Module II

Fast Fourier Transform Algorithms

(proposed by Cooley and Turkey in 1965)

Fast Fourier transform is an algorithm for calculating DFT.

FFT reduces the computation time and improves performance by a factor of 100 or more over direct computation of DFT.

— The FFT exploits two basic properties of twiddle factor W_N.

① symmetry property: $W_N^{k+\frac{N}{2}} = -W_N^k$.

② periodicity property: $W_N^{k+N} = W_N^k$.

— Consider N point DFT

let $N = r^v$.

Then number 'r' is the radix of the FFT Algorithm.

— Then N-point DFT can be obtained by v-stage radix r FFT.

Radix-2 FFT Algorithm

- most widely used FFT Algorithm.
- $N = 2^{v}$ (N must be a power of 2).
  (eg: N=2, 4, 8, 16 .....)

Then to evaluate N point DFT using FFT v stages of Radix 2 FFT Algorithm.

- There are basically two classes of FFT Algorithms:

① Decimation in Time FFT (~~DIT F~~ (DIT-FFT Algorithm).

② Decimation in Frequency FFT (DIF-FFT Algorithm).

Radix-2 DIT-FFT Algorithm:

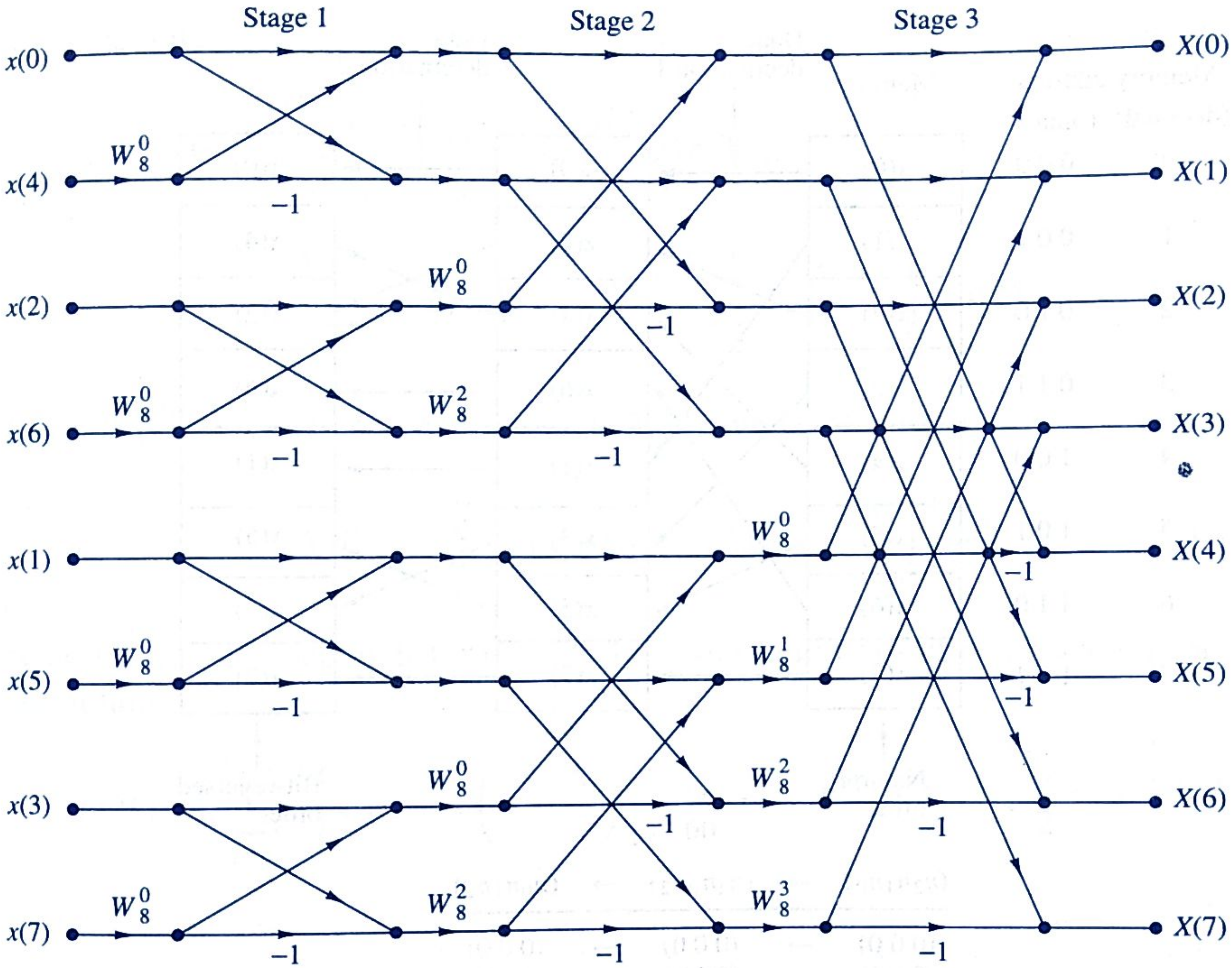
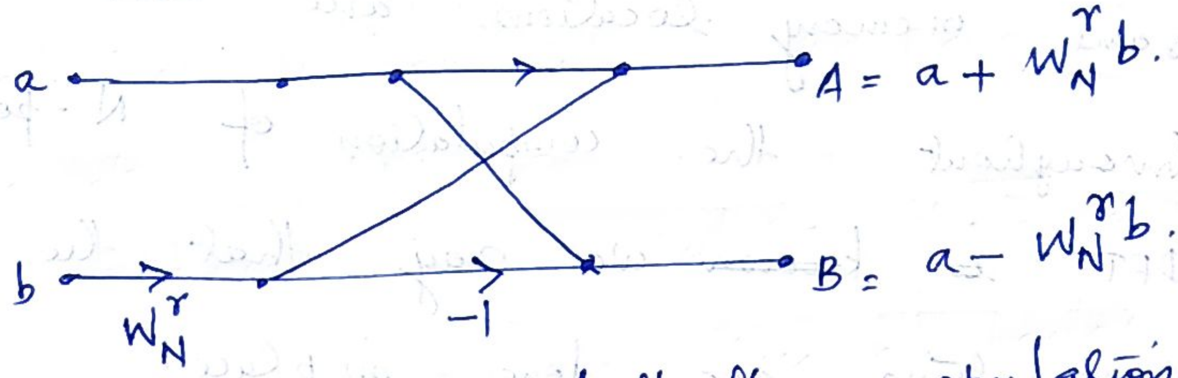


