Digital System and binary Number

Digital systems are used in communication, business, transactions, traffic controls, medical treatment, weather monitoring, internet and many others commercial and industrial.

- Binary digit called a bit (0, 1)
- Group of bits called binary codes

* Number System:

A physical system whose behaviour is described by mathematical equations is simulated in a digital computer by means of number system.

In any number system, there is an ordered set of symbols known as digits, there are two types of number system:

(i) Weighted number system (Positional Number System)

(ii) Non-weighted number system (non-positional number system)

* Positional number system:

The position of each digit of a number has some positional weight. e.g. Binary, octal, decimal, hexadecimal, BCD, etc.

* Non-Positional number system:

In a non-positional number system a digit does not indicate any significance in position and weight.

e.g. Gray Code, Excess-3 Code.

⇒ Radix or base of a number system must be an integer and greater than 1.

* The following number systems are ~~unit~~ used in digital system.

1. Decimal number System
2. Binary number System
3. Octal number System
4. Hexadecimal number System

⇒ Decimal no. System :

The decimal no. system has 10 symbols, so that the radix or base of this no. system is 10. The 10 symbols are –

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

$$---\ \underbrace{10^2\ 10^1\ 10^0}_{\text{integer part}}\ \underset{\text{decimal point}}{\cdot}\ \underbrace{10^{-1}\ 10^{-2}}_{\text{Fractional part}}\ ----$$

e.g:

$(434.75)_{10}$

$$\Rightarrow 4\times10^2 + 3\times10^1 + 4\times10^0 + 7\times10^{-1} + 5\times10^{-2}$$

⇒ Binary no. System :

The base of binary number system is two, only two digits are used 0 and 1.

$$----\ 2^2\ 2^1\ 2^0\ \underset{\text{Binary point}}{\cdot}\ 2^{-1}\ 2^{-2}\ ----$$

⇒ **Octal number System:**

The base of the number system is eight. The digits are 0, 1, 2, 3, 4, 5, 6, 7

$$8^2 \; 8^1 \; 8^0 \,.\, 8^{-1} \; 8^{-2}$$

↓ Octal point.

⇒ **Hexadecimal Number System:**

The base of the hexadecimal no. system is 16 and the digits are – 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, $\underbrace{A}_{10}$, $\underbrace{B}_{11}$, $\underbrace{C}_{12}$, $\underbrace{D}_{13}$, $\underbrace{E}_{14}$, $\underbrace{F}_{15}$

① A number system with base or radix (r), contains r different digit and they are from (0 to r−1).

* $\boxed{\text{Conversion}}$:

(i) **Decimal to other:**

To convert decimal into any other base (r), divide integer part and multiply fractional part with r.

Q

$(26.125)_{10} = (?)_2 = (11010.001)_2 = (122.02)_4 = (42.043)_6$

2	26	0
2	13	1
2	6	0
2	3	1
	1	

$0.125 \times 2 = 0.250$

$0.250 \times 2 = 0.500$

$0.500 \times 2 = 1.00$

$= (11010.001)_2$

4	26	2
4	6	2
	1	

$0.125 \times 4 = 0.500$

$0.500 \times 4 = 2.000$

$= (122.02)_4$

$$\begin{array}{c|c|c} 6 & 26 & 2 \\ \hline & 4 & \end{array}$$

$0.125 \times 6 = 0.750$

$0.750 \times 6 = 4.500$

$0.500 \times 6 = 3.000$

$= (42.043)_6$

$$\begin{array}{c|c|c} 8 & 26 & 2 \\ \hline & 3 & \end{array}$$

$0.125 \times 8 = 1.000$

$= (32.1)_8$

$$\begin{array}{c|c|c} 16 & 26 & A\ (=10) \\ \hline & 1 & \end{array}$$

$0.125 \times 16 = 2.000$

$= (1A.2)_{16}$

(Q) $(27)_{10} = (?)_8 = (?)_{16} = (?)_4$

$$\begin{array}{c|c|c} 8 & 27 & 3 \\ \hline & 3 & \end{array}$$

$= (33)_8$

$$\begin{array}{c|c|c} 16 & 27 & 11 = B \\ \hline & 1 & \end{array}$$

$= (1B)_{16}$

$$\begin{array}{c|c|c} 4 & 27 & 3 \\ \hline 4 & 6 & 2 \\ \hline & 1 & \end{array}$$

