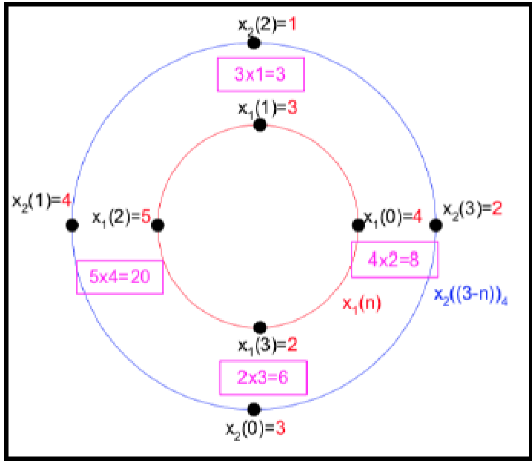
This rotated sequence multiples $x_1(n)$ to yield the product sequence as illustrated in figure. Thus

$$x_3(2) = (4 + 12 + 15 + 4) = (35)$$



For $m = 3$, we have

$$x_3(3) = \sum_{n=0}^{3} x_1(n)\, x_2\big((3-n)\big)_4$$



$x_2\big((3-n)\big)_4$ is simply the sequence $x_2\big((-n)\big)_4$ rotated counterclockwise by three unit in time and this rotated sequence multiples $x_1(n)$ to yield the product sequence as illustrated in figure. Thus

$$x_3(2) = (3 + 20 + 6 + 8) = (37)$$

Therefore, the circular convolution of the sequences $x_1(n)$ and $x_2(n)$ yields the sequence $x_3(n) = (\mathbf{31}, 37, 35, 37)$.

Proof using multiplication: $x_1(n) = (\mathbf{4}, 3, 5, 2)$ and $x_2(n) = (\mathbf{3}, 4, 1, 2)$

$$\boldsymbol{X_1(k)} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 4 \\ 3 \\ 5 \\ 2 \end{bmatrix} = \begin{bmatrix} 14 \\ -1-j \\ 4 \\ -1+j \end{bmatrix}$$

$$\boldsymbol{X_2(k)} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 3 \\ 4 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 10 \\ 2-2j \\ -2 \\ 2+2j \end{bmatrix}$$

And $\boldsymbol{X_3(k)} = \boldsymbol{X_1(k)} \times \boldsymbol{X_2(k)} = (\mathbf{140}, -4, -8, -4)$

$$\boldsymbol{x_3}(n) = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 140 \\ -4 \\ -8 \\ -4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 140-4-8-4 \\ 140-4j+8+4j \\ 140+4-8+4 \\ 140+4j+8-4j \end{bmatrix} = \begin{bmatrix} 31 \\ 37 \\ 35 \\ 37 \end{bmatrix}$$

Thus $x_3(n) = (\mathbf{31}, 37, 35, 37)$

Circular Convolution Using Martix Method

The graphical method is quite tedious, especially when many samples are present. While the martix method is more convenient. The circular convolution is given as:

$$x_3(n) = x_1(n) \circledN x_2(n)$$

$$x_3(m) = \sum_{n=0}^{N-1} x_1(n)\, x_2\big((m-n)\big)_N, \qquad \mathrm{m} = 0,1,\dots,\mathrm{N}-1$$

$$x_3(0) = x_1(0)x_2(0) + x_1(1)x_2(-1) + x_1(2)x_2(-2) + \cdots$$
$$+x_1(N-2)x_2\big(-(N-2)\big) + x_1(N-1)x_2\big(-(N-1)\big)$$
$$= x_1(0)x_2(0) + x_1(1)x_2(N-1) + x_1(2)x_2(N-2) + \cdots + x_1(N-2)x_2(2) + x_1(N-1)x_2(1)$$

$$x_3(1) = x_1(0)x_2(1) + x_1(1)x_2(0) + x_1(2)x_2(N-1) + \cdots + x_1(N-2)x_2(3) + x_1(N-1)x_2(2)$$

.

.

$$x_3(N-1)$$
$$= x_1(0)x_2(N-1) + x_1(1)x_2(N-2) + x_1(2)x_2(N-3) + \cdots + x_1(N-2)x_2(1) + x_1(N-1)x_2(0)$$

In matrix

$$\begin{bmatrix} x_3(0) \\ x_3(1) \\ x_3(2) \\ . \\ . \\ . \\ x_3(N-2) \\ x_3(N-1) \end{bmatrix} = \begin{bmatrix} x_2(0) & x_2(N-1) & x_2(N-2) & . & . & . & x_2(2) & x_2(1) \\ x_2(1) & x_2(0) & x_2(N-1) & . & . & . & x_2(3) & x_2(2) \\ x_2(2) & x_2(1) & x_2(0) & . & . & . & x_2(4) & x_2(3) \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ x_2(N-2) & x_2(N-3) & x_2(N-4) & . & . & . & x_2(0) & x_2(N-1) \\ x_2(N-1) & x_2(N-2) & x_2(N-3) & . & . & . & x_2(1) & x_2(0) \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_1(1) \\ x_1(2) \\ . \\ . \\ . \\ x_1(N-2) \\ x_1(N-1) \end{bmatrix}$$

$x_1(n) = (\mathbf{4}, 3, 5, 2)$ and $x_2(n) = (\mathbf{3}, 4, 1, 2)$

$$\begin{bmatrix} x_3(0) \\ x_3(1) \\ x_3(2) \\ x_3(3) \end{bmatrix} = \begin{bmatrix} 3 & 2 & 1 & 4 \\ 4 & 3 & 2 & 1 \\ 1 & 4 & 3 & 2 \\ 2 & 1 & 4 & 3 \end{bmatrix} \begin{bmatrix} 4 \\ 3 \\ 5 \\ 2 \end{bmatrix} = \begin{bmatrix} 12+6+5+8 \\ 16+9+10+2 \\ 4+12+15+4 \\ 8+3+20+6 \end{bmatrix} = \begin{bmatrix} 31 \\ 37 \\ 35 \\ 37 \end{bmatrix}$$

Question : Compute Circular convolution of the two sequences using DFT

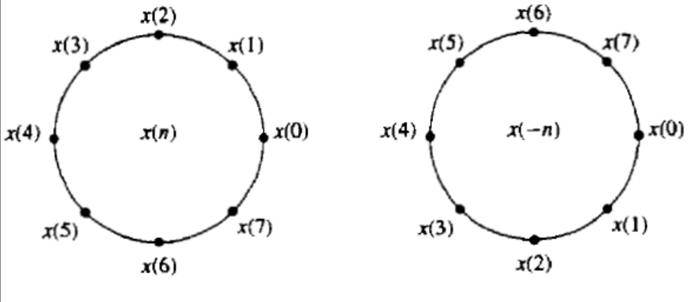
$x_1(n) = (\mathbf{2}, 1, 2, 1)$ and $x_2(n) = (\mathbf{1}, 2, 3, 4)$

Answer:

$$X_1(k) = (0, 1, 2, 3)$$
$$X_2(k) = (10, -2 + j2, -2, -2 - j2)$$
$$X_3(k) = (60, 0, -4, 0)$$
$$x_3(n) = (14, 16, 14, 16)$$

4. Time Reversal of a sequence

If



Time reversal of a sequence.

$$x(n) \xleftrightarrow[N]{DFT} X(k)$$

Then

$$x((-n))_N = x(N-n) \xleftrightarrow[N]{DFT} X\big((-k)\big)_N = X(N-k)$$

Hence reversing the N-point sequence in time is equivalent to reversing the DFT values. Time reversal of a sequence $x(n)$ is shown in figure. The time reversal is equivalent to plotting $x(n)$ in a clockwise direction on a circle.

Proof:

$$DFT\{x(N-n)\} = \sum_{n=0}^{N-1} x(N-n)e^{-j\frac{2\pi}{N}kn}$$

Let take $m = N - n$ thus $n = N - m$

$$\sum_{n=0}^{N-1} x(N-n)e^{-j\frac{2\pi}{N}kn} = \sum_{m=0}^{N-1} x(m)e^{-j\frac{2\pi}{N}k(N-m)} = \sum_{m=0}^{N-1} x(m)e^{j\frac{2\pi}{N}km}\, e^{-j\frac{2\pi}{N}kN}$$

$$= \sum_{m=0}^{N-1} x(m)e^{j\frac{2\pi}{N}km}\, e^{-j2\pi k} = \sum_{m=0}^{N-1} x(m)e^{j\frac{2\pi}{N}km}$$

$$= \sum_{m=0}^{N-1} x(m)e^{j\frac{2\pi}{N}km}\, e^{-j2\pi m}$$

$$= \sum_{m=0}^{N-1} x(m)e^{j\frac{2\pi}{N}km}\, e^{-j\frac{2\pi}{N}mN}$$

$$= \sum_{m=0}^{N-1} x(m)e^{-j\frac{2\pi}{N}m(N-k)}$$

$$= X(N-k)$$

We note that $X(N-k) = X\big((-k)\big)_N, 0 \le k \le N-1$

5. Time shift (circular time shift of a sequence)

The circular time shift property of DFT says that if a discrete time signal is circularly shifted time by m units then its DFT is multiplied by $e^{-j\frac{2\pi}{N}km}$.

i.e., if $x(n)\underset{N}{\overset{DFT}{\longleftrightarrow}}X(k)$, then $x((n-m))_N \underset{N}{\overset{DFT}{\longleftrightarrow}}X(k)e^{-j\frac{2\pi}{N}km}$

Proof:

$$DFT\{x(n-m)_N\} = \sum_{n=0}^{N-1} x(n-m)_N e^{-j\frac{2\pi}{N}kn}$$

Let $p = n - m$ then $n = p + m$

$$\sum_{n=0}^{N-1} x(n-m)_N e^{-j\frac{2\pi}{N}kn} = \sum_{p=0}^{N-1} x(p)e^{-j\frac{2\pi}{N}k(p+m)} = \sum_{n=0}^{N-1} x(p)e^{-j\frac{2\pi}{N}kp}\, e^{-j\frac{2\pi}{N}km}$$

$$= \sum_{n=0}^{N-1} [x(p)e^{-j\frac{2\pi}{N}kp}]\, e^{-j\frac{2\pi}{N}km} = X(k)e^{-j\frac{2\pi}{N}km}$$

6. Circular Frequence shift

The circular frequency shift property of DFT says that if a discrete time signal is multiplied by $e^{j\frac{2\pi}{N}mn}$ its DFT is circularly shifted by m units.

i.e., if $x(n)\overset{DFT}{\underset{N}{\longleftrightarrow}}X(k)$, then $x(n)e^{j\frac{2\pi}{N}mn}\overset{DFT}{\underset{N}{\longleftrightarrow}}X((k-m))_N$

Proof:

$$DFT\left\{x(n)e^{j\frac{2\pi}{N}mn}\right\} = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}\, e^{j\frac{2\pi}{N}mn}$$

$$= \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}n(k-m)}$$

$$= X((k-m))_N$$

7. Duality

If $x(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} X(k)$, then $\mathrm{X}(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} Nx((-k))_N$

Proof:

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

If we take DFT of the sequence $X(n)$, then

$$Y(k) = \sum_{n=0}^{N-1} X(n) e^{-j\frac{2\pi}{N}kn}, \qquad for\ k = 0,1,2,\ldots,N-1$$

$$= \mathrm{N}(\frac{1}{N} \sum_{n=0}^{N-1} X(n) e^{j\frac{2\pi}{N}(-k)n}), \qquad for\ k = 0,1,2,\ldots,N-1$$

$$= Nx((-k))_N$$

8. Multiplication

The multiplication property of DFT says ,the DFT of product of two discrete time sequences is equivalent to circular convolution of the DFTs of the individual sequences scaled by a factor $\frac{1}{N}$ i.e,

If $x_1(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} X_1(k)$ and $x_2(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} X_2(k)$,

then $x_1(n) \times x_2(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} \frac{1}{N}[X_1(k) Ⓝ X_2(k)]$

Proof:

$$x_1(n) = \frac{1}{N}\sum_{m=0}^{N-1} X_1(m) e^{j\frac{2\pi}{N}mn}, \qquad for\ n = 0,1,2,\ldots,N-1$$

Now

$$DFT\ [x_1(n) \times x_2(n)] = \sum_{n=0}^{N-1} x_1(n) x_2(n) e^{-j\frac{2\pi}{N}kn}$$

$$= \sum_{n=0}^{N-1} [\frac{1}{N} \sum_{m=0}^{N-1} X_1(m) e^{j\frac{2\pi}{N}mn}] x_2(n) e^{-j\frac{2\pi}{N}kn}$$

Interchanging the order of summation

$$= \frac{1}{N} \sum_{m=0}^{N-1} X_1(m) \sum_{n=0}^{N-1} x_2(n)\, e^{j\frac{2\pi}{N}mn} e^{-j\frac{2\pi}{N}kn}$$

$$= \frac{1}{N} \sum_{m=0}^{N-1} X_1(m) \sum_{n=0}^{N-1} x_2(n)\, e^{-j\frac{2\pi}{N}(m-k)n}$$

$$= \frac{1}{N} \sum_{m=0}^{N-1} X_1(m)\, X_2((m-k))_N$$

$$= \frac{1}{N} [X_1(k) Ⓝ X_2(k)]$$

9. Conjugation & conjugate symmetry

Let $x(n)$ be a complex N –point discrete sequence and $x^*(n)$ be its conjugate sequence.

Now if $x(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} X(k)$, then $x^*(n) \underset{N}{\overset{DFT}{\longleftrightarrow}} X^*(N-k)$

Proof:

$$DFT\{x^*(n)\} = X(k) = \sum_{n=0}^{N-1} x^*(n) e^{-j\frac{2\pi}{N}kn}, \qquad for\ k = 0,1,2,\ldots,N-1$$

$$= [\sum_{n=0}^{N-1} x(n) e^{j\frac{2\pi}{N}kn}]^* = [\sum_{n=0}^{N-1} x(n) e^{j\frac{2\pi}{N}kn} e^{-j\frac{2\pi}{N}kN}]^*$$

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

Example: DFT of a sequence $x(n)$ is given by

$$X(k) = (4, 1+2j, j, 1-3j)$$

Using the DFT property, determine DFT of $x^*(n)$

Solution: Using if $x(n) \xleftrightarrow[N]{DFT} X(k)$, then $x^*(n) \xleftrightarrow[N]{DFT} X^*(N-k)$

$$X(k) = (4, 1+2j, j, 1-3j)$$

$$X^*(\mathrm{k}) = (4, 1-2j, -j, 1+3j)$$

$X^*(N-k)$ is the plotting the samples of $X^*(k)$ in clockwise direction

$$X^*(N-k) = (4, 1+3j, -j, 1-2j)$$

Note : $X(k) = \mathcal{DFT}\{x(n)\}$; $X_1(k) = \mathcal{DFT}\{x_1(n)\}$; $X_2(k) = \mathcal{DFT}\{x_2(n)\}$; $Y(k) = \mathcal{DFT}\{y(n)\}$

Property	Discrete time signal	Discrete Fourier Transform
Linearity	$a_1x_1(n) + a_2x_2(n)$	$a_1X_1(k) + a_2\ X_2(k)$
Periodicity	$x(n + N) = x(n)$	$X(k + N) = X(k)$
Circular time shift	$x((n - m))_N$	$X(k)\, e^{\frac{-j2\pi k m}{N}}$
Time reversal	$x(N - n)$	$X(N - k)$
Conjugation	$x^*(n)$	$X^*(N - k)$
Circular frequency shift	$x(n)\, e^{\frac{j2\pi m n}{N}}$	$X((k - m))_N$
Multiplication	$x_1(n)\, x_2(n)$	$\frac{1}{N}[X_1(k) \circledast X_2(k)]$
Circular convolution	$x_1(n) \circledast x_2(n) = \sum_{m=0}^{N-1} x_1(m)\, x_2((n-m))_N$	$X_1(k)\, X_2(k)$

Note : $X(k) = \mathcal{DFT}\{x(n)\}$; $X_1(k) = \mathcal{DFT}\{x_1(n)\}$; $X_2(k) = \mathcal{DFT}\{x_2(n)\}$; $Y(k) = \mathcal{DFT}\{y(n)\}$

Property	Discrete time signal	Discrete Fourier Transform
Symmetry of real signals	$x(n)$ is real	$X(k) = X^*(N-k)$ $X_r(k) = X_r(N-k)$ $X_i(k) = -X_i(N-k)$ $\lvert X(k)\rvert = \lvert X(N-k)\rvert$ $\angle X(k) = -\angle X(N-k)$
Symmetry of real and even signal	$x(n)$ is real and even $x(n) = x(N-n)$	$X(k) = X_r(k)$ and $X_i(k) = 0$
Symmetry of real and odd signal	$x(n)$ is real and odd $x(n) = -x(N-n)$	$X(k) = jX_i(k)$ and $X_r(k) = 0$

Frequency response of LTI system

- Fourier transform is an useful tool for the analysis of discrete time systems in frequency domain.
- But the main drawback in Fourier transform is that it is a continuous function of ω and so it will not be useful for digital processing of signals and systems.
- Hence DFT is propesd, therefore the analysis of discrete time systems in frequency domain can be conveniently performed using DFT for digital processing of signals and systems.
- Efficient algorithms like FFT are available for computing the DFT of a finite duration sequence.
- Because of these algorithms are available, it is computationally efficient to implement a convolution of two sequences by following procedure.

i. Compute the $N-$point DFTs $X_1(k)$ and $X_2(k)$ of the two sequences $x_1(n)$ and $x_2(n)$ respectively.

ii. Compute the product $X_3(k) = X_1(k)X_2(k)$ for $0 \leq k \leq N-1$.

iii. Compute the sequence $x_3(n) = x_1(n) \textcircled{N} x_2(n)$ as the inverse DFT of $X_3(k)$.

- In most applications, we are interested in implementing a linear convolution of two sequence; i.e., we wish toe implement a linear time-invariant system.
- This is certainly true, for example, in filtering a sequence such as a speech waveform or a radar signals.
- The multiplication of discrete Fourier transforms corresponds to a circular convolution of the sequences.
- To obtain a linear convolution, we must ensure that circular convolution has the effect of linear convolution.

Response of LTI Discrete time system using DFT

The response of an LTI discrete time system is given by linear convolution of input and impulse respose of the system.