Figure 8.1.6 Eight-point decimation-in-time FFT algorithm.

Here we can see that basic computation at every stage is similar.



(Basic butterfly computation in DIT-FFT Algorithm).

— Here it take two complex numbers a and b as input and produces two complex numbers as output.

$A = a + W_N^r b$ and

$B = a - W_N^r b$.

— This basic computation is called butterfly because the flow graph resembles a butterfly.

— Once a butterfly operation is performed on a pair of complex numbers (a, b) to produce (A, B), there is no need to save input pair (a, b)

So in the same memory location (a, b) is replaced by (A, B). Since the same memory locations are used throughout the computation of N-point DFT ~~is known~~ we say that the computations are done in place.

(In place Computation).

— The second observation is regarding the order of input data.

~~In radix-2 DIT FFT Algorithm~~

Bit reversal

In DIT-FFT Algorithm the output sequence X(k) is in natural order i.e. k = 0, 1, 2 .. N-1 and the input sequence ~~is~~ has to be ~~in~~ stored in shuffled order.

— For radix-2 DIT-FFT algorithm (N is power of 2) the input sequence must be stored in bit reversed order for output to be computed in natural order.

For $N=8$ – Bit reversal. ($N=2^r$; $8=2^3$)

i/p $x(n)$ bit reversed order	bit reversed	r bit binary of (3 bit binary)	o/p $X(K)$ (natural order)
$x(0)$	000	000	$X(0)$
$x(4)$	100	001	$X(1)$
$x(2)$	010	010	$X(2)$
$x(6)$	110	011	$X(3)$
$x(1)$	001	100	$X(4)$
$x(5)$	101	101	$X(5)$
$x(3)$	011	110	$X(6)$
$x(7)$	111	111	$X(7)$

For $N=4$ Bit reversal

$$N=2^r \Rightarrow 4=2^2 \Rightarrow r=2.$$

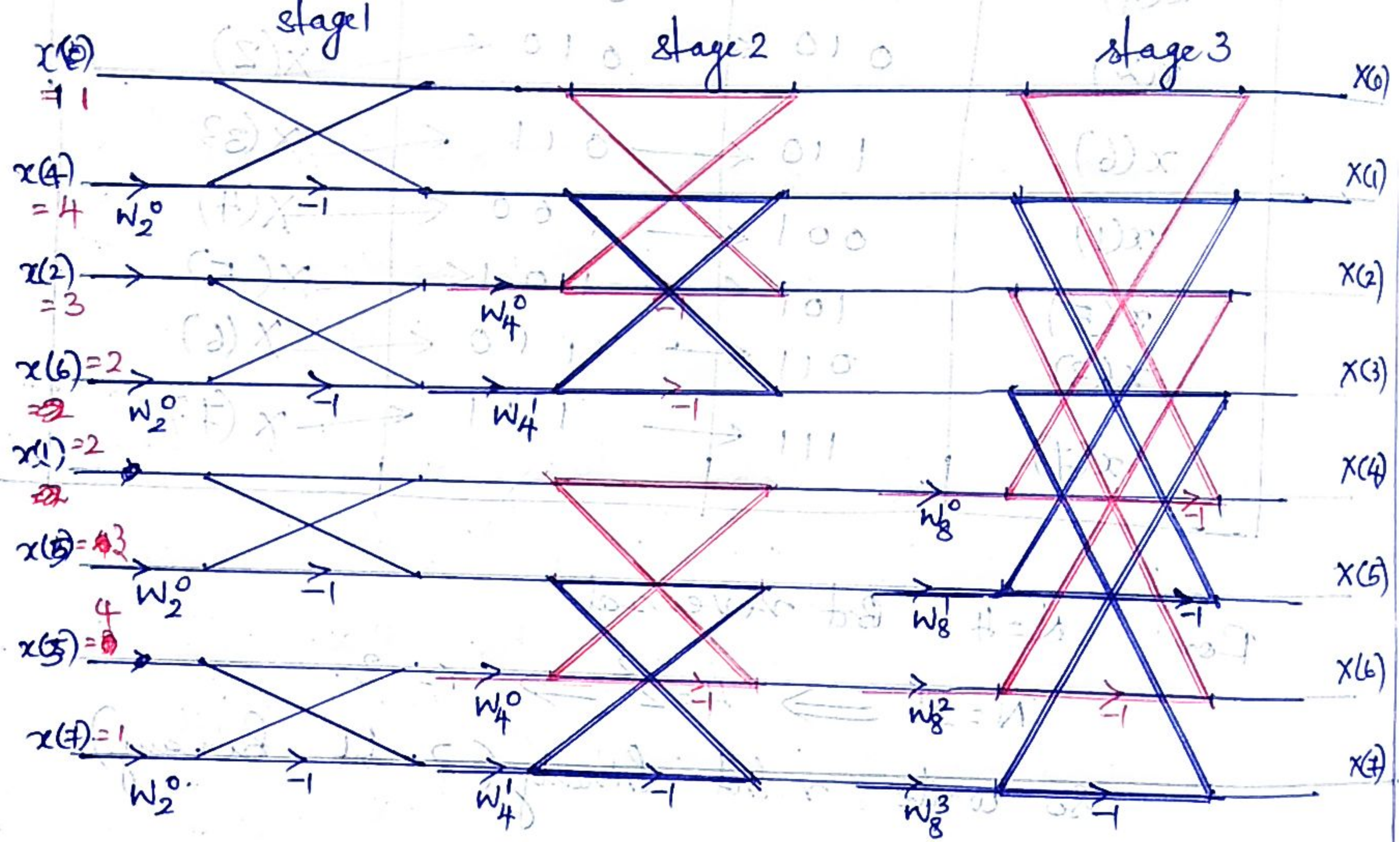
So write r bit binary (2-bit binary).