$= (123)_4$

(ii) Other ~~Octal~~ to Decimal :

To convert any other base r to decimal, multiply each digit with positional weight then add.

example:

(Q-(i)) $(1101.011)_2 = (?)_{10}$

$\Rightarrow 1\times2^3 + 1\times2^2 + 0\times2^1 + 1\times2^0 + 0\times2^{-1} + 1\times2^{-2} + 1\times2^{-3}$

$\Rightarrow 8 + 4 + 0 + 1 + 0 + 0.25 + 0.125$

$\Rightarrow (13.375)_{10}$

(Q-ii) $(426.35)_8 = (?)_{10}$

$= 6\times8^0 + 2\times8^1 + 4\times8^2 + 3\times8^{-1} + 5\times8^{-2}$

$= 6 + 16 + 256 + 0.375 + 0.078$

$= (278.453)_{10}$

(iii) $(231.22)_4 = (?)_{10}$

$= 1\times4^0 + 3\times4^1 + 2\times4^2 + 2\times4^{-1} + 2\times4^{-2}$

$= 1 + 12 + 32 + 0.5 + 0.125$

$= (45.625)_{10}$

(iv) $(ABD9.C2)_{16} = (?)_{10}$

$= A\times16^3 + B\times16^2 + D\times16^1 + 9\times16^0 + C\times16^{-1} + 2\times16^{-2}$

$= 10\times16^3 + 11\times16^2 + 13\times16^1 + 9\times16^0 + 12\times16^{-1} + 2\times16^{-2}$

$= (43993.757)_{10}$

(v) $(542.33)_6 = (?)_{10}$

$$= 5 \times 6^2 + 4 \times 6^1 + 2 \times 6^0 + 3 \times 6^{-1} + 3 \times 6^{-2}$$

$$= (5 \times 36) + (4 \times 6) + 2 + \frac{3}{6} + \frac{3}{36}$$

$$= (206.583)_{10}$$

(iii) Binary to Octal :

$(1110110.110100)_2 = (?)_8 = (166.64)_8$

$$\underbrace{001}_{1}\,\underbrace{110}_{6}\,\underbrace{110}_{6}\,.\,\underbrace{110}_{6}\,\underbrace{100}_{4}$$

$8 = 2^{3}$ bit

$(11101101.11011)_2 = (?)_8 = (355.66)_8$

$$\underbrace{011}_{3}\,\underbrace{101}_{5}\,\underbrace{101}_{5}\,.\,\underbrace{110}_{6}\,\underbrace{110}_{6}$$

(iv) Octal to Binary !

$(2657.43)_8 = (?)_2$

$(010\,110\,101\,111\,.\,100\,011)_2$

$$\begin{bmatrix} 4 & 2 & 1 \\ 2^2 & 2^1 & 2^0 \end{bmatrix}$$

(V) Hexadecimal to Binary !

$(ABD8.9C)_{16} = (?)_2$

$$(\underbrace{1010}_{A}\,\underbrace{1011}_{B}\,\underbrace{1101}_{D}\,\underbrace{1000}_{8}\,.\,\underbrace{1001}_{9}\,\underbrace{1100}_{C})_2$$

$$\begin{bmatrix} 8 & 4 & 2 & 1 \\ 2^3 & 2^2 & 2^1 & 2^0 \end{bmatrix}$$

$16 = 2^{4}$ bit

(vi) Binary to Hexadecimal :

$$(\underbrace{1\,1011}_{11\,(B)}\,\underbrace{0011}_{3}\,\underbrace{1101}_{D(13)}\,.\,\underbrace{1101}_{D}\,\underbrace{0001}_{1})_2$$

$$= (1B3D.D1)_{16}$$

(vii) Octal to Hexadecimal :

$$(765.464)_8 = (?)_{16}$$

$$\underbrace{0001}_{1}\,\underbrace{1111}_{15}\,\underbrace{0101}_{5}\,.\,\underbrace{1001}_{9}\,\underbrace{1010}_{10}\,0$$

$$(1F5.9A0)_{16}$$

A	B	C	D	E	F
10	11	12	13	14	15

(viii) Hexadecimal to Octal :

$$(ABDE8.ACF)_{16}$$

$$\underbrace{010}_{2}\,\underbrace{101}_{5}\,\underbrace{011}_{3}\,\underbrace{110}_{6}\,\underbrace{111}_{7}\,\underbrace{101}_{5}\,\underbrace{000}_{0}\,.\,\underbrace{101}_{5}\,\underbrace{011}_{3}\,\underbrace{001}_{1}\,\underbrace{111}_{7}$$