Let $x(n)$ be the input to the LTI system and $\mathrm{h}(n)$ be the impulse response of the LTI system. Now, the response or output of the system $y(n)$ is given by,

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

The DFT supports only circular convolution and so, the linear convolution of above equation has to be computed via circular convolution.

If $x(n)$ is N_1 −point sequence and $\mathrm{h}(n)$ is N_2 −point sequence then linear convolution $x(n)$ and $\mathrm{h}(n)$ will generate $y(n)$ of size $N_1 + N_2 - 1$.

Therefore in order to perform linear convolution via circular convolution that $x(n)$ and $\mathrm{h}(n)$ should be converted to $N_1 + N_2 - 1$ point sequence by appending zeros.

Now the circular convolution of $N_1 + N_2 - 1$ point sequences $x(n)$ and $\mathrm{h}(n)$ will give same result as that of linear convolution. This is known as zero-padding.

Example: Determine the output response $y(n)$ if $h(n) = (\mathbf{1}, 1, 1)$, $x(n) = (\mathbf{1}, 2, 3,1)$ by using

i. Linear Convolution
ii. Circular Convolution
iii. Circular Convolution with zero padding

h(n)

x(n)	1	1	1
1	1	1	1
2	2	2	2
3	3	3	3
1	1	1	1

Solution:

i. We know for linear convolution

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

Using the matrix method

$$y(n) = (1, 3, 6, 6, 4, 1)$$

The number of samples in linear convolution is $N_1 + N_2 - 1 = 4 + 3 - 1 = 6$

ii. For circular convolution

$$h(n) = (\mathbf{1}, 1, 1, 0),$$

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

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 1+0+3+1 \\ 1+2+0+1 \\ 1+2+3+0 \\ 0+2+3+1 \end{bmatrix} = \begin{bmatrix} 5 \\ 4 \\ 6 \\ 6 \end{bmatrix}$$

$$y(n) = (5, 4, 6, 6)$$

The number of samples in linear convolution is $\max(N_1,\ N_2) = \max(3, 4) = 4$

iii. Circular with Zero Padding

To get the result of linear convolution with circular convolution we have to add appropriate number of zeros ($N_1 - 1$ and $N_2 - 1$) to both sequences, so that the length of both sequences becomes $N_1 + N_2 - 1$. Now

$$h(n) = (\mathbf{1}, 1, 1, 0,0, 0),$$
$$x(n) = (\mathbf{1}, 2, 3,1,0, 0)$$

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1+0+0+0+0+0 \\ 1+2+0+0+0+0 \\ 1+2+3+0+0+0 \\ 0+2+3+1+0+0 \\ 0+0+3+1+0+0 \\ 0+0+0+1+0+0 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 6 \\ 6 \\ 4 \\ 1 \end{bmatrix}$$

$$y(n) = (1, 3, 6, 6, 4, 1)$$

7.2 Fast Fourier Transform (FFT) algorithm
(decimation in time algorithm, decimation in frequency algorithm)

- DTF plays an important role in many applications of digital signal processing including linear filtering, correlation analysis, and spectrum analysis.
- A major reason for its importance is the existence of efficient algorithms for computing the DTF.
- The formulas for the N-point DFT and IDFT can be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad for\ k = 0,1,2,\dots,N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \qquad for\ n = 0,1,2,\dots,N-1$$

where

$$W_N = e^{-j\frac{2\pi}{N}}$$

- Since the DFT and IDFT involve basically the same type of computations, so the computational algorithms for the DFT applies as well to the efficient computation of the IDFT.

- In general, the data sequence $x(n)$ is assumed to be complex valued.

$$X_R(k) + jX_I(k) = \sum_{n=0}^{N-1}[x_R(n)\cos(\frac{2\pi}{N}kn) + x_I(n)\sin(\frac{2\pi}{N}kn)]$$
$$-j\sum_{n=0}^{N-1}[x_R(n)\sin\left(\frac{2\pi}{N}kn\right) - x_I(n)\cos(\frac{2\pi}{N}kn)]$$

- For each value of k, direct computation of $\mathrm{X}(k)$ involves
  - N complex multiplications ($4N$ real multiplications) and
  - $N - 1$ complex additions ($4N - 2$ real additions).
- Consequently, to compute all N values of the DFT requires
  - N^2 complex multiplications ($4N^2$ real multiplications) and
  - $N(N - 1)$ complex additions ($N(4N - 1)$ real additions).
- Besides multiplication and additions, for computer to process, also requires provisions for storing and accessing the N complex input sequeuces values $x(n)$ and vlaues of W_N^{kn}.

- Direct computation of the DFT is basically inefficient also primarily because it does not exploit the symmetry and periodicity properties of the phase factor W_N
- In particular, these two properties are:

  Symmetry property: $W_N^{k+\frac{N}{2}} = -W_N^k$

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

- For large value of N to compute the DFT by direct method for amount of computation and computation time becomes vary large.
- In 1965 Cooley and Tukey publish an algorithm for the computation of DFT that is highly efficient computational algorithm known as Fast Fourier Transform (FFT).
- In computing the DFT, dramatic efficiency results from decomposing the computation into successively smaller DFT computations.
- Algorithms in which the decomposition is based on decomposing the sequence $x(n)$ into successively smaller subsequences are called decimation-in-time (DITFFT) algorithms.
- Algorithms in which the decomposition is based on decomposing the sequence $X(k)$ into successively smaller subsequences are called decimation-in-frequency (DIFFFT) algorithms.

Decimation-in-Time FFT algorithms

- Algorithms in which the decomposition is based on decomposing the sequence $x(n)$ into successively smaller subsequences are called decimation-in-time (DITFFT) algorithms.
- The principle of the decimation-in-time algorithm consider that N is an integer power of 2, i.e., $N = 2^v$.
- Since N is an even integer, so we compute $\mathrm{X}(k)$ by separating $x(n)$ into two $^N/_2$ –point sequences consisting of the even-numbered points in $x(n)$ and the odd-numbered points in $x(n)$. With $X(k)$ given by

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}\,, \qquad for\ k = 0,1,2,\ldots,N-1$$

And separating $x(n)$ into its even and odd numbered points, we obtain

$$X(k) = \sum_{even} x(n) W_N^{kn} + \sum_{odd} x(n) W_N^{kn}$$

With substitution of variables $n = 2r$ for n even and $n = 2r + 1$ for n odd,

$$X(k) = \sum_{r=0}^{N/_2-1} x(2r)W_N^{2rk} + \sum_{r=0}^{N/_2-1} x(2r+1)W_N^{(2r+1)k}$$

$$= \sum_{r=0}^{N/_2-1} x(2r)(W_N^2)^{rk} + W_N^k \sum_{r=0}^{N/_2-1} x(2r+1)(W_N^2)^{rk}$$

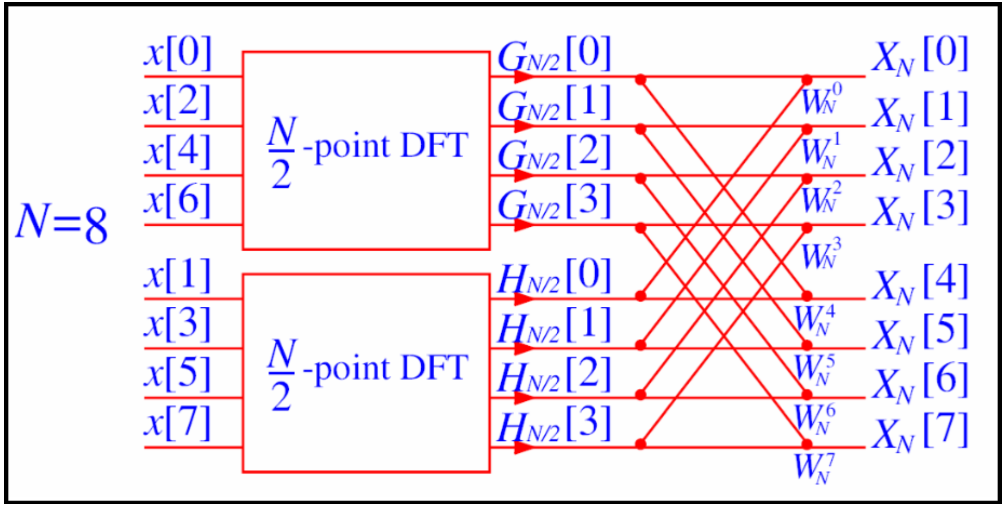
But $W_N^2 = e^{-2j(2\pi/_N)} = e^{-j\frac{2\pi}{N/_2}} = W_{N/_2}$.

Thus,

$$X(k) = \sum_{r=0}^{N/_2-1} x(2r)W_{N/_2}^{rk} + W_N^k \sum_{r=0}^{N/_2-1} x(2r+1)W_{N/_2}^{rk}$$

$$= G_{N/_2}(k) + W_N^k H_{N/_2}(k), \qquad k = 0,1,\dots,N-1$$

- Each of the sums is recognized as an ${}^{N}/_{2}$ $-$point DFT, the first sum $G_{N/_2}$ being the ${}^{N}/_{2}$ $-$point DFT of the even-numbered points of the original sequence and the second $H_{N/_2}$ being the ${}^{N}/_{2}$ $-$point DFT of the odd-numbered points of the original sequence.
- Figure depicts this computation for $N = 8$, where $G_{N/_2}(k)$ represent 4-point even-numbered points and $H_{N/_2}(k)$ represent 4-point odd-numbered points.
- The computation for all values of k requires at most $N + 2({}^{N}/_{2})^2$ complex multiplication and complex addition compared to N^2 complex multiplication and addition for direct computation.



- Now, if ${}^{N}/_{2}$ is even, as it is when N is equal to a power of 2, then we can consider computing each of the ${}^{N}/_{2}$ −point DFTs by breaking each of the sums into two ${}^{N}/_{4}$ −point DFTs, which then be combined to yield the ${}^{N}/_{2}$ −point DFTs.
- Thus $G(k)$ would be represented as

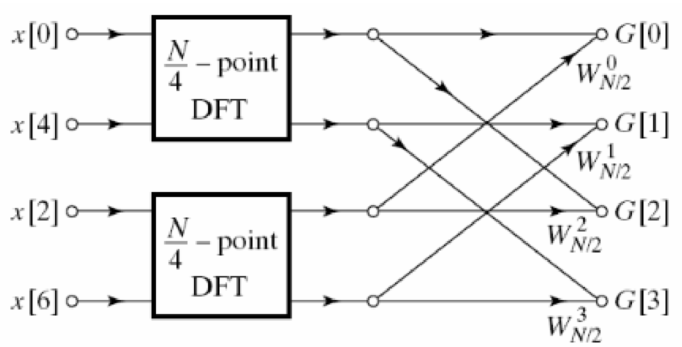
$$G(k) = \sum_{l=0}^{{}^{N}/_{4}-1} g(2l)W_{{}^{N}/_{2}}^{2lk} + W_{{}^{N}/_{2}}^{k} \sum_{l=0}^{{}^{N}/_{4}-1} g(2l+1)W_{{}^{N}/_{2}}^{2lk}$$

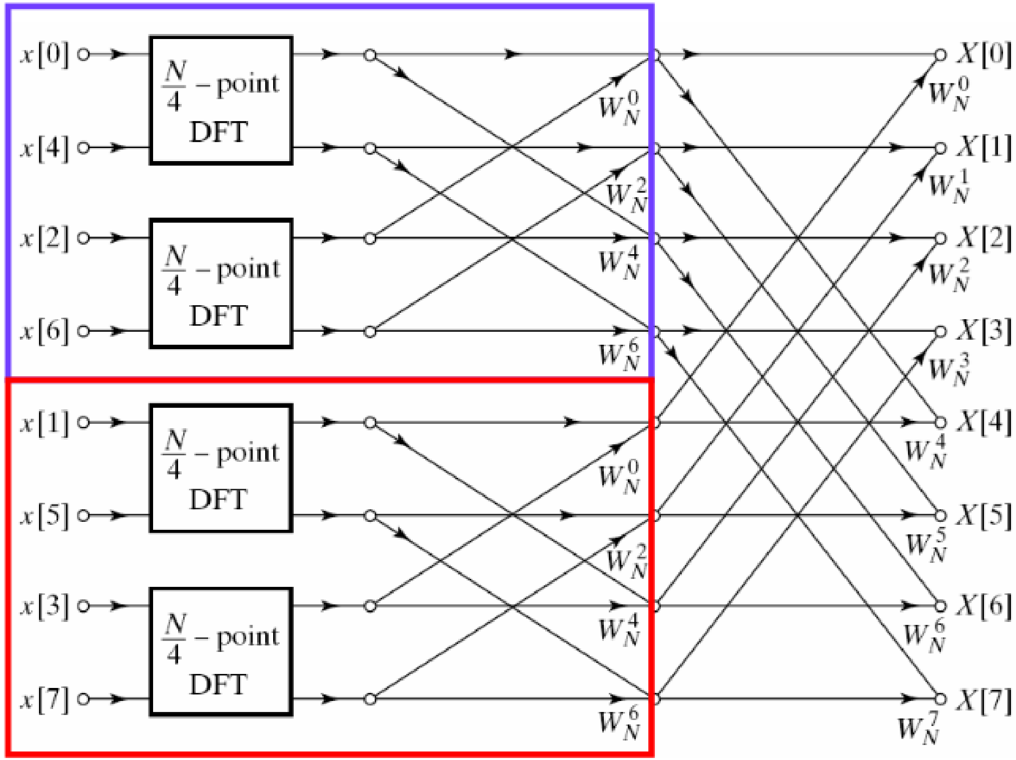
$$= \sum_{l=0}^{{}^{N}/_{4}-1} g(2l)W_{{}^{N}/_{4}}^{lk} + W_{{}^{N}/_{2}}^{k} \sum_{l=0}^{{}^{N}/_{4}-1} g(2l+1)W_{{}^{N}/_{4}}^{lk}$$

- And $H(k)$ as

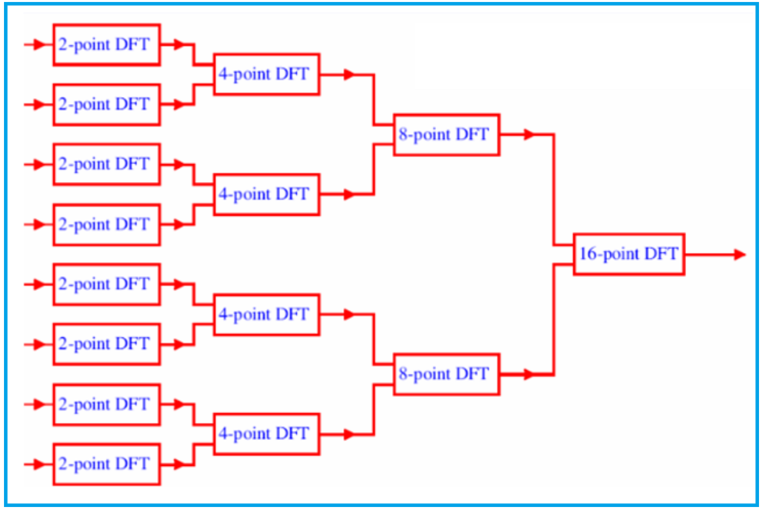
$$H(k) = \sum_{l=0}^{{}^{N}/_{4}-1} h(2l)W_{{}^{N}/_{4}}^{lk} + W_{{}^{N}/_{2}}^{k} \sum_{l=0}^{{}^{N}/_{4}-1} h(2l+1)W_{{}^{N}/_{4}}^{lk}$$

Flow Graph of the DIT FFT



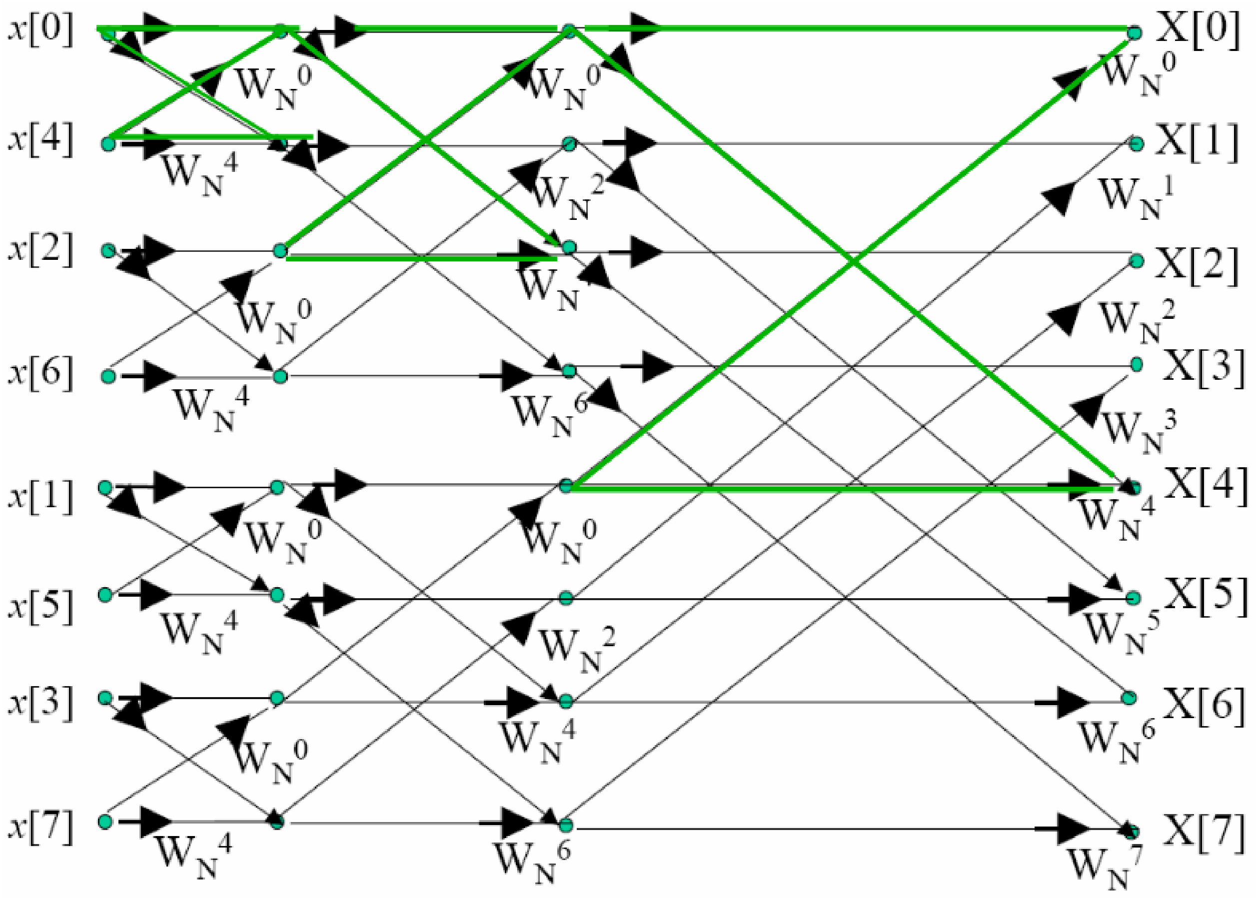


- Any N –point DFT with even N can be computed via two $^N/_2$ –point DFTs.
- In turn, if $^N/_2$ is even then each of these $^N/_2$ –point DFTs can be computed via two $^N/_4$ –point DFTs and so on, such process of "splitting" ends up with all 2-point DFT as shown in figure for 16-point.