i/p $x(n)$ order	r bit binary bit reversed	r bit binary (2 bit binary)	o/p $X(K)$ natural order
$x(0)$	00	00	$X(0)$
$x(2)$	10	01	$X(1)$
$x(1)$	01	10	$X(2)$
$x(3)$	11	11	$X(3)$

Q) Find DFT of a sequence (Ramesh Babu 4.17).

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

Answer



$$\left.\begin{array}{ll} x(0) = 1 & x(4) = 4 \\ x(1) = 2 & x(5) = 3 \\ x(2) = 3 & x(6) = 2 \\ x(3) = 4 & x(7) = 1 \end{array}\right\} \text{inputs}$$

Twiddle factors associated with flowgraph are.

$$W_2^0 = W_4^0 = W_8^0 = 1$$

$$W_8^1 = \left(e^{-j2\pi/8}\right)^1 = e^{-j\pi/4} = 0.707 - j0.707.$$

$$W_4^1 = W_8^2 = \left(e^{-j2\pi/8}\right)^2 = e^{-j\pi/2} = -j$$

$$W_8^3 = \left(e^{-j2\pi/8}\right)^3 = e^{-j3\pi/4} = -0.707 - j0.707.$$

stage 1

$x(0) = 1$

$x(4) = 4$

$x(2) = 3$

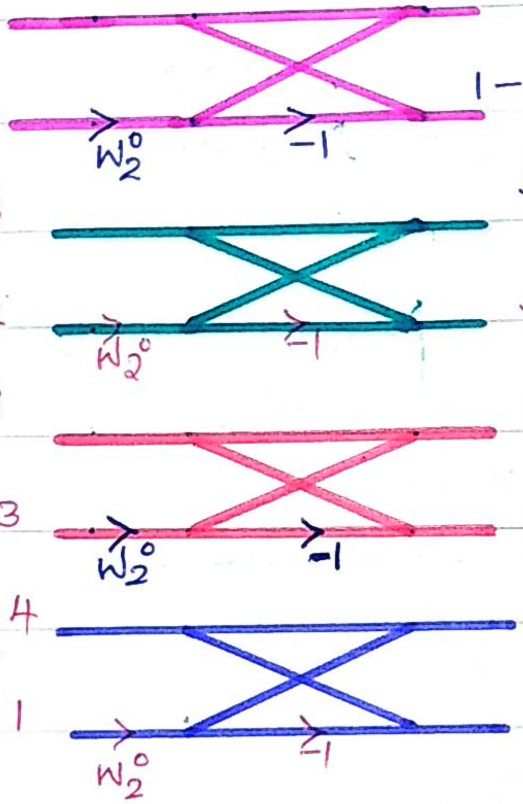
$x(6) = 2$

$x(1) = 2$

$x(5) = 3$

$x(3) = 4$

$x(7) = 1$



$1 + 4(W_2^0) = 1 + 4 = 5$

$1 - 4W_2^0 = 1 - 4 = -3$

$3 + 2 = 5$

$3 - 2 = 1$

$2 + 3 = 5$

$2 - 3 = -1$

$4 + 1 = 5$

$4 - 1 = 3.$

stage 2

5

-3

5

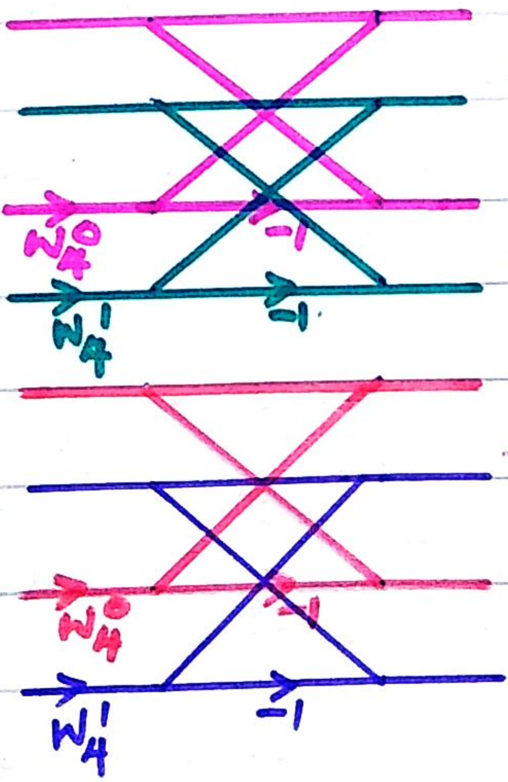
1

5

-1

5

3.