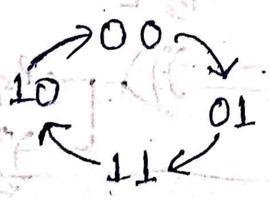
$$8 = 2^{3\ \text{bit}}$$

$$(2536750.5317)_8$$

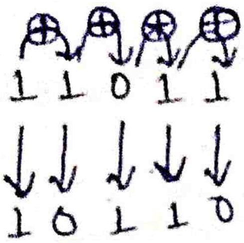
* Gray Code :

- It is unweighted code.
- Successive no. is differ by one bit.
- It is also called unit distance code.
- It is also called cyclic code and minimum error code.

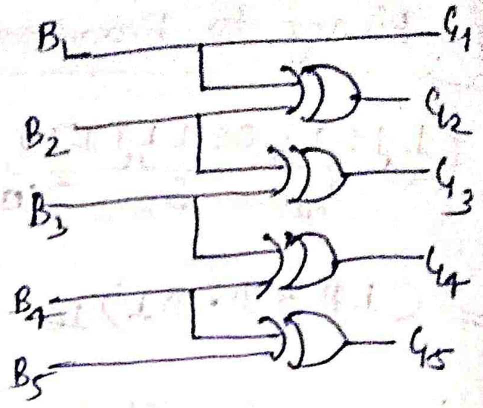
e.g.

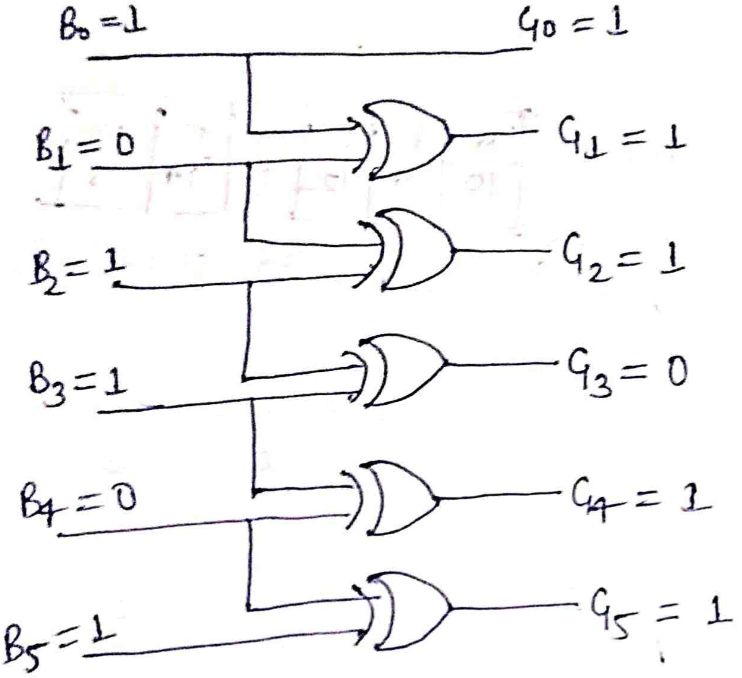


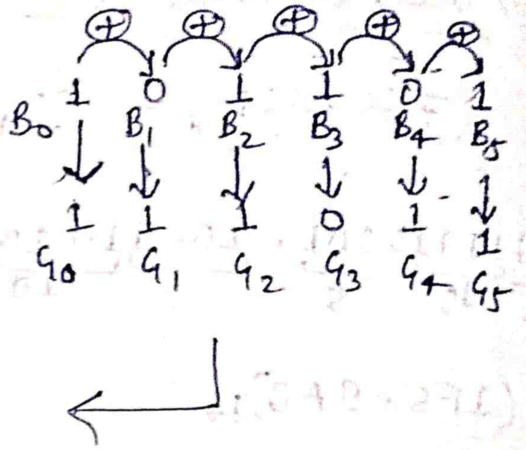
* Binary code to gray code conversion:



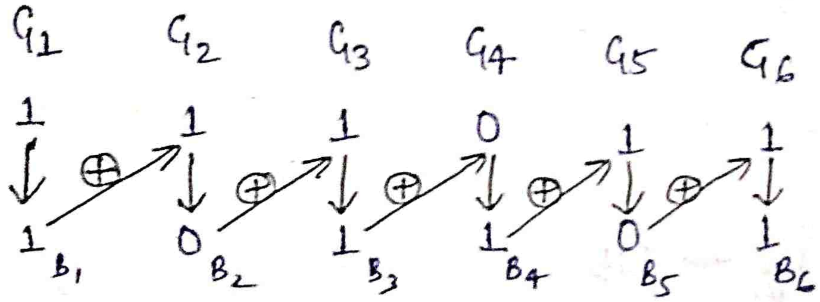
0	0	→ 0
1	1	→ 0
1	0	→ 1
0	1	→ 1

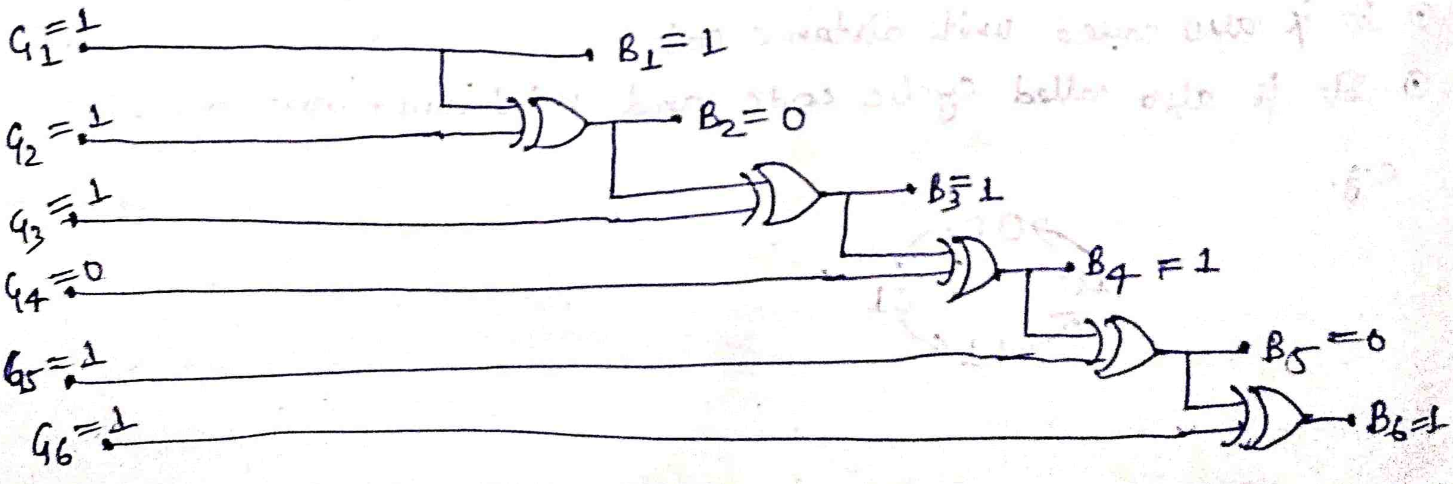






* Gray to binary conversion;





* Binary Codes:

(i) BCD (Binary Coded decimal):

- It is called binary coded decimal
- It has four bit code.
- It is also called 8421 code.
- Each ~~demic~~ decimal digit is represented with 4 bit.
- It is weighted code

Decimal	8421 (BCD)
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Example :

Binary of 10   1010
                      + 0110
In BCD   0001 0000

* BCD Addition :

Consider the addition of two decimal digits in BCD together with a carry from a previous less significant pair of digits.

Since each digit doesnot exceed 9, the sum cannot be greater than 9+9+1 equal to 19 with 1 being a previous carry.