- The decomposition of an N –point transform into two $^N/_2$ –point transforms, the number of complex multiplications and additions requires was $N + 2(^N/_2)^2$.
- When $^N/_2$ –point transforms is decomposed into $^N/_4$ –point transforms, the factor $(^N/_2)^2$ is replaced by $^N/_2 + 2(^N/_4)^2$, so the overall computation then requires $N + N + 4(^N/_4)^2$ complex multiplications and additions.
- If $N = 2^v$, this can be done at most $v = \log_2 N$ times, so that after carrying out this decomposition as many times as possible, the number of complex multiplications and additions is equal to $Nv = N \log_2 N$.

8-point DIT DFT



- The computation in the flow graph can be reduced further by exploiting the symmetry and periodicity of the coefficient W_N^r.
- In obtaining a pair values in one stage from a pair of values in the preceding stage, where the coefficients are always powers of W_N and the exponents are separated by ${}^{N}/_{2}$.
- Because of the shape of the flow graph, this elementary computation is called a butterfly.

$$W_N^{r+{}^{N}/_{2}} = W_N^r W_N^{{}^{N}/_{2}} = W_N^r e^{-j\pi} = -W_N^r$$

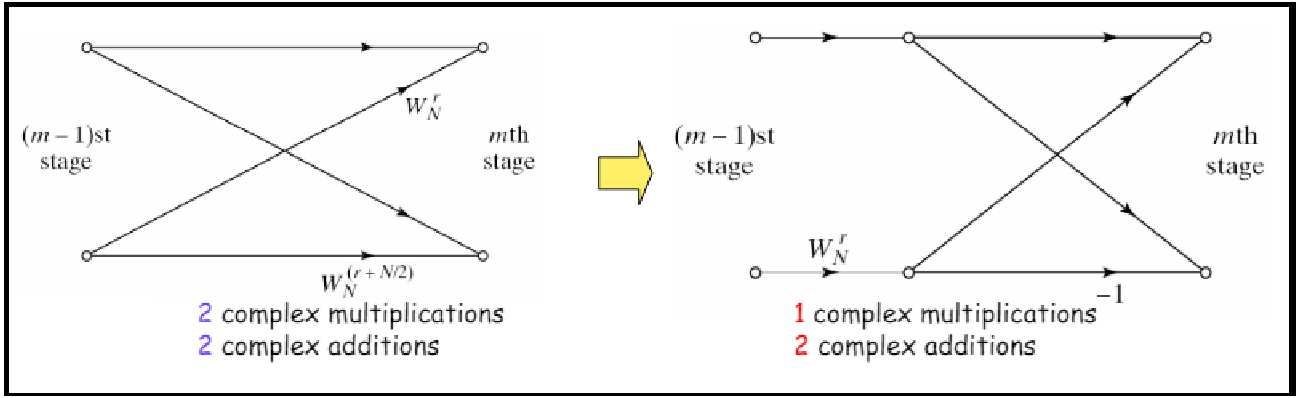
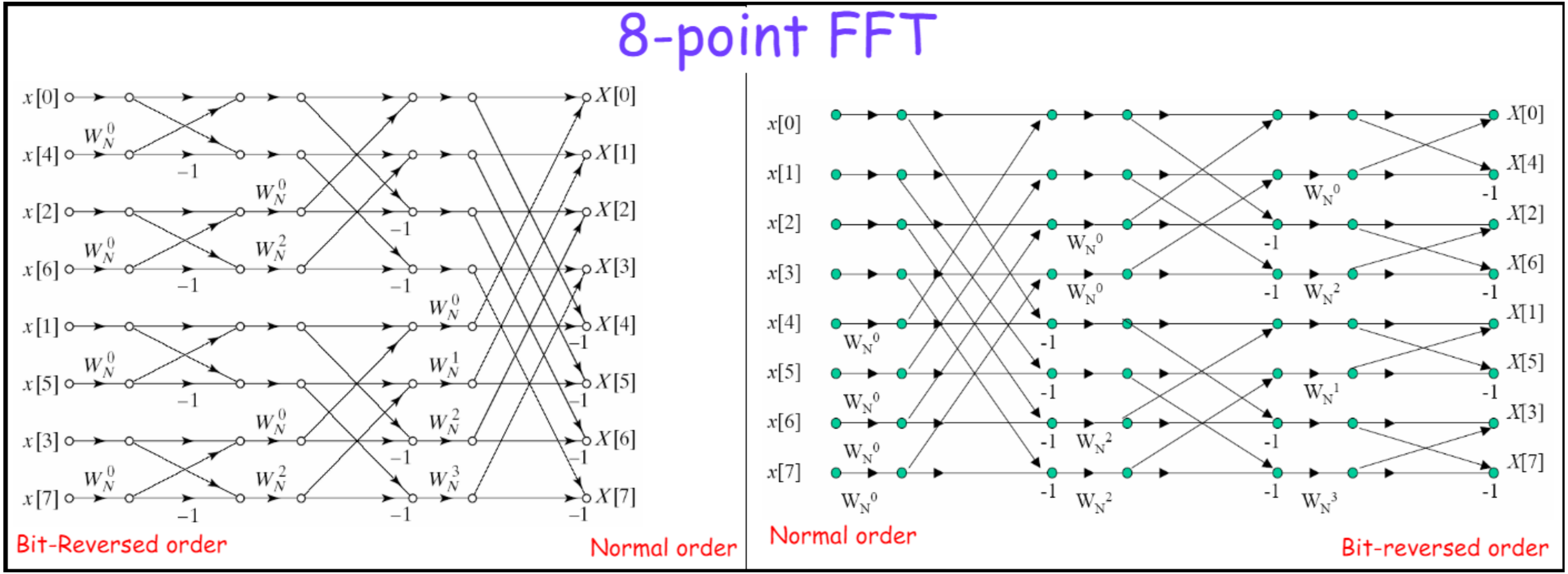
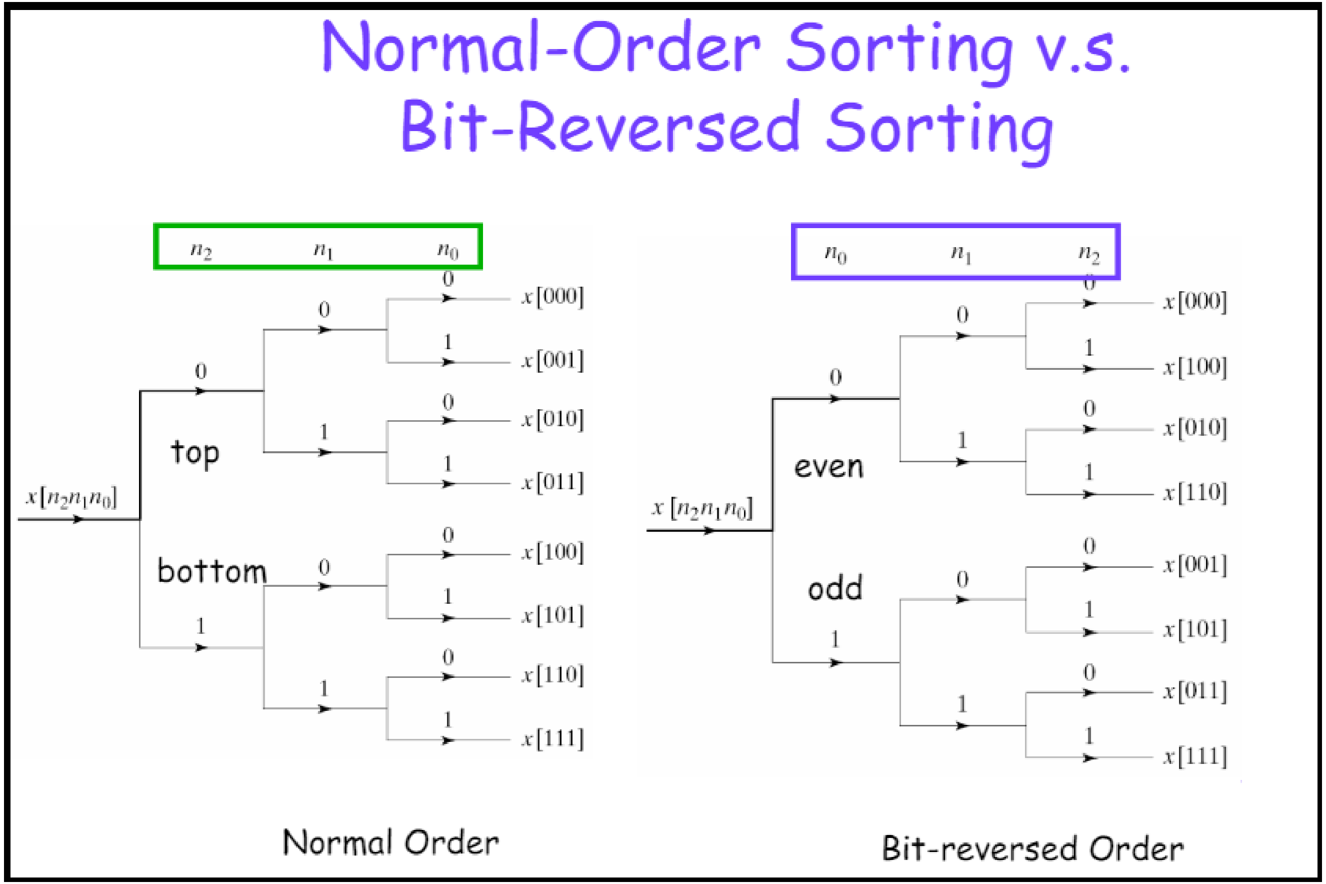


Figure shows a 8-point DIT-FFT left flow graph that shows the input $x(n)$ in bit-revesed order with output $X(k)$ in Normal order, while right shows the input $x(n)$ in normal order with output $X(k)$ in Bit-revesed order.

Note : W_N^r multiply the flow before going up and (-1) after going up.



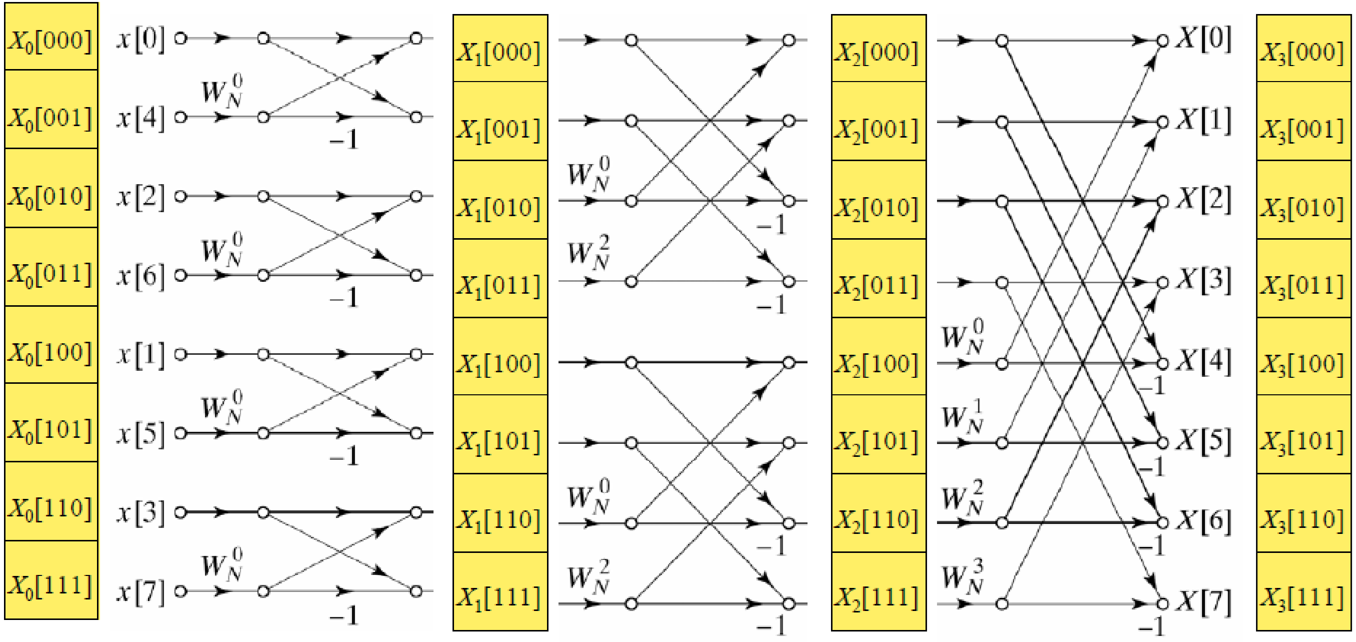
The shorting in Normal and Bit-reversed order for 8-bit FFT is shown below.



Index	Binary	Bit-reversed binary	Bit-reversed index
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

In-Place Computation

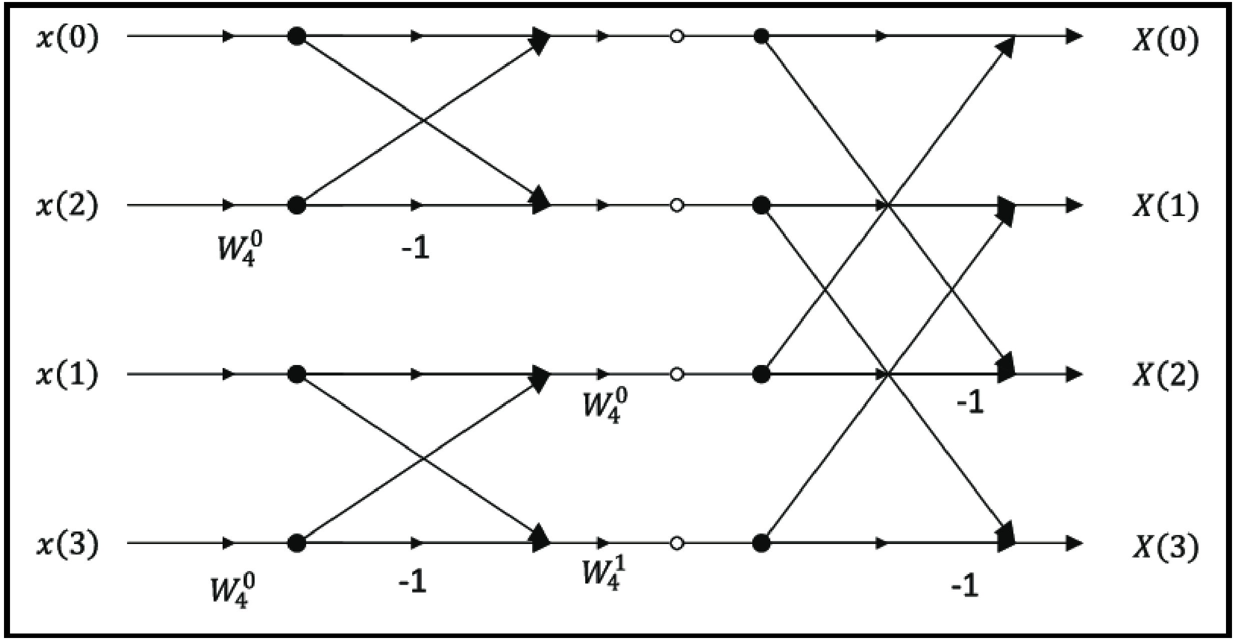
The same register array can be used in each stage



Stage 1 Stage 2 Stage 3

Example 1: Find 4-point DFT of $x(n) = (1, 1, 1, 0)$ using radix-2 DIT-FFT.

Solution: The 4-point DIT-FFT butterfly diagram is shown in figure.