$5 + 5(W_4^0) = 5 + 5 = 10.$

$-3 + 1(W_4^1) = -3 + -j = -3 - j.$

$5 - 5(W_4^0) = 5 - 5 = 0.$

$-3 - 1(W_4^1) = -3 - (-j) = -3 + j.$

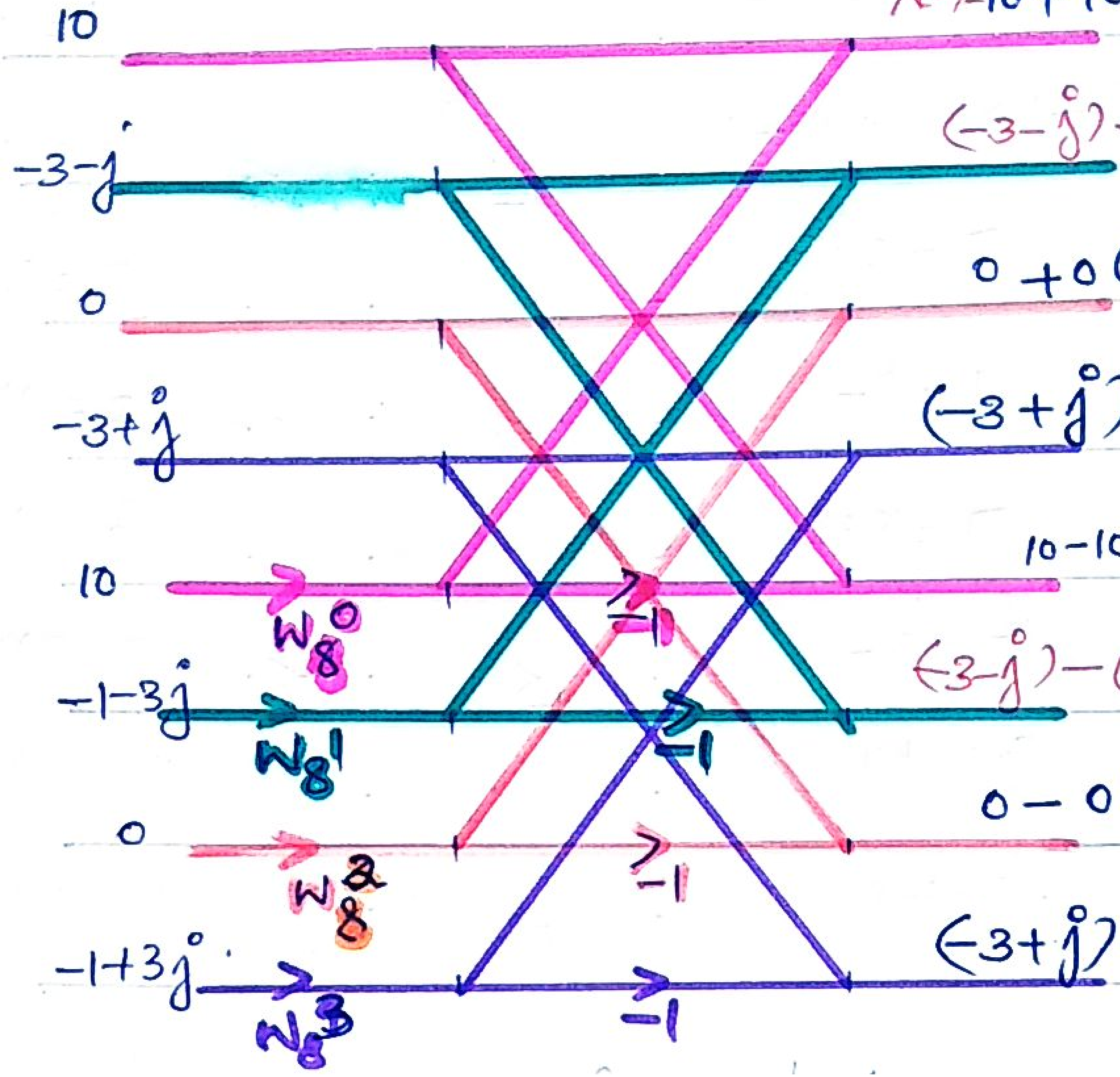
$5 + 5(W_4^0) = 5 + 5 = 10$

$-1 + 3(W_4^1) = -1 + 3(-j) = -1 - 3j$

$5 - 5(W_4^0) = 5 - 5 = 0.$

$-1 - 3W_4^1 = -1 - 3(-j) = -1 + 3j$

stage 3



$X(0) = 10 + 10(W_8^0) = 10 + 10 = 20 = X(0)$

$(-3-j) + (-1-3j)\,W_8^1 = 8 = X(1)$

$0 + 0(W_8^2) = 0 = X(2)$

$(-3+j) + (-1+3j)\,W_8^3 = X(3)$

$10 - 10(W_8^0) = 10 - 10 = 0 = X(4)$

$(-3-j) - (-1-3j)(W_8^1) = ?$
$= X(5)$