- When binary sum less than equal to 9(1001), the corresponding BCD digit is correct.
- When binary sum is greater than equal to 10 (1010), the result is invalid BCD digit.
- The addition of six (0110) to the binary sum converts it to the correct digit and also produces a carry.

25	0010	0101
+ 41	0100	0001
66	0110	0110

(0110 0110)$_{BCD}$

89	1000	1001
+ 49	+ 0100	+ 1001
138	1101	10010
	+ 0110	+ 0110
	10011	11000

$= (0001\ 0011\ 1000)_{BCD}$

(Q) Express decimal numbers in BCD and perform the addition in BCD.

$(789)_{10}$ and $(497)_{10}$

$$
\begin{array}{rcllll}
(789)_{10} & \rightarrow & (0111 & 1000 & 1001)_{BCD} & \\
(497)_{10} & \rightarrow & (0100 & 1001 & 0111)_{BCD} & \\
\hline
(1286)_{10} & & 1100 & 10010 & 10000 & \text{Invalid BCD, add 6} \\
 & & 0110 & 0110 & 0110 & \\
\hline
 & & 10010 & 11000 & 10110 &
\end{array}
$$

$= (0001\ 0010\ 1000\ 0110)_{BCD}$

* Excess-3 Code:

- Excess-3 code = BCD + 3
- It is non-weighted code.
- It is also called self complement code.
- It is 4 bit code.
- Unused bit combination 0000, 0001, 0010, 1101, 1110, 1111
- Only non-weighted code which is self compliment is Excess-3 Code.

Excess-3 Code = BCD + 3.

Decimal	Excess-3 Code
0	0011
1	0100
2	0101
3	0110
4	0111
5	1000
6	1001
7	1010
8	1011
9	1100

Self Complement code

```
  9 5 4        1001    0101    0100
+ 2 9 9        0010    1001    1001
-------       -----   -----   -----
1 2 5 3        1100    1111    1101    Invalid BCD, add 6
               0110    0110    0110
              -----   -----   -----
              10010   10101   10011
```

= (0001 0010 0101 0011) BCD

```
  8 9 6        1000    1001    0110
  4 8 9        0100    1000    1001
-------       -----   -----   -----
 13 8 5        1101   10010    1111
               0110    0110    0110
              -----   -----   -----
              10011   11000   10101
```

= (0001 0011 1000 0101) BCD

e.g.

$(24)_{10}$

	0010	0100	
	0011	0011	(add 3 in BCD)
	0101	0111	

Note: the code whose addition is 9 is self complement code.

e.g. 2421, 3321, 5211, 4311.

(Self complement and weighted code).

Decimal	2421 code
0	0000
1	0001
2	0010
3	0011
4	0100
5	1011
6	1100
7	1101
8	1110
9	1111

e.g:

4 → 0100   4 → 1010

5 – 1011   5 → 0101

Binary Addition:

```
      1 1 1 1 1
      0 0 1 1 0
      1 1 1 1 1
      1 1 0 1 1
   ------------
   10 1 1 1 1 1
```

```
     (4)(3)(3)
      1  1  1  0  1
      1  1  1  1  1
      1  1  1  1  1
      1  1  0  1  1
      1  1  1  1  1
   ----------------
   1001  0  1  0  1
```

* Complement :

Complement are used in digital computer to simplify the subtraction operation and for logical manipulation.

Complement represents the negative number.

There are two types of complements for each base r :-

(i) Radix Complement (r's)

(ii) Diminished radix complement {(r-1)'s}

⇒ Diminished radix complement (r-1)'s :

Given a no. N in base r having n digits, the {(r-1)'s} complement of N is defined as -

$$\boxed{(r^n - 1) - N}$$

9's complement :

The 9's complement of a decimal no. is obtained by subtracting each digit from 9.

example:

$$\begin{array}{r} 99999 \\ -\ 49065 \\ \hline 50934 \\ \hline \end{array}$$

1's complement:

The 1's complement of a binary no. is ~~defined~~ obtained by subtracting each digit from 1.

example:

1's complement of 110011 = 001100

⇒ The 1's complement of binary no. is obtained by changing (1's to 0's) and (0's to 1's).

Radix complement:

The $(r's)$ complement of an n digit no. N in base r is $(r^n - N)$ for $N \neq 0$.