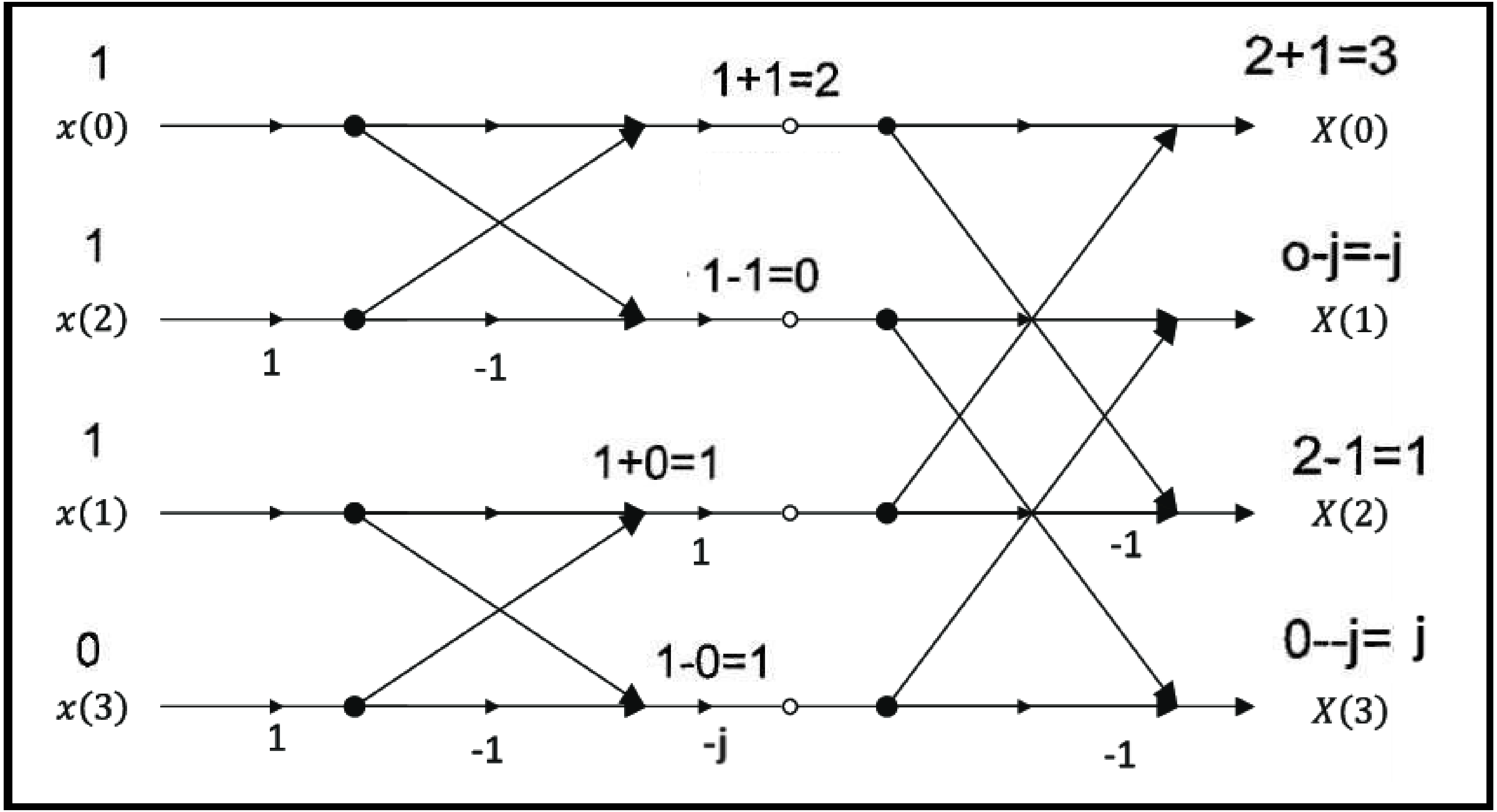
We know,

$W_N^k = e^{-j\frac{2\pi}{N}k}$

Thus,

$W_4^0 == e^{-j\frac{2\pi}{4}\times 0} = e^0 = 1$

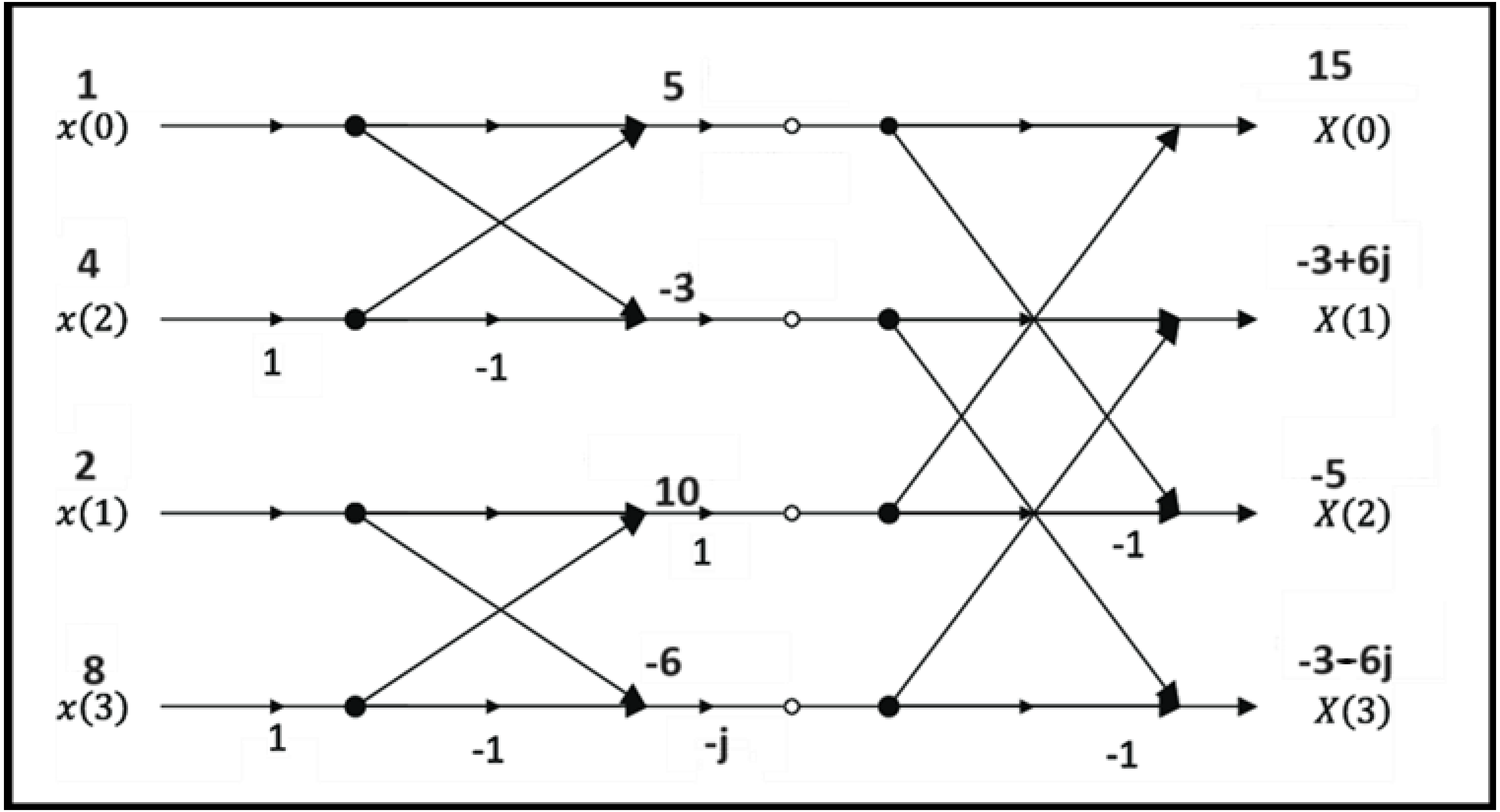
$W_4^1 == e^{-j\frac{2\pi}{4}\times 1} = e^{-j\frac{\pi}{2}} = \cos\frac{\pi}{2} - j\sin\frac{\pi}{2} = -j$


1
x(0)
1
x(2)
1
x(1)
0
x(3)
1
-1
1
-1
1+1=2
1-1=0
1+0=1
1-0=1
1
-j
2+1=3
X(0)
o-j=-j
X(1)
2-1=1
X(2)
0--j= j
X(3)
-1
-1

$$X(k) = (3, -j, 1, j)$$

Question: Find the 4-point DFT of $x(n) = (1, 2, 4, 8)$

$$X(k) = (15, -3 + 6j, -5, -3 - 6j)$$



Example 2: Find 8-point DFT of $x(n) = (2,2,2,2,1,1,1,1)$ using radix-2 DIT-FFT.

Solution

We know,

$W_N^k = e^{-j\frac{2\pi}{N}k}$

[Note: $W_N^2 = W_{N/2}$].

$W_2^0 = W_4^0 = 1$

$W_4^1 = -j$

and,

$W_8^0 = e^{-j\frac{2\pi}{8}\times 0} = e^0 = 1$

$W_8^1 = e^{-j\frac{2\pi}{8}\times 1} = e^{-j\frac{\pi}{4}}$

$= \cos\frac{\pi}{4} - j\sin\frac{\pi}{4}$

$= 0.707 - j0.707$

$W_8^2 = e^{-j\frac{2\pi}{8}\times 2}$

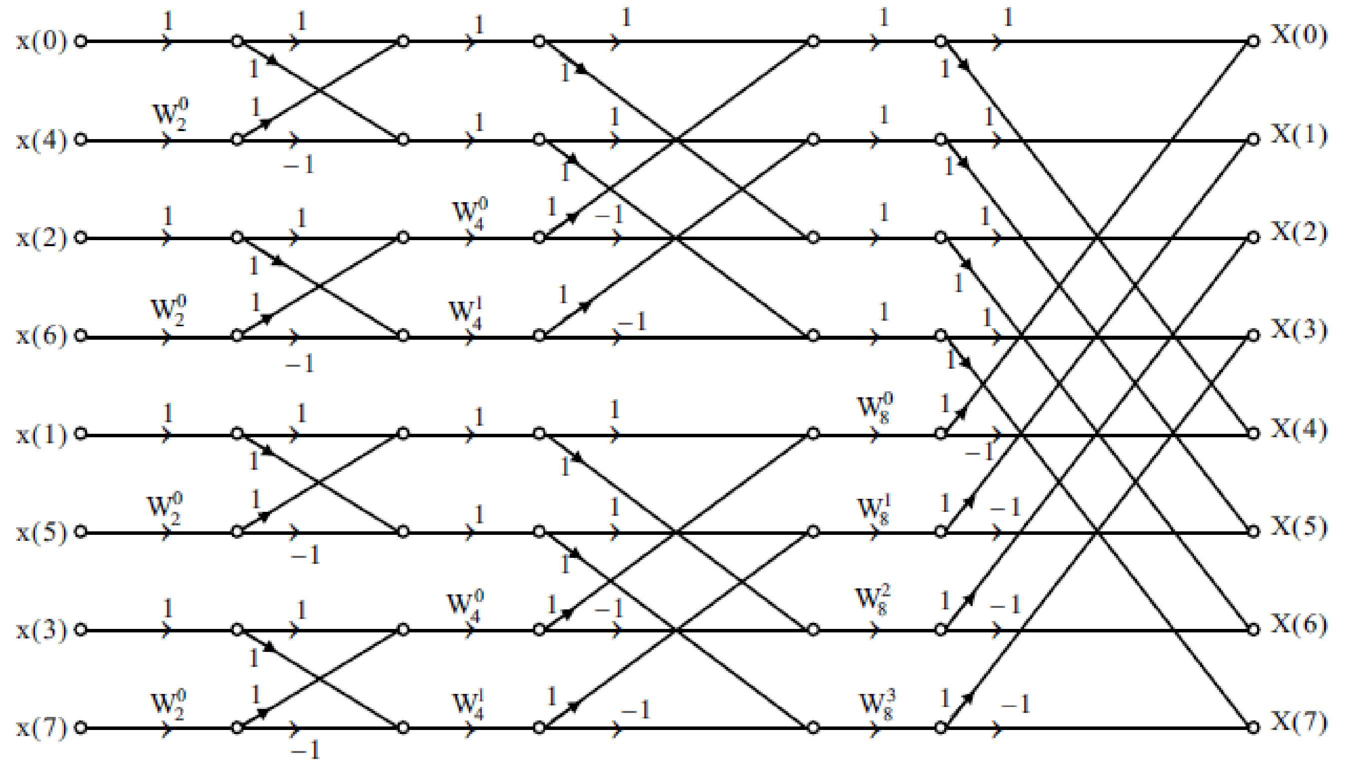
$= \cos\frac{\pi}{2} - j\sin\frac{\pi}{2}$

$= -j$

$W_8^3 = e^{-j\frac{2\pi}{8}\times 3} = e^{-j\frac{3\pi}{4}}$

$= \cos\frac{3\pi}{4} - j\sin\frac{3\pi}{4}$

$= -0.707 - j0.707$



The flow graph (or butterfly diagram) for 8-point DIT radix-2 FFT.

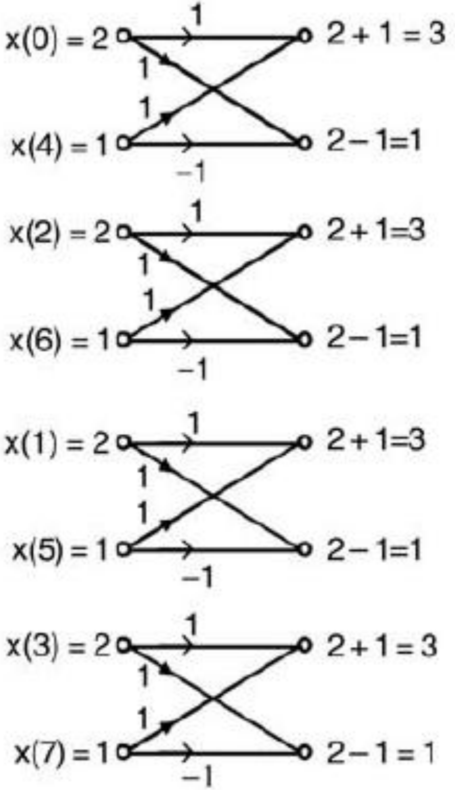


***Fig 1 :** Butterfly diagram for first stage of radix-2 DIT FFT.*

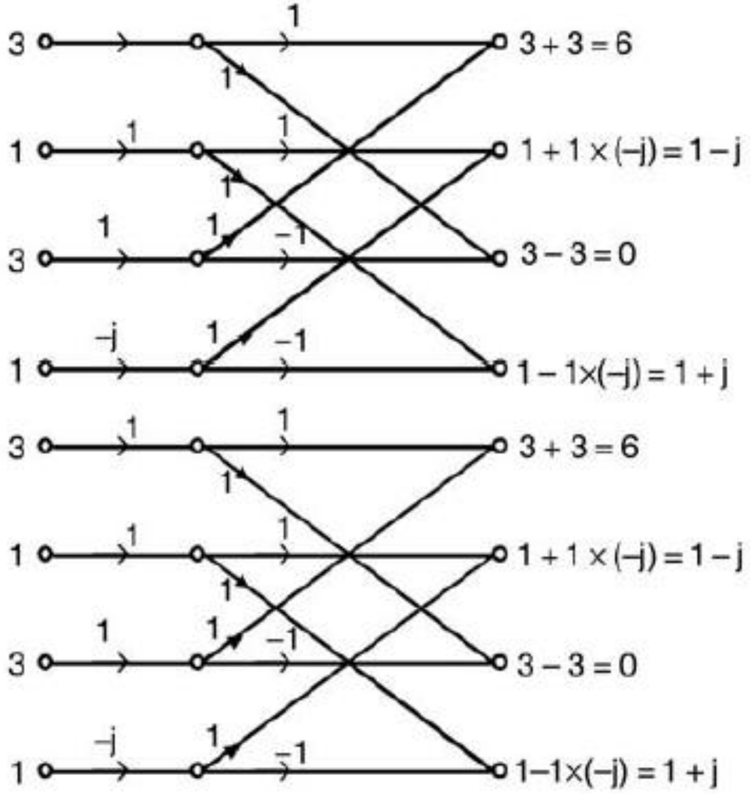
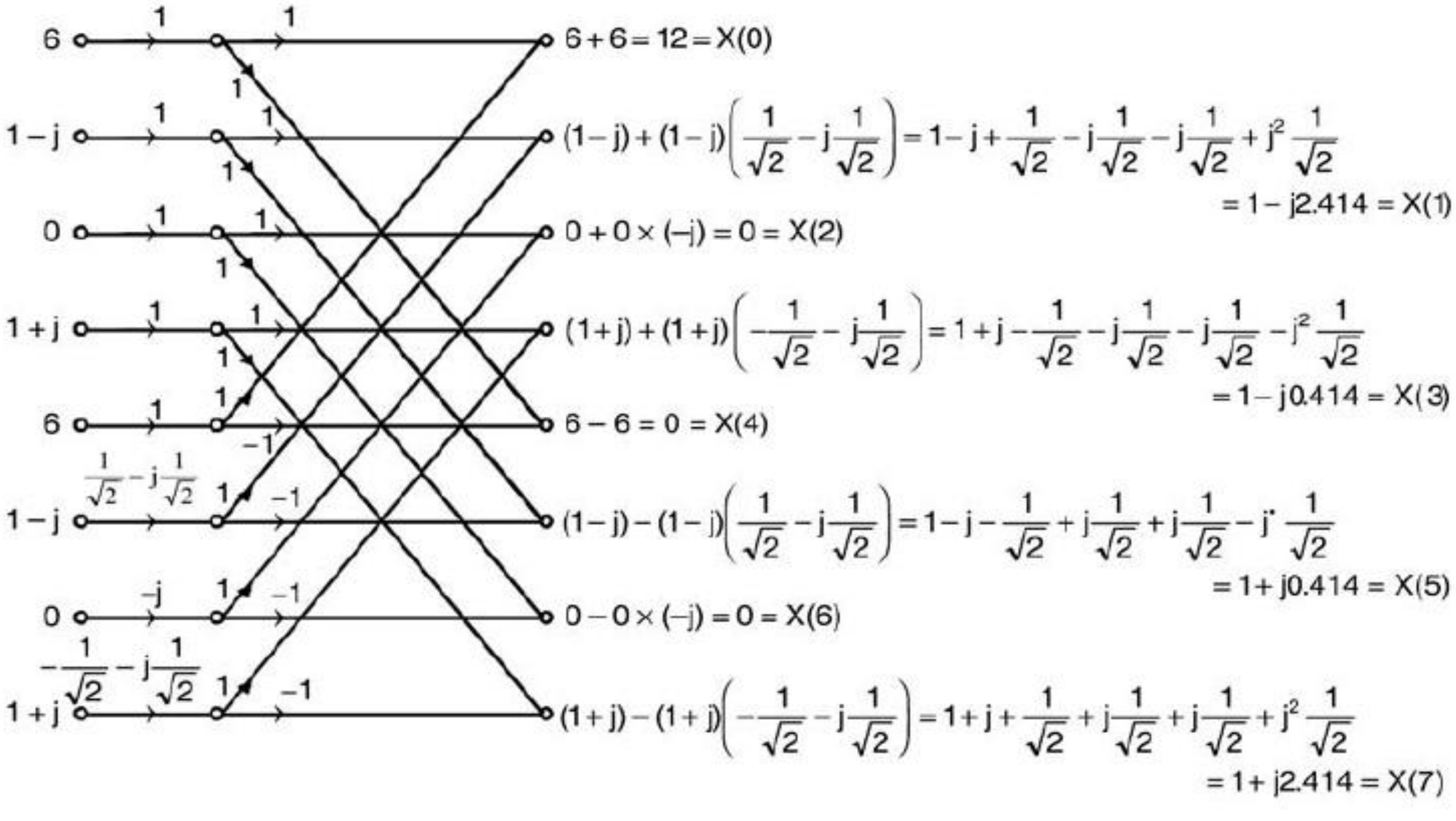