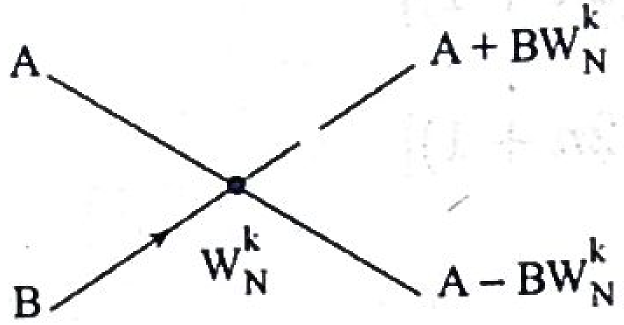
$0 - 0(W_8^2) = 0 = X(6)$

$(-3+j) - (-1+3j)\,W_8^3 = X(7)$

Input	Output of stage 1 (S_1)	Output of stage 2 (S_2)	Output
1	$1+4=5$	$5+5=10$	$10+10=20$
4	$1-4=-3$	$-3+(-j)1=-3-j$	$-3-j+(0.707-j0.707)(-1-3j)$ $=-5.828-j2.414$
3	$3+2=5$	$5-5=0$	0
2	$3-2=1$	$-3-(-j)1=-3+j$	$(-3+j)+(-0.707-j0.707)(-1+3j)$ $=-0.172-j0.414$
2	$2+3=5$	$5+5=10$	$10-10=0$
3	$2-3=-1$	$-1+(-j)3=-1-3j$	$-3-j-(0.707-j0.707)(-1-3j)$ $=-0.172+j0.414$
4	$4+1=5$	$5-5=0$	0
1	$4-1=3$	$-1-(-j)3=-1+3j$	$(-3+j)-(-0.707-j0.707)(-1+3j)$ $=-5.828+j2.414$