The r's complement is obtained by adding 1 to the $(r-1)$'s complement.

10's complement:

Firstly to find the 9's complement than add 1 to the LSB (least significant bit or right most bit).

eg.:

10's complement of 49056 = 9's complement + 1

$$= \begin{array}{r} 50934 \\ +\ 1 \\ \hline 50935 \\ \hline \end{array}$$

2's complement;

Firstly to find the 1's complement than add 1 to the LSB.

e.g. 2's complement = 1's complement + 1

$$= \begin{array}{r} 001011 \\ +\quad 1 \\ \hline 001100 \end{array}$$

* Subtraction with complements;

The subtraction of two n digit unsigned no. (M−N) in base r can be done as −

(i) Add the minus M to the r's complement of the subtrahend N.

(ii) If $M \geq N$, the sum will produce on end carry r^n, which can be discarded.

(iii) If $M \leq N$, the sum does not produce an end carry then take the r's complement of the sum and place a −ve sign in front,

a) e.g.

(i) Using 10's complement, subtract 52532 − 3250 (M = 52532, N = 3250)

M = 52532

N = 03250

10's complement of 03250 = 9's complement of 03250 + 1

= (99999 − 03250) + 1

= 96749 + 1

= 96750

$M = 52532$

10's comp. of N $= 96750$

$+$

$\overline{149282}$

discard carry.

(ii) $59024 - 6534$

$M = 59024$

$N = 06534$

10's complement of 06534 = (9's complement of 06534) + 1

$= (99999 - 06534) + 1$

$= 93465 + 1$

$= 93466$

$M = 59024$

10's comp. of N $= 93466$

$+$

$\overline{152490}$

discard carry

(iii) $59467 - 46312$

$M = 59467$

$N = 46312$

10's complement of 46312 = 9's complement of N + 1

$= (99999 - 46312) + 1$

$= 53687 + 1$

$= 53688$

$$M = 59467$$

10's comp of $N = 53688$
$+$
113155

discard carry

(iv) $3250 - 52532$

$M = 03250$, $N = 52532$

10's complement of $N = (99999 - 52532) + 1$

$= 47467 + 1$

$= 47468$

$M = 03250$

10's comp. of $N = 47468$
$+$
50718

10's complement of $50718 = (49281 + 1)$

$= -49282$

(v) $6534 - 59024$

$M = 06534$, $N = 59024$

10's complement of $N = (99999 - 59024) + 1$

$= 40975 + 1$

$= 40976$

$$M = 06534$$

10's complement of $N = 40976$

$$\begin{array}{r} + \\ \hline 47510 \\ \hline \end{array}$$

10's complement of $47510 = 52489 + 1$

$= -52490$

(vi) $46312 - 59467$

$M = 46312$, $N = 59467$

$$\begin{array}{rl} M = & 46312 \\ \text{10's comp. of } N = & +40533 \\ \hline & 86845 \\ \hline \end{array}$$

10's complement of $86845 = 13154 + 1$

$= -13155$

(b)(i) Subtract 34 − 17 using 2's complement.

$M = 34 = 100010$

$N = 17 = 010001$

2's complement of 010001 = 1's complement + 1

$= 101110 + 1$

$= 101111$

$$\begin{array}{rl} M = & 100010 \\ \text{2's of } N = & 101111 \\ & + \\ \hline & 1(010001)_2 \end{array}$$

carry discarded

(ii) Subtract 17 − 34

M = 17 = 010001

N = 34 = 100010

2's comp. of N = (1's comp. + 1)

= (101110 + 1)

=

M = 010001

2's comp of N = (1's comp. of N + 1)

~~101110 + 1~~

= (011101 + 1)

= 011110

M = 010001

2's of N = 011110

101111

2's complement of (101111) = 010000 + 1

= − 010001.

* Subtraction with {(r−1)'s} complement ;

The result of adding the minuend to the complement of subtrahend produces a sum i.e. 1< the correct difference when an end carry occurs. Remove the end carry and adding one to the sum is refered to as end around carry.

① 65345 − 4250

Subtract this using 9's complement.