***Fig 2 :** Butterfly diagram for second stage of radix-2 DIT FFT.*



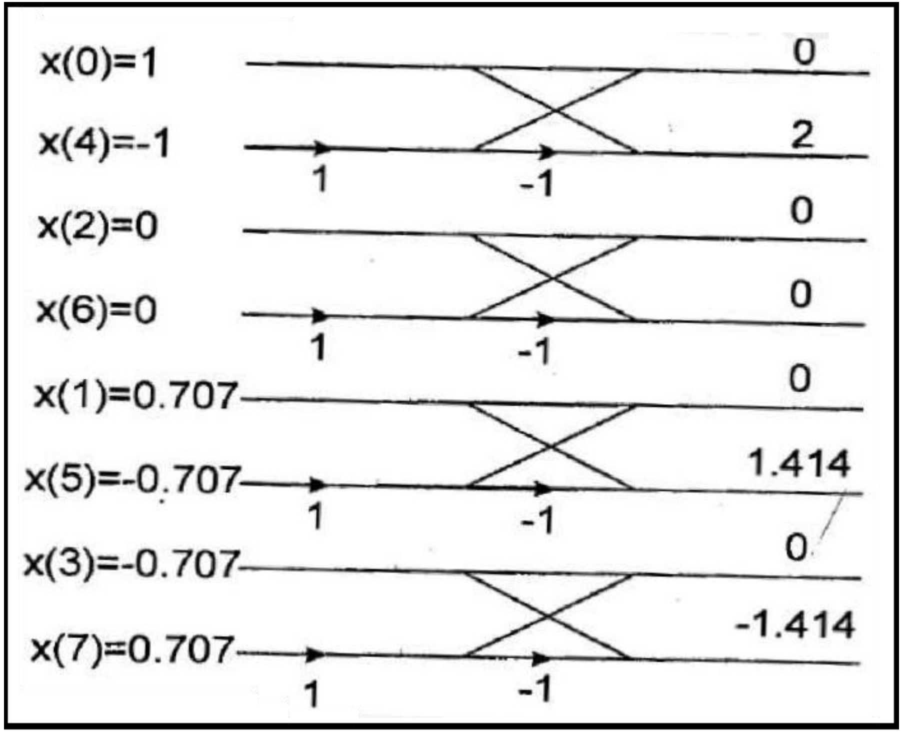
$$X(k) = (12, 1 - j2.424, 0, 1 - j0.414, 0, 1 + j0.414, 0, 1 + j2.424)$$

Example 3: Using DIT-FFT compute DFT of $x(n) = \cos\left(\frac{n\pi}{4}\right),\ 0 \le n \le 7$

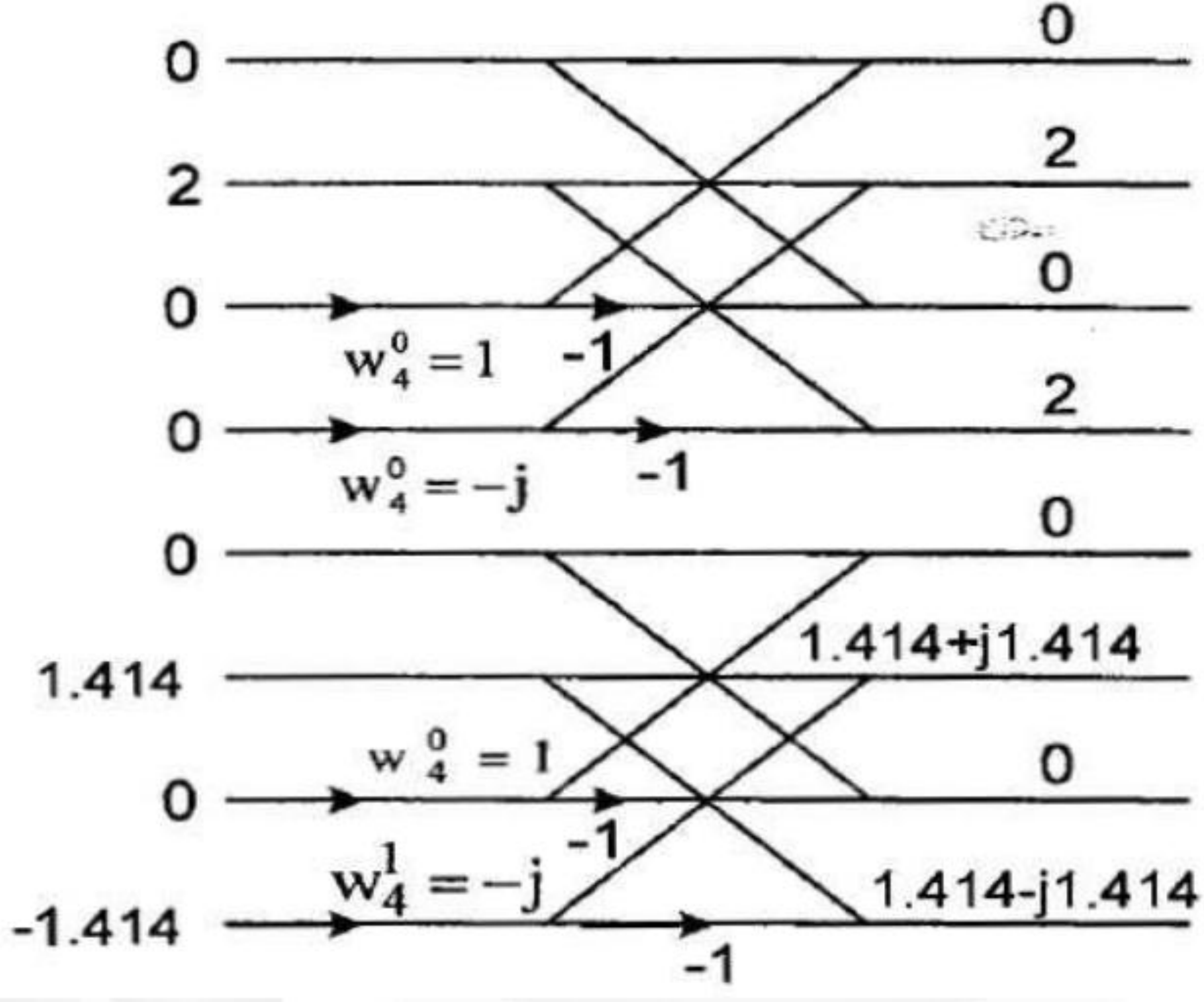
Solution: $N = 8, x(n) = \cos\left(\frac{n\pi}{4}\right),\ 0 \le n \le 7$

Thus, $x(n) = (1,\ 0.707,\ 0,\ -0.707,\ -1,\ -0.707,\ 0,\ 0.707)$

First stage of computation as in figure.

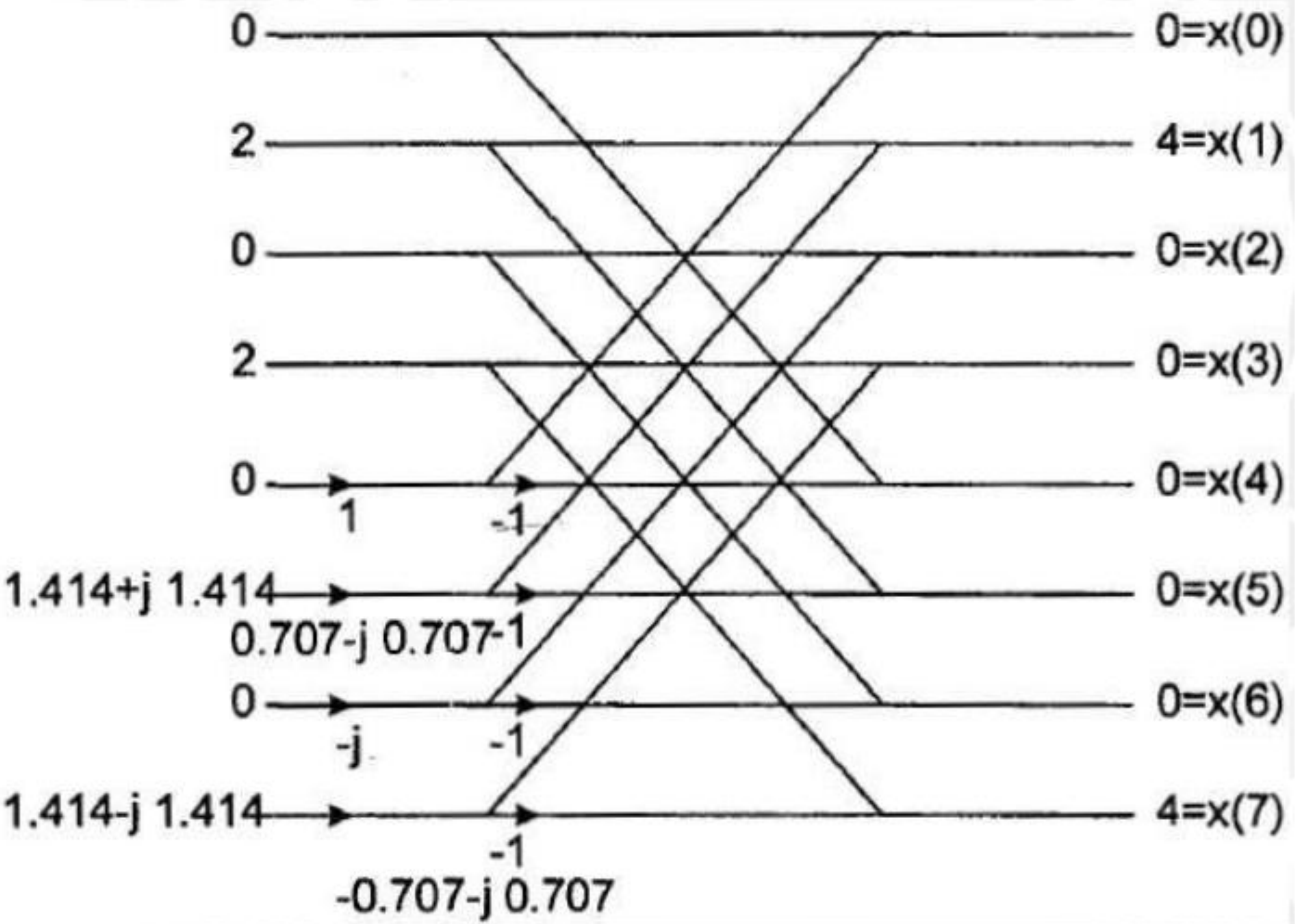


Second stage Computation as in figure



Third stage of computation as in figure.

$$X(k) = (0, 4, 0, 0, 0, 0, 0, 4)$$



Example 4: Compute 8-point DFT of the sequence using DIT FFT

$x(n) = (1, 2, \underset{\uparrow}{\mathbf{3}}, 2, 1, 0)$

Here $x(-2) = 1, x(-1) = 2,\ x(0) = 3, x(1) = 2,\ x(2) = 1,\ x(3) = 3$

This is 6-point so we have to add 2 point with zeros.

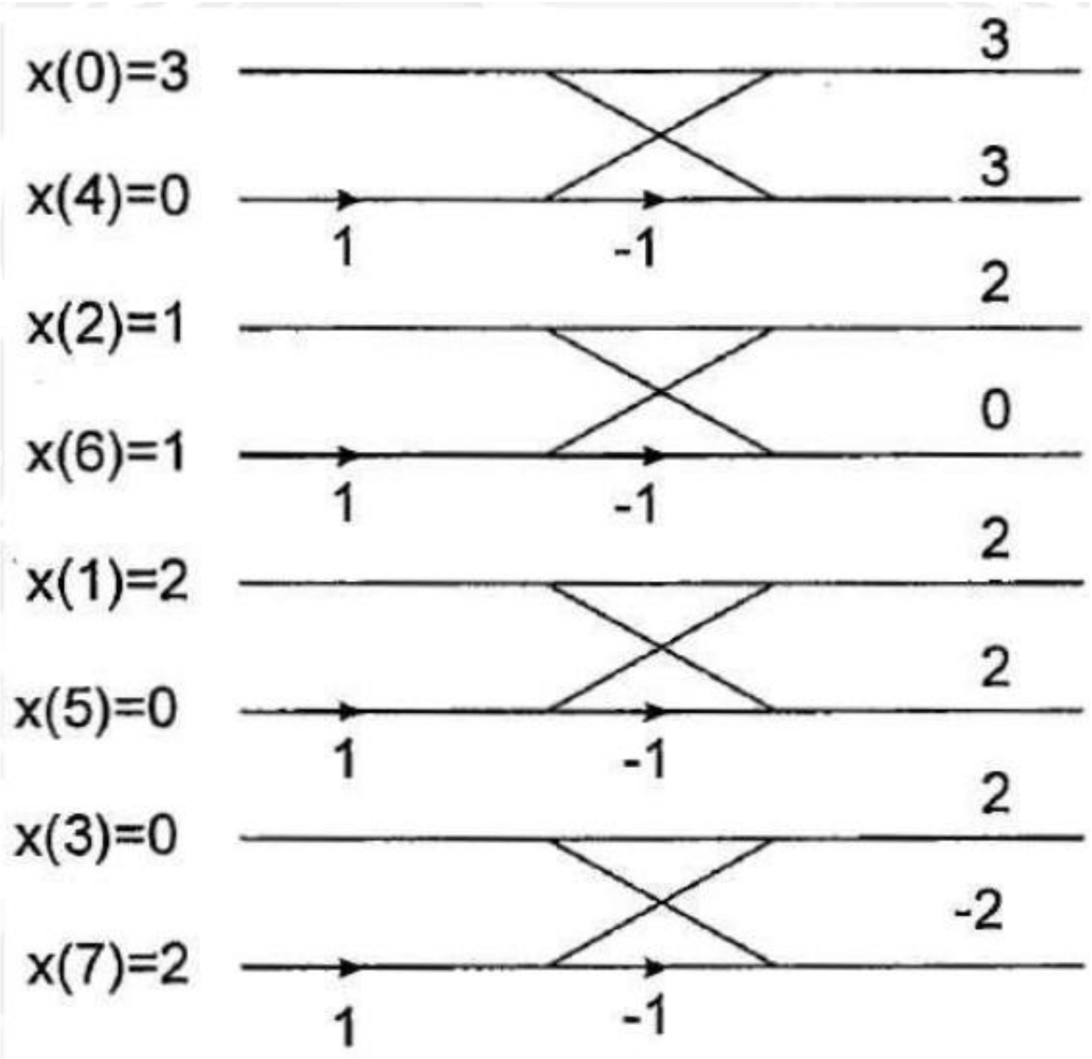
$x(-2) = 1, x(-1) = 2,\ x(0) = 3, x(1) = 2,\ x(2) = 1,\ x(3) = 0, x(4) = 0, x(5) = 0$

Since $x(n - N) = x(n)$ so

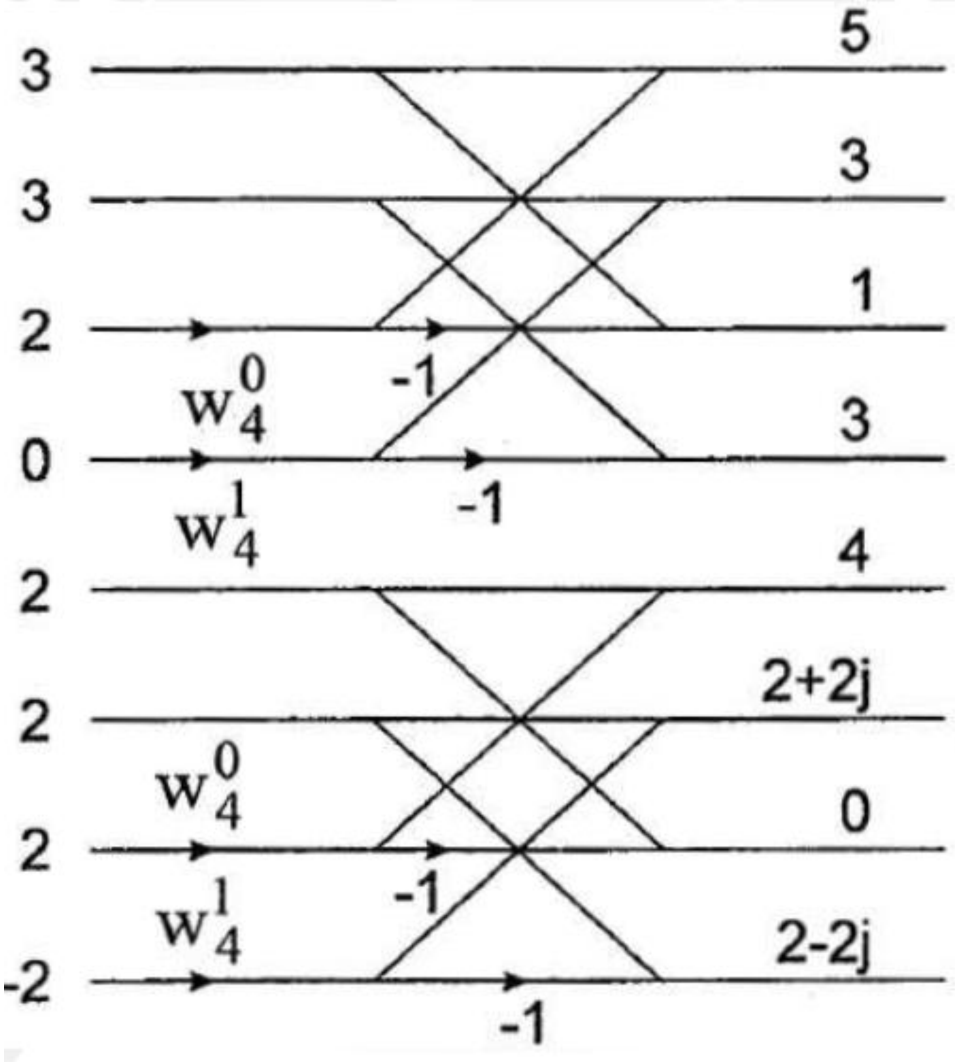
$x(0) = 3, x(1) = 2,\ x(2) = 1,\ x(3) = 0,\ \ x(4) = 0,\ \ x(5) = 0,\ x(6) = 1,$
$x(7) = 2,$