$$X(k)=\{20,-5.828-j2.414,0,-0.172-j0.414,0,-0.172+j0.414,0,\\ -5.828+j2.414\}$$

The basic operation is



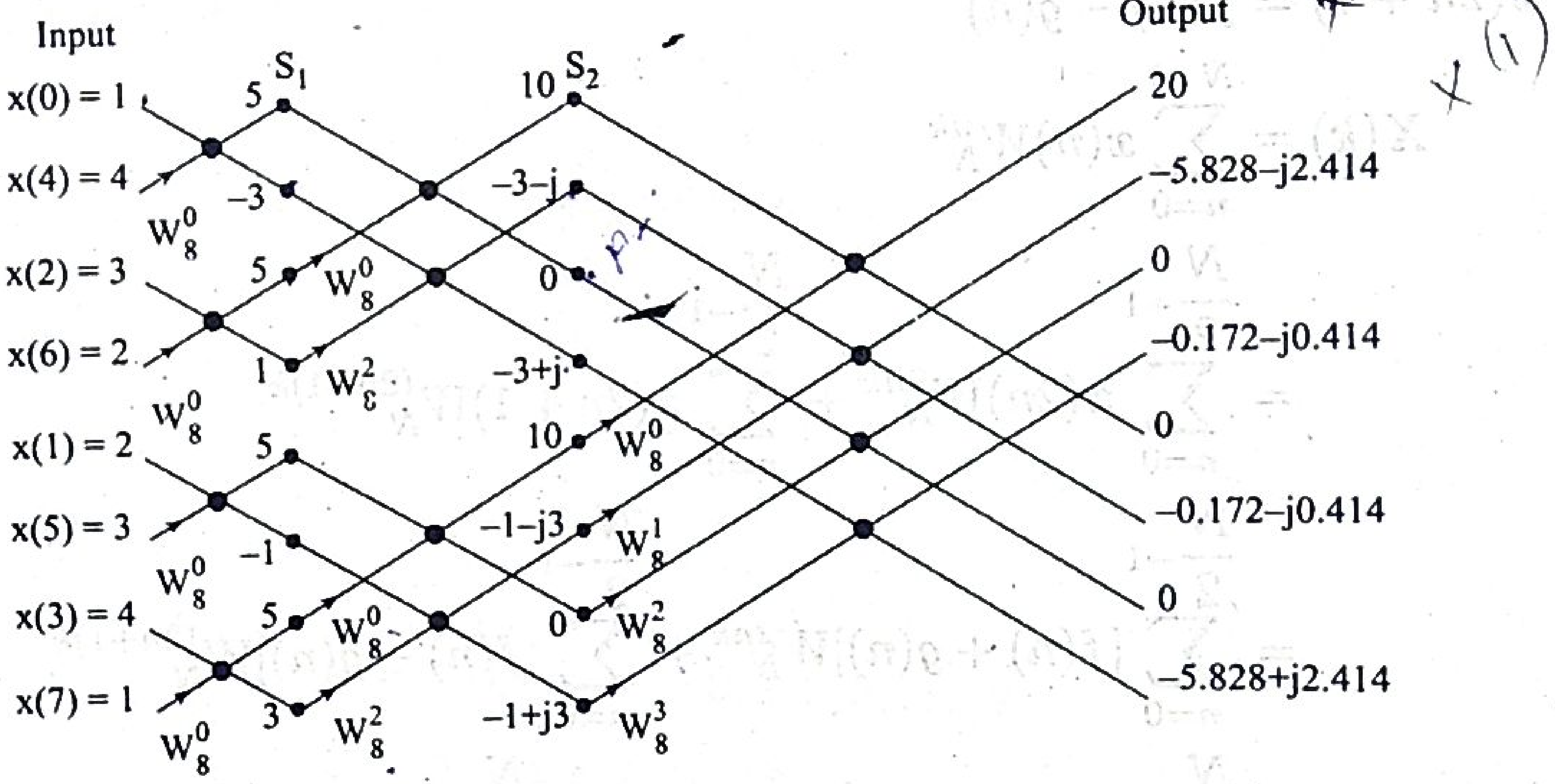


table from text

Q) Find the 4-point DFT of the sequence.