$$x(n) = (3, 2, 1, 0, 0, 0, 1, 2)$$

First stage of computation

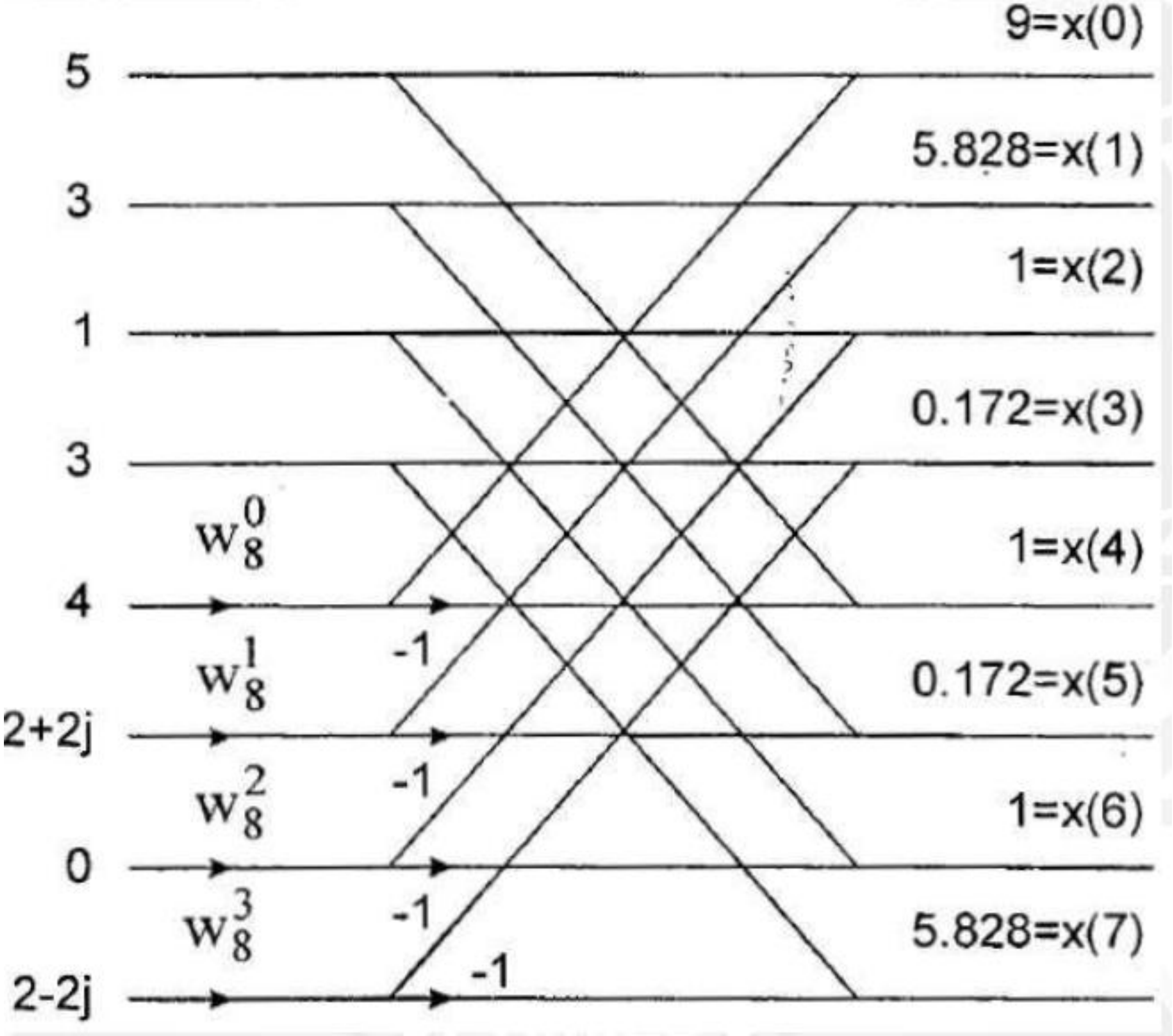


Second stage of computation



Third stage of computation

$$X(k) = (9, 5.828, 1, 0.172, 1, 0.172, 1, 5.828)$$



Questions 1: Find 8-point DIT-FFT of sequence

$$x(n) = u(n) + u(n-2) - u(n-6) - u(n-8)$$

Solution :

$x(0) = u(0) + u(-2) - u(-6) - u(-8) = 1,$ $x(1) = u(1) + u(-1) - u(-5) - u(-7) = 1$

$x(2) = u(2) + u(0) - u(-4) - u(-6) = 2,$ $x(3) = u(3) + u(1) - u(-3) - u(-5) = 2$

$x(4) = u(4) + u(2) - u(-2) - u(-4) = 2,$ $x(5) = u(5) + u(3) - u(-1) - u(-3) = 2$

$x(6) = u(6) + u(4) - u(0) - u(-2) = 1,$ $x(7) = u(7) + u(5) - u(1) - u(-1) = 1$

$$x(n) = (1, 1, 2, 2, 2, 2, 1, 1)$$

Using DIT-FFT $X(k) = (12, -2.4 + j, 0, 0.41 - j, 0, 0.41 + j, 0, -2.4 - j)$

Question 2: Find 4-point DFT on a continuous input signal given by

$$x(t) = \sin(2\pi 100t) + \sin(2\pi 2000t + \frac{3\pi}{4})$$

With sampling frequency $f_s = 8KHz$

Solution: With sampling at $f_s = 8KHz$, $x(n) = x_a(\frac{n}{f_s})$, thus

$$x(n) = \sin\left(2\pi 100 \frac{n}{8000}\right) + \sin(2\pi 2000 \frac{n}{8000} + \frac{3\pi}{4})$$

$$x(n) = (0.707, 0, 2.92, 1.414)$$

DFT $X(k) = (5.041, -2.213 + j1.414,\ 2.313,\ -2.213 - j1.414)$

Decimation In Frequency FFT algoritms (DIF-FFT)

- The decimation-in-time FFT algorithms are all based on structuring the DFT computation by froming smaller and smaller subsequences of the input sequence $x(n)$.
- Alternatively, we consider dividing the output sequence $X(k)$ into smaller and smaller subsequences in the same manner.
- FFT algorithms based on this procedure are known as decimation-in-frequency algorithms.
- Let N be a power of 2 and consider computing separately the even-numbered frequency samples and the odd-numbered frequency samples. Since,

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}\,, \qquad for\; k = 0,1,2,\dots,N-1$$

- The even-numbered frequency samples are

$$X(2r) = \sum_{n=0}^{N-1} x(n)W_N^{n2r}\,, \qquad for\; r = 0,1,2,\dots,({}^{N}/_{2}) - 1$$

Which can be expresses as

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

Changing the summation of second variables as

$$X(2r) = \sum_{n=0}^{N/2-1} x(n)W_N^{2nr} + \sum_{n=0}^{N/2-1} x\left(n + {}^{N}/_{2}\right)W_N^{2r(n+N/2)}$$

And $W_N^{2r(n+N/2)} = W_N^{2rn}W_N^{rN} = W_N^{2rn}e^{-j(2\pi/N)rN} = W_N^{2rn}e^{-j2\pi r} = W_N^{2rn}$

With $W_N^2 = e^{-2j(2\pi/N)} = e^{-j\frac{2\pi}{N/2}} = W_{N/2}$ the equation becomes

$$X(2r) = \sum_{n=0}^{N/2-1} \left[x(n) + x\left(n + {}^{N}/_{2}\right)\right]W_{N/2}^{rn}\ , \qquad r = 0, 1, \dots, ({}^{N}/_{2}) - 1$$

The even $({}^{N}/_{2})$ −point DFT of the $({}^{N}/_{2})$ −point sequence obtained by adding the first half and the last half of the input sequence.

Now obtaining the odd-numbered frequency points as

$$X(2r+1) = \sum_{n=0}^{N-1} x(n) W_N^{n(2r+1)}, \qquad for\ r = 0,1,2,\ldots,({}^{N}\!/_{2}) - 1$$

Which can be expresses as

$$X(2r+1) = \sum_{n=0}^{{}^{N}\!/_{2}-1} x(n) W_N^{n(2r+1)} + \sum_{n={}^{N}\!/_{2}}^{N-1} x(n) W_N^{n(2r+1)}$$

Changing the summation of second variables as

$$X(2r+1) = \sum_{n=0}^{{}^{N}\!/_{2}-1} x(n) W_N^{n(2r+1)} + \sum_{n=0}^{{}^{N}\!/_{2}-1} x\left(n + {}^{N}\!/_{2}\right) W_N^{(2r+1)(n+{}^{N}\!/_{2})}$$

$$W_N^{(2r+1)(N/2)} = W_N^{(rN)} W_N^{(N/2)} = e^{-j(2\pi/N)rN} e^{-j(2\pi/N)N/2} = e^{-j(2\pi)r} e^{-j\pi} = -1$$

Thus,

$$X(2r+1) = \sum_{n=0}^{N/2-1} x(n) W_N^{n(2r+1)} - \sum_{n=0}^{N-1} x\left(n + N/2\right) W_N^{(2r+1)n}$$

$$X(2r+1) = \sum_{n=0}^{N/2-1} \left[x(n) - x\left(n + N/2\right)\right] W_N^n W_{N/2}^{rn}, \qquad r = 0, 1, \ldots, (N/2) - 1$$

The odd $(N/2)$ −point DFT of the $(N/2)$ −point sequence obtained by subtraction the second half from the first half and multiplying the resulting sequences by W_N^n.

Let,

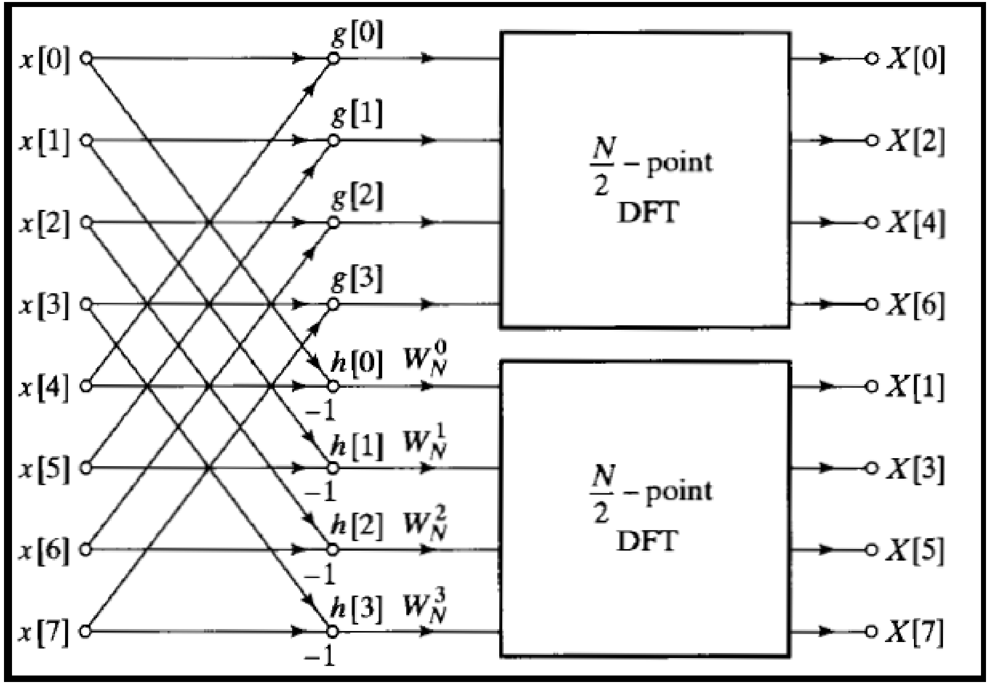
$$g(n) = x(n) + x\left(n + N/2\right)$$

And

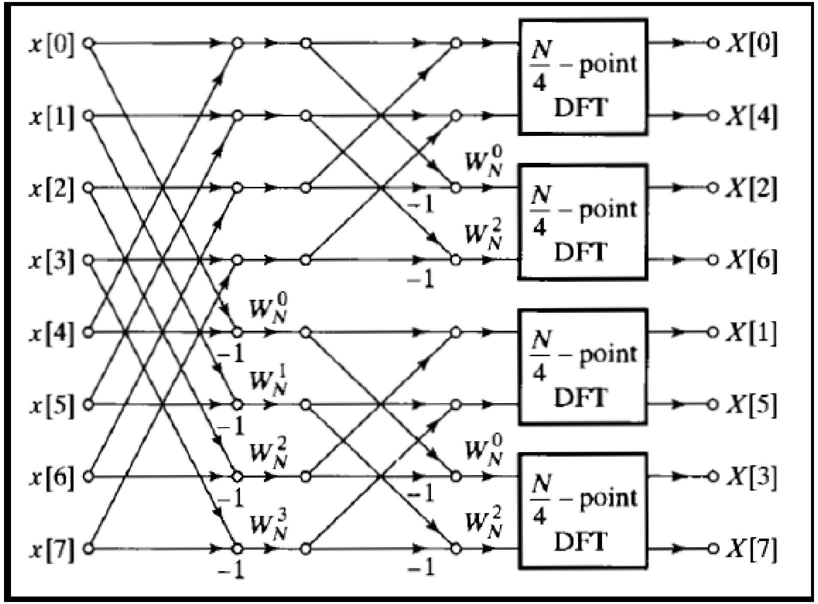
$$h(n) = x(n) - x\left(n + N/2\right)$$

The DFT can be computed by first forming the sequences $g(n)$ and $h(n)$, then computing $h(n)W_N^n$, finally computing the $(^N/_2)$ −point DFTs of these two sequences to obtain the even-numbered output points and the odd-numbered outputs, respectively. The procedure of 8-point DFT is as in figure.

Since N is a power of 2, $^N/_2$ is even. Consequently, the $^N/_2$ −point DFTs can be computed by computing the even-numbered and odd-numbered output for those DFTs separately.
This is accomplish by combining the first half and last half of the input points for each of the $^N/_2$ −point DFTs and then computing $^N/_4$ −points DFTs.

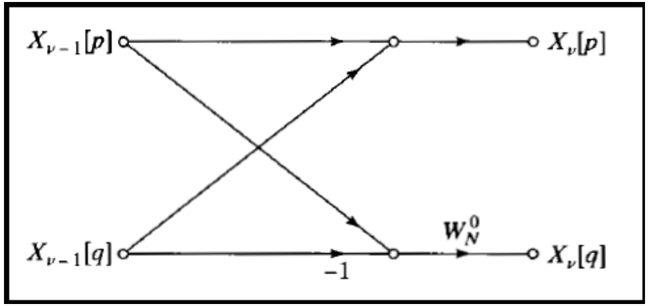


The flow graph resulting from taking $^N/_4$ −point DFTs for 8-point DFT is shown in figure.

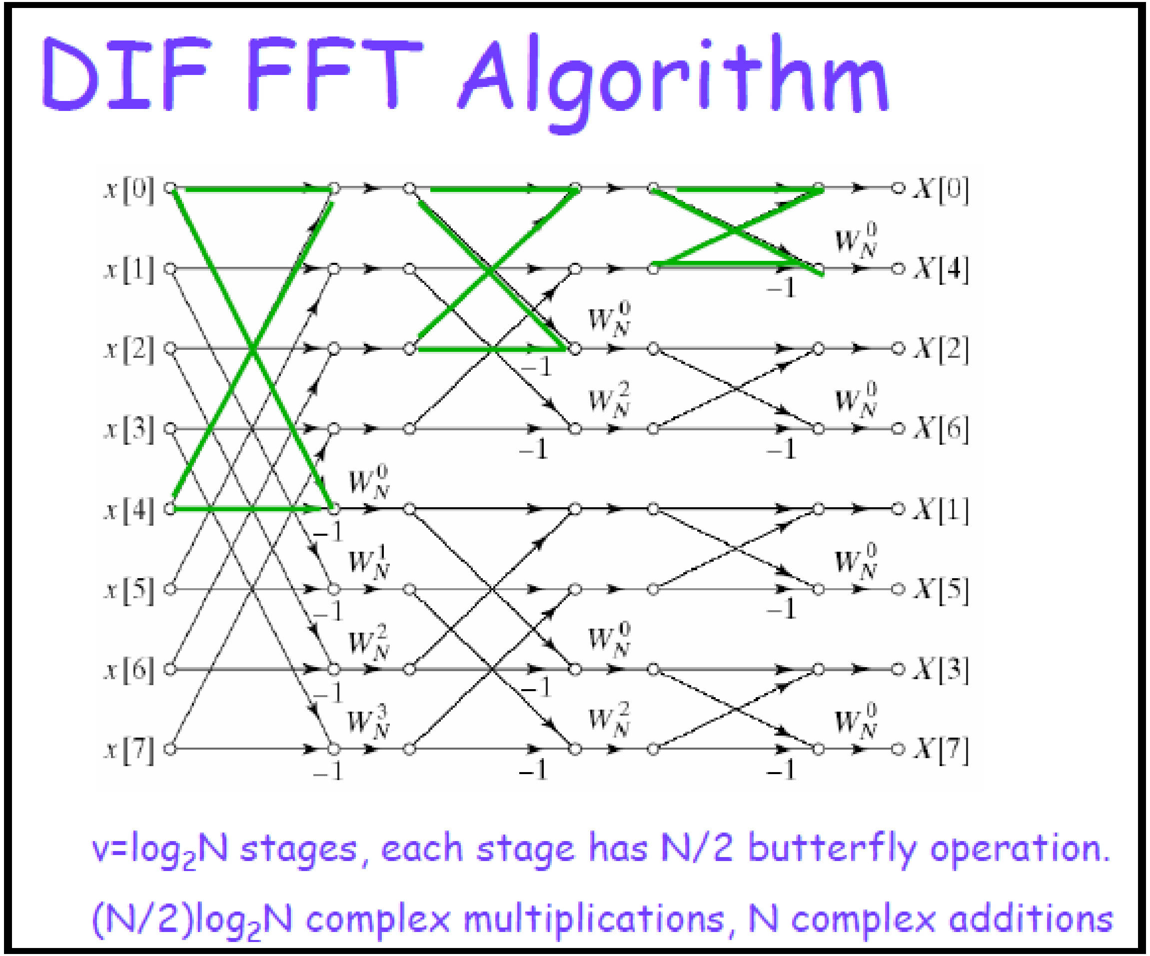


For this 8-point DFT, the computation has now been reduced to the computations of 2-point DFTs, which are implemented by adding and subtracting the input points.

Thus, the two point DFTs replaced by the following figure.

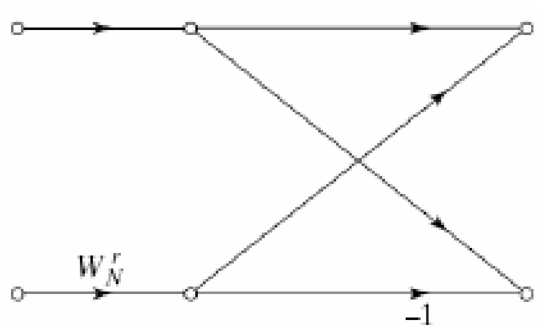


Thus the computation of 8-point DFT can be accomplishe by the algorithm as in figure

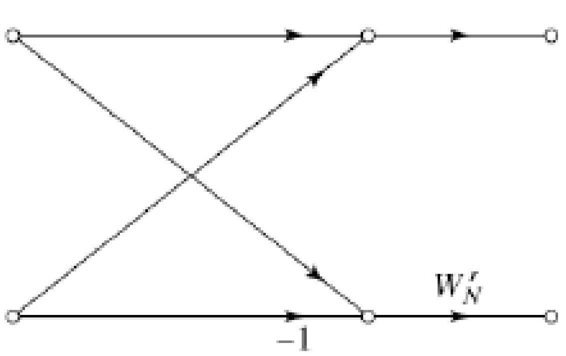


Note:

- The basic butterfly operations for DIT FFT and DIF FFT respectively are transposed-form pair.



DIT BF unit



DIF BF unit

- The I/O values of DIT FFT and DIF FFT are the same
- Applying the transpose transform to each DIT FFT algorithm, one obtains DIF FFT algorithm

Comparision of DIT and DIF Radix-2 FFT

Difference in DIT and DIF

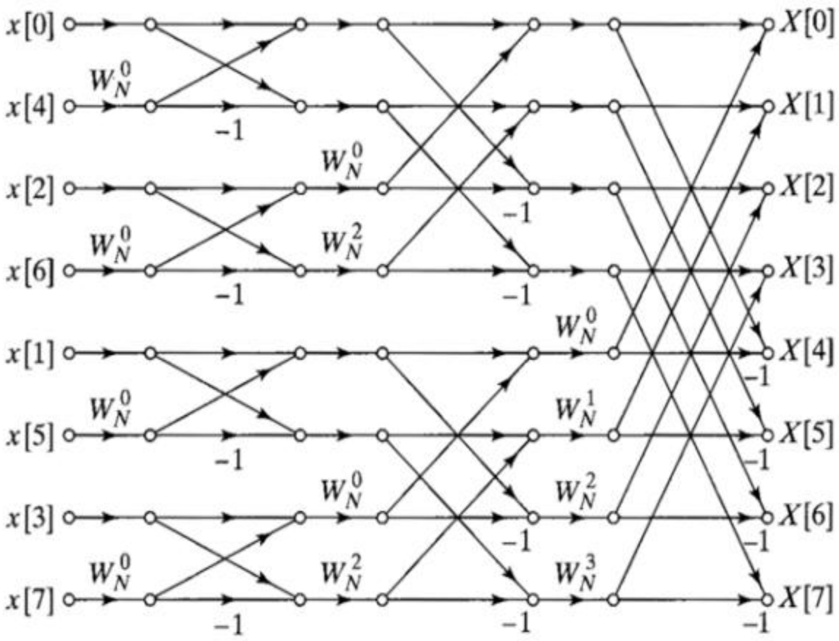
- In DIT the time domain sequence is decimated, while in DIF the frequency domain sequence is decimated.
- In DIT the input should be in bit-reversed order and the output will be in normal order. For DIF the reverse in true, i.e., input is normal order, while output is bit reversed.
- Considering the butterfly diagram, in DIT the compelx multiplication takes place before the add-subtract operation, whereas, in DIF the complex multiplication takes place after the add-subtract operation.

Similarities in DIT and DIF

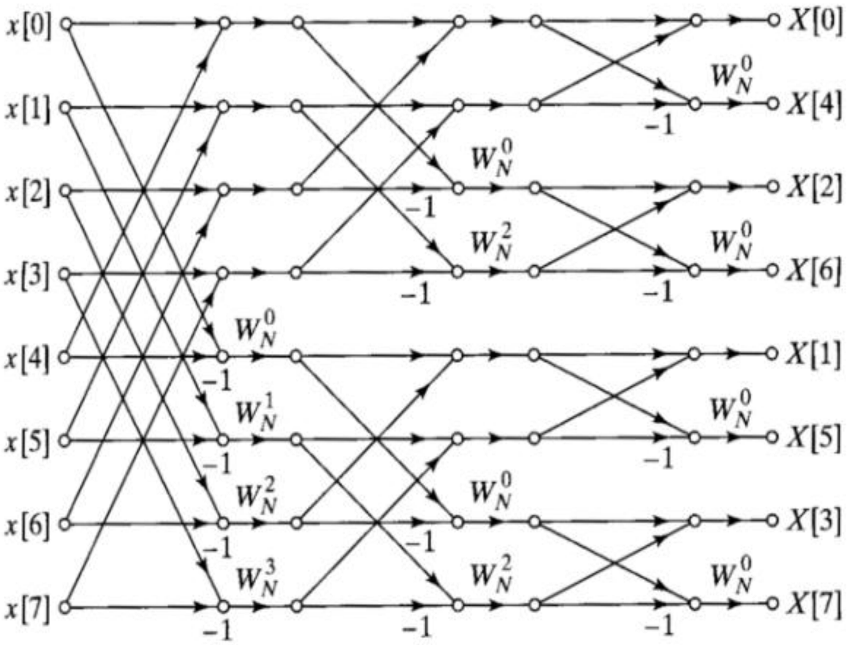
- For both the algorithms tha values of N should be such that, $N = 2^v$, and there will be v stages of butterfly computations.
- Both algorithms involve same number of operations. The total number of complex additions are $N \log_2 N$ and total number of complex multiplication are $({}^{N}/_{2}) \log_2 N$.
- Both algorithms require bit reversal at some place during computation.

■ Comparing DIT and DIF structures:

DIT FFT structure:



DIF FFT structure:



Example: Compute the 8-point DFT of the sequence

$$x(n) = (2, 2, 2, 2, 1, 1, 1, 1)$$

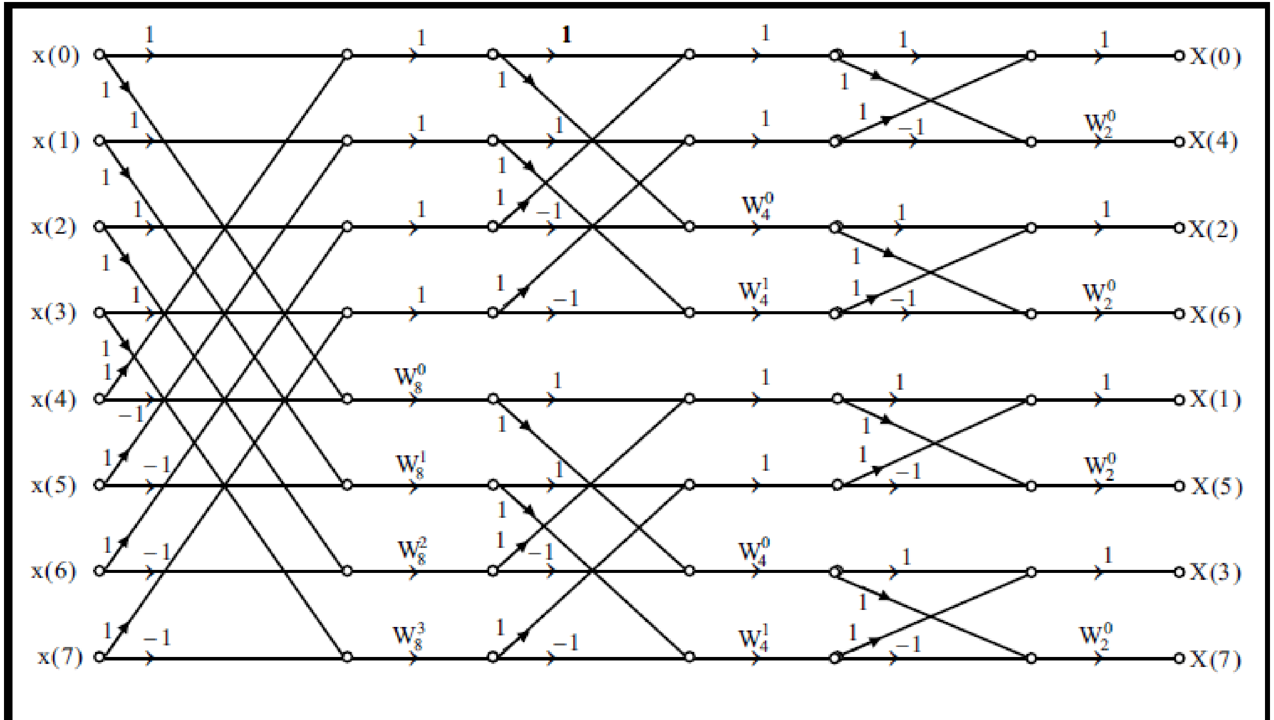
Using radix-2 DIF FFT algorithm:

Solution:

Here, input should be in normal order, output will be in bit reversed order.

The butterfly factors involves in the computations are $W_8^0, W_8^1, W_8^2, W_8^3$ in first stage, $W_8^0(W_4^0)$ and $W_8^2(W_4^1)$ in second stage and $W_8^0(W_2^0)$.

[Note: $W_N^2 = W_{N/2}$].



The flow graph (or butterfly diagram) for 8-point DIF radix-2 FFT.

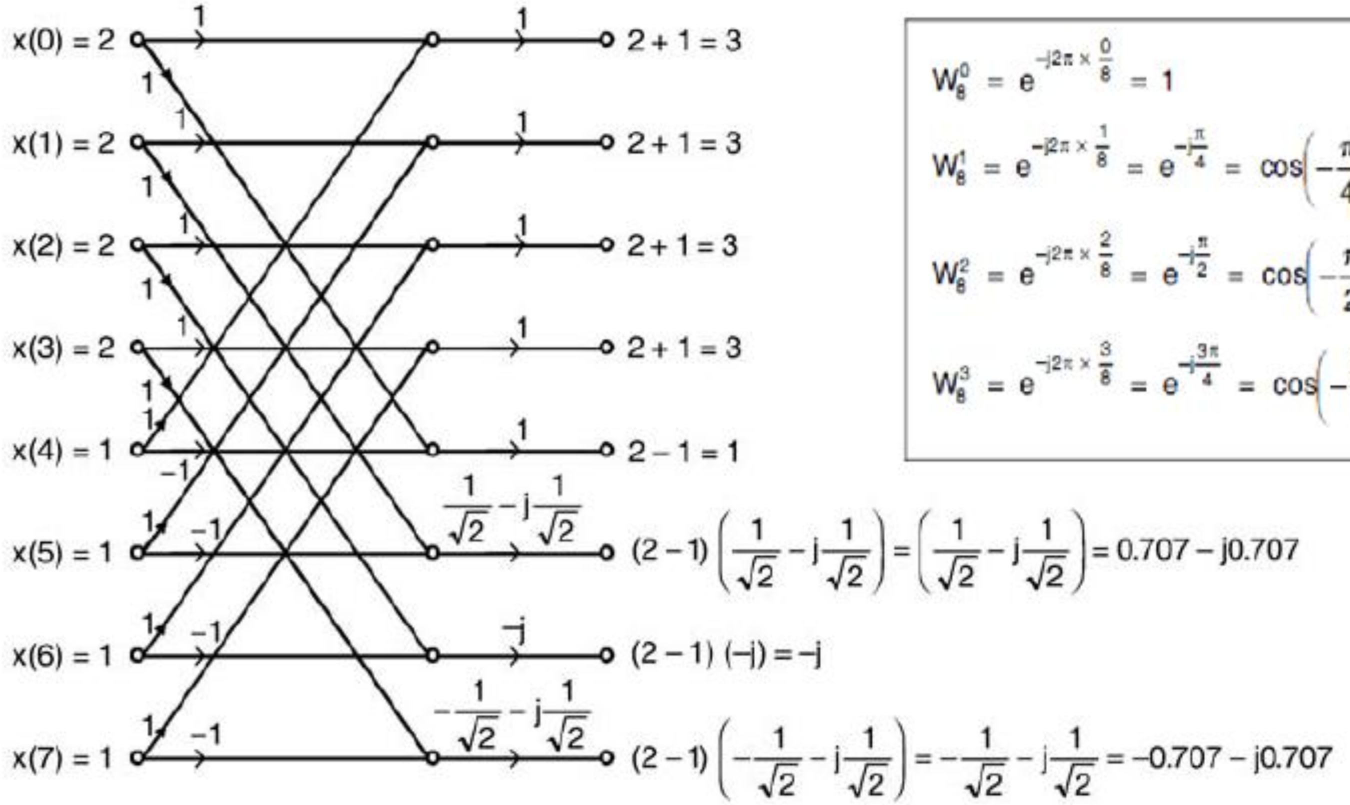
First stage computation

$$W_8^0 = e^{-j2\pi \times \frac{0}{8}} = 1$$

$$W_8^1 = e^{-j2\pi \times \frac{1}{8}} = e^{-j\frac{\pi}{4}} = \cos\left(-\frac{\pi}{4}\right) + j\sin\left(-\frac{\pi}{4}\right) = \frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$$

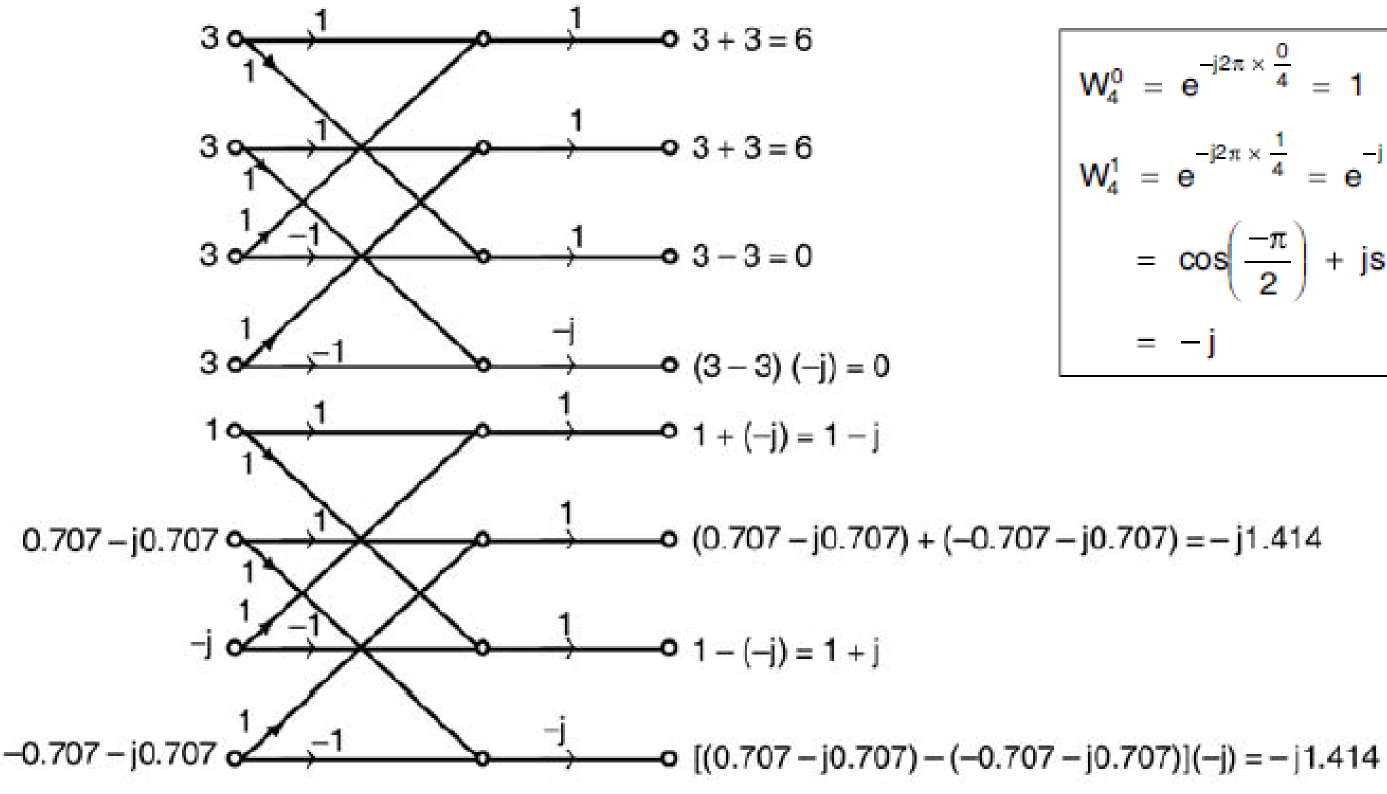
$$W_8^2 = e^{-j2\pi \times \frac{2}{8}} = e^{-j\frac{\pi}{2}} = \cos\left(-\frac{\pi}{2}\right) + j\sin\left(-\frac{\pi}{2}\right) = -j$$

$$W_8^3 = e^{-j2\pi \times \frac{3}{8}} = e^{-j\frac{3\pi}{4}} = \cos\left(-\frac{3\pi}{4}\right) + j\sin\left(-\frac{3\pi}{4}\right) = -\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}$$



Butterfly diagram for first stage of radix-2 DIF FFT.