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

Answer:

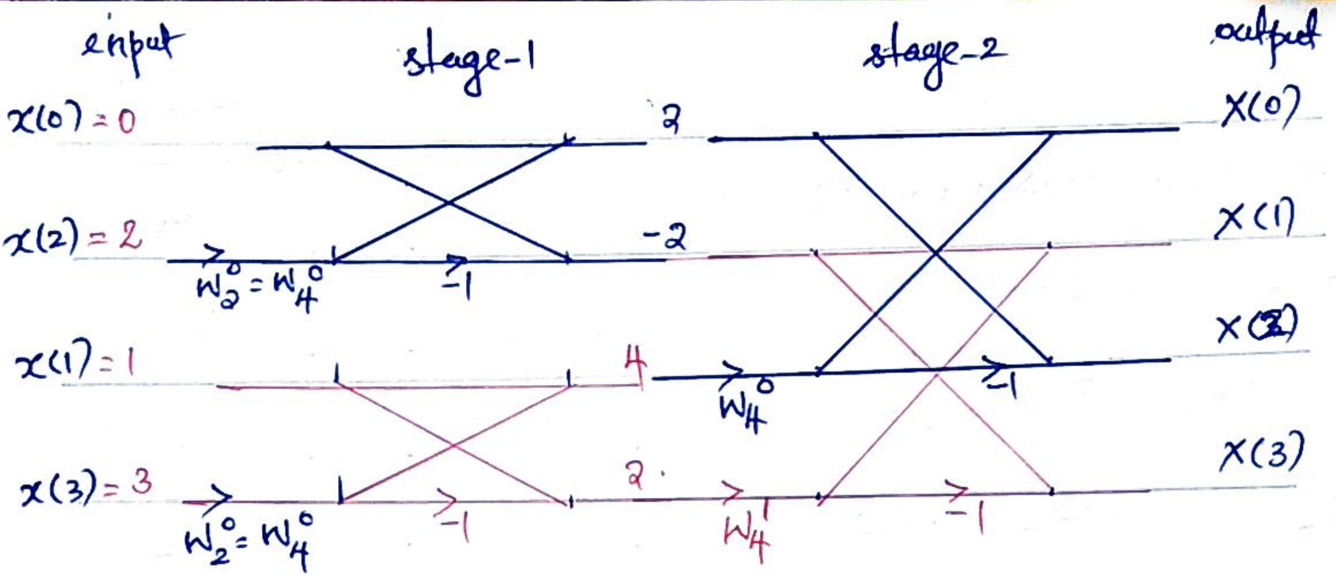
Bit reversal.

i/p			o/p
$x(0)$	00	$\leftarrow$ 00	$X(0)$
$x(2)$	10	01	$X(1)$
$x(1)$	01	10	$X(2)$
$x(3)$.	11	11	$X(3)$

Associated twiddle factor

$W_4^0 = 1$

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



input	output of stage-1 S_1	output of stage 2 (Output)
0	$0+2 = 2$	$2+4 = 6$
2	$0-2 = -2$	$-2+-2(W_4^1) = -2+-2(-j) = -2+2j$
1	$1+3 = 4$	$2-4 = -2.$
3	$1-3 = -2.$	$-2-(-2)\,W_4^1 = -2+2(-j) = -2-2j.$

$$\therefore X(k) = \{6,\ -2+2j,\ -2,\ -2-2j\}$$