Second stage computation



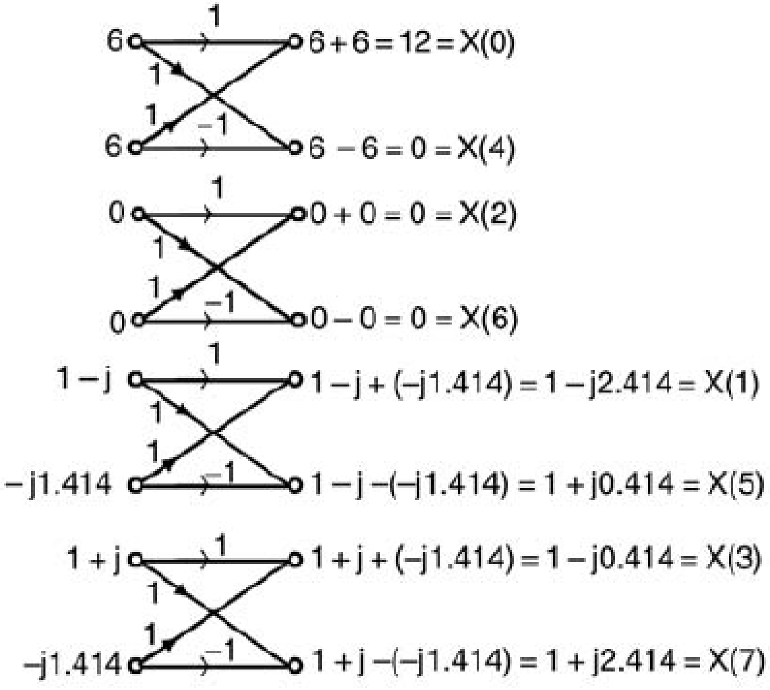
$$W_4^0 = e^{-j2\pi \times \frac{0}{4}} = 1$$

$$W_4^1 = e^{-j2\pi \times \frac{1}{4}} = e^{-j \times \frac{\pi}{2}}$$

$$= \cos\left(\frac{-\pi}{2}\right) + j\sin\left(\frac{-\pi}{2}\right)$$

$$= -j$$

Butterfly diagram for second stage of radix-2 DIF FFT.



Butterfly diagram for third stage of radix-2 DIF FFT.

The sequence X(k) in bit reversed order	The sequence X(k) in normal order
$X(0) = 12$	$X(0) = 12$
$X(4) = 0$	$X(1) = 1 - j2.414$
$X(2) = 0$	$X(2) = 0$
$X(6) = 0$	$X(3) = 1 - j0.414$
$X(1) = 1 - j2.414$	$X(4) = 0$
$X(5) = 1 + j0.414$	$X(5) = 1 + j0.414$
$X(3) = 1 - j0.414$	$X(6) = 0$
$X(7) = 1 + j2.414$	$X(7) = 1 + j2.414$

$$X(k) = (12, 1 - j2.414, 0, 1 - j0.414, 0, 1 + j0.414, 0, 1 + j2.414)$$

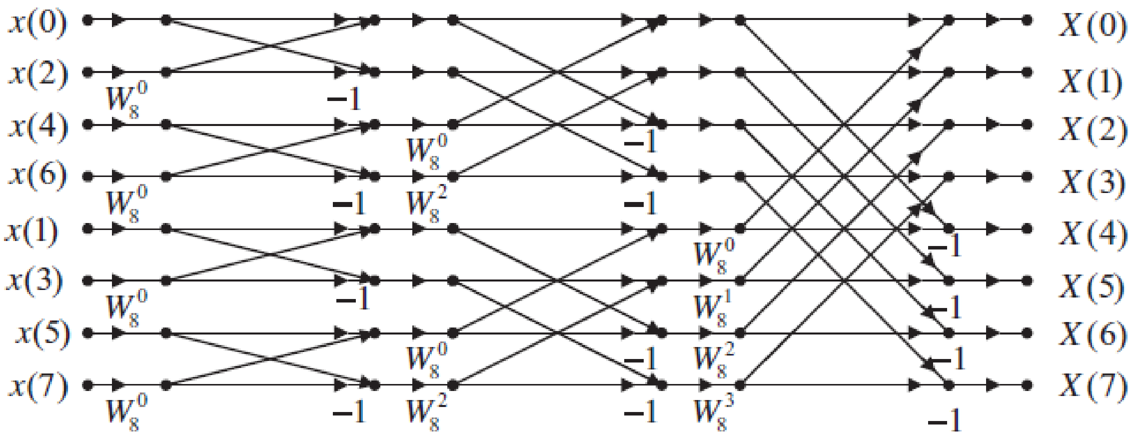
Computation of Inverse DFT using FFT

The following procedure can be followed to compute inverse DFT using FFT algorithm.

1. Take N-point frequency domain sequence $X(k)$ as input sequence.
2. Compute FFT by using conjugate of phase factors.
3. Divide the output sequence obtained in FFT computation by N, to get the sequence $x(n)$.

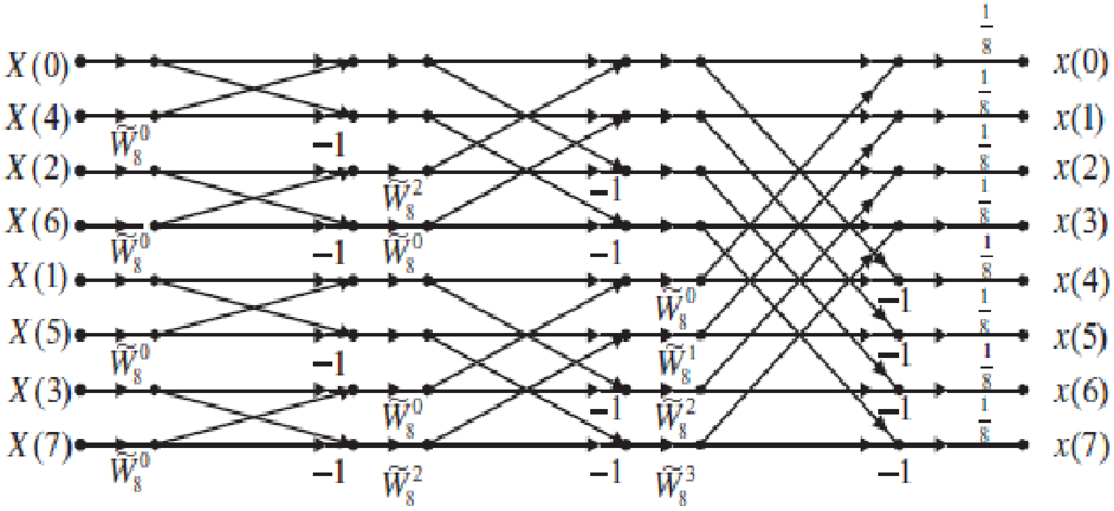
Thus a single FFT algorithm can be used to evaluation of both DFT and IDFT.

DFT using DIF-FFT

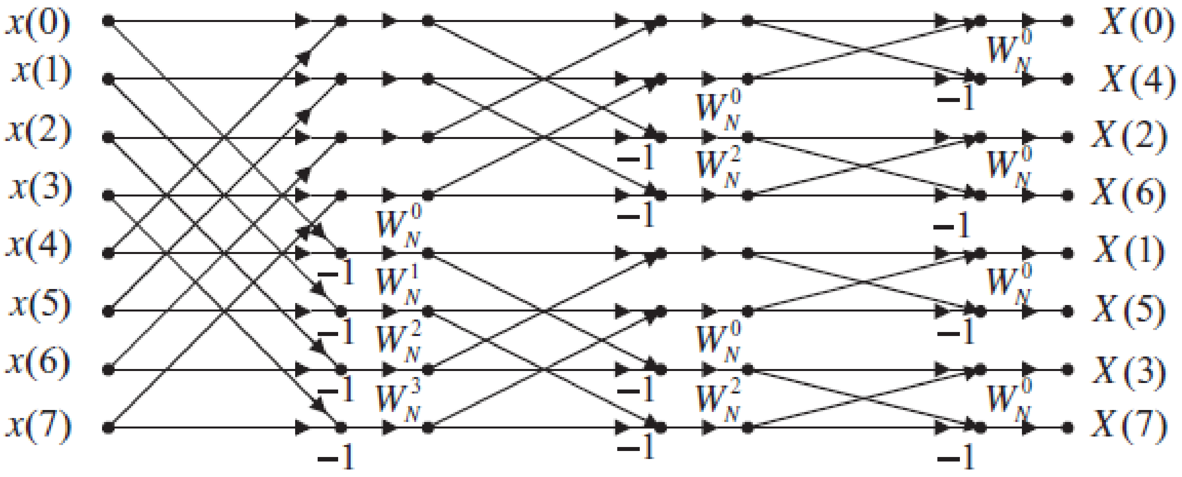


Where $\widetilde{W_N^r}$ is the complex conjugate of W_N^r.

IDFT using DIT-FFT

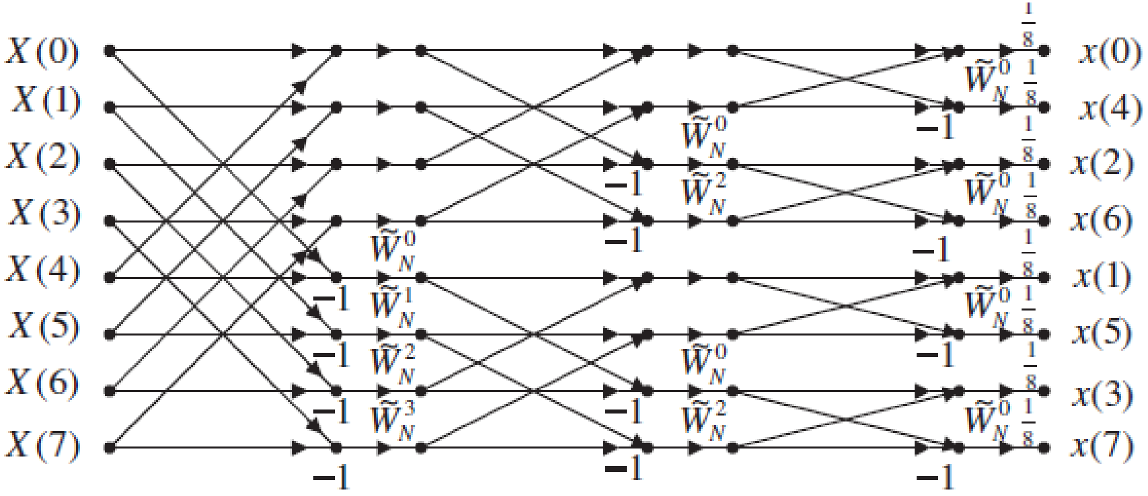


DFT using DIF-FFT



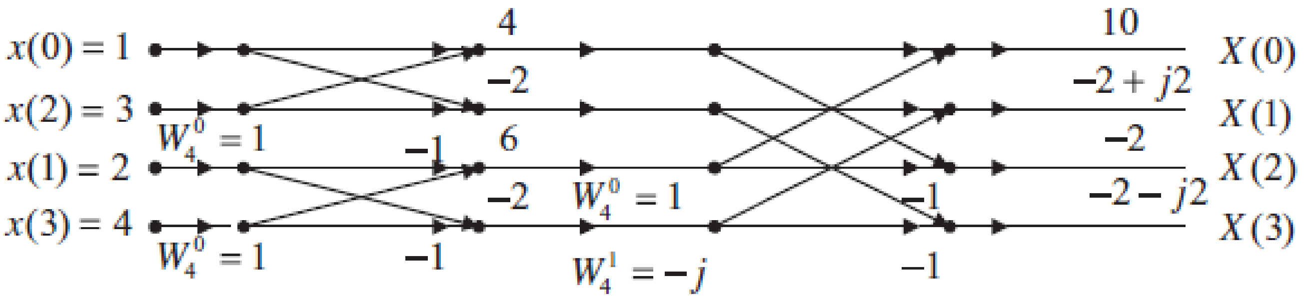
Where $\widetilde{W}_N^r$ is the complex conjugate of W_N^r.

IDFT using DIF-FFT

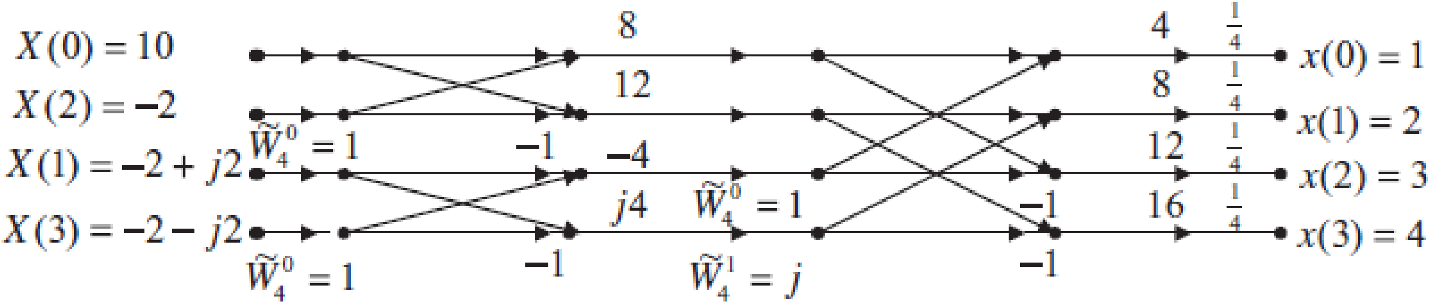


Example 1: Given $x(n) = (1, 2, 3, 4)$, evaluate its DFT using the DIT-FFT method. And also evaluate its IDFT using DIT-FFT method.

Solution: The DFT using DIT is as in figure.

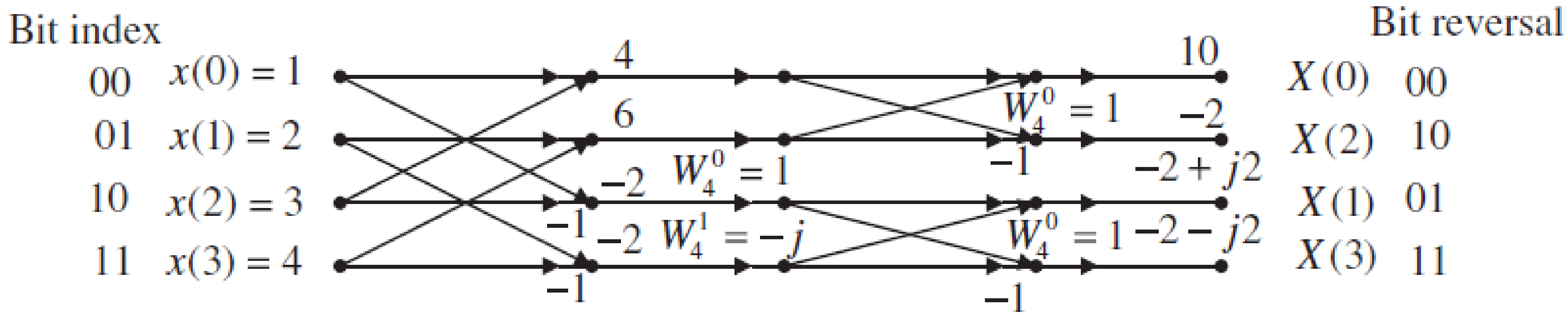


Thus $X(k) = (10, -2 + j2, -2, -2 - j2)$. Now IDFT using DIT is as in figure.

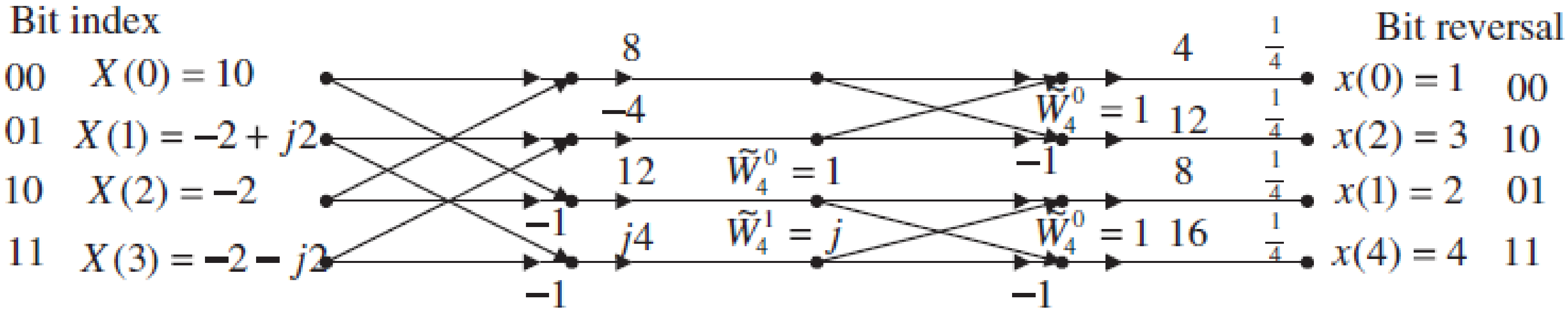


Example 2 : Same as example 1 using DIF-FFT.

Solution: The DFT using DIF is as in figure.



And the IDFT using DIF is as in figure.



Questions:

1. Why do we need DFT when we have DTFT? Determine the circular convolution of the sequence $x_1(n) = (1,2,3,4)\ and\ x_2(n) = (5,6,7,8)$ Also verify your answer.
2. Why DFT is needed and how the DFT solve the problem associated with DTFT.
3. Show that the multiplication of two DFT sequences results in circular convolution.
4. What is zero padding? Find the circular convolution of $x_1(n) = (1,2)\ and\ x_2(n) = (3,2,1)$
5. Using DIT-FFT compute DFT of $x(n) = \sin\left(\frac{3n\pi}{4}\right),\ 0 \le n \le 7$.
6. Use DIF-FFT algorithm to compute 8-point DFT of $x(n) = (2,1,1,1)$. Discuss the result.
7. Find 8-point DFT of the sequence {1,2,3,4,5,4,3,2,1} using radix-2 decimation in time algorithm.

8. What is the difference between linear and circular convolution? Find the linear and circular convolution of the following sequences. x(n)=(1,0,0,1) and h(n)=(2,0,2)
9. Why is DFT preferred over DTFT in the analysis of discrete time signals? Also determine the DFT of the signal x(n)=u(n)-u(n-4) using DIF-FFT algorithm.
10. Write short notes on computational complexity of DFT.
11. With the help of N=8, explain radix 2 decimation in time (DIT) FFT algorithm for computation of DFT, give the computational efficiency of FFT over DFT.
12. Using circular convolution method, determine the linear convolution of the following sequences: x(n)= (1,2,4) and h(n)=(1,2,1,3).
13. Write short notes on frequency shift property of DFT.
14. Why we need DFT when we have DTFT?