X = 65345 , Y = 04250

9's complement of (04250) = 95749

$$
\begin{array}{lr}
X = & 65345 \\
9\text{'s of } Y = & 95749 \\
+ & \\
\hline
 & 1\,61094 \\
\text{end around carry} & +1 \\
\hline
 & 61095 \\
\hline
\end{array}
$$

② 04250 − 65345

X = 04250 , Y = 65345

$$
\begin{array}{lr}
X = & 04250 \\
9\text{'s comp of } Y = & 34654 \\
+ & \\
\hline
 & 38904
\end{array}
$$

9's complement of 38904 = 61095

Ans = −(61095) = −(9's comp. of 38904)

③ Subtract 54 − 27 using 1's complement.

X = 54 , Y = 27

X = 110110 , Y = 011011

1's complement of 27 (011011) = 100100

$$
\begin{array}{rl}
X = & 110110 \\
 & 100100 \\
+ & \\
\hline
 & 1011010 \\
\text{end around carry} & +1 \\
\hline
 & 011011 \\
\hline
\end{array}
$$

(4) 17 − 129 using 1's complement.

X = 17 , Y = 129

X = 10001 , Y = 10000001

$$
\begin{array}{rl}
\cancel{X =} & \cancel{00010001} \\
\cancel{\text{1's of } Y =} & \cancel{10000001} \\
+ & \\
\hline
 & \cancel{10010010} \\
\hline
\end{array}
\qquad
\begin{array}{rl}
X = & 00010001 \\
\text{1's of } Y = & 01111110 \\
+ & \\
\hline
 & 10001111 \\
\end{array}
$$

1's of 10001111 = 01110000

Answer = −01110000

* Signed Numbers!

Data Representation

- Magnitude
  - Unsigned
  - Signed
- Complement
  - 1's (+ve, −ve)
  - 2's (+ve, −ve)

	Unsigned	Signed	1's (+ve, −ve)	2's (+ve, −ve)
+6	110	0110	0110	0110
−6	×	1110	1001	1010

Signed Binary Numbers :

- If the data ~~is~~ has +ve as well as −ve numbers, then the signed binary no. must be used.

- The sign with a bit placed in the left most position of the no.

  sign bit 0 ⟶ +ve

  sign bit 1 ⟶ −ve

- If the binary no. is signed, then the left most bit represents the sign and the rest of the bits represent the no.

- If the binary no. is unsigned then the left most bit is the most significant bit of the no.

e.g. 9 is unsigned no. and ~~the~~ represented in binary 1001.
but +9 is a signed no. and represented in binary 01001.

25 — 11001 (unsigned)

−9 — 11001 (signed)

1's complement of 27(011011) = 100100

X = 110110

1's of Y = 100100

+

1011010

1

+

011011

There are two types of signed no. representation:

① Signed Magnitude representation.

② Signed Complement representation.

* Signed Complement representation:

ⓐ Signed 1's complement

ⓑ Signed 2's complement.

(1) Signed Magnitude representation:

The no. consist of a magnitude and a symbol (+ or −) or a bit (0, 1) indicating the sign. This is the representation of signed numbers used in ordinary arithmetic.

When arithmetic operations are implemented in a computer, it is more convient to use a different system refered to as signed complement system.

(2) Signed complement representation:

Complement is used for representing -ve no. In this system a -ve no. is indicated by its complement.

(a) Signed 1's complement representation;

For 1's complement representation of signed no. we have to find 1's complement including sign bit.

(b) Signed 2's complement representation:

Find 2's complement, add 1 to the 1's complement of a binary no.

(Q) Represent each of the following signed decimal no. as a signed binary no. in 8 bit sign magnitude representation 1's complement and 2's complement representation.

(i) +13 = 00001101

(ii) -13 = 10001101

sign bit

+13

Sign magnitude represents = 00001101

" 1's complement = 00001101

" 2's complement = 00001101

-13

Signed magnitude = 10001101

" 1's complement = 11110010

" 2's " = 11110011

(-49)

Signed magnitude = 1110001 — remains same bcoz it is sign bit

" 1's complement = 1001110

" 2's complement = 1001111

Note: In 1's and 2's complement of a +ve no., representation is same for signed magnitude representation.

* Arithmetic Addition:

The rule for adding no. in the signed complement system doesnot require a comparison or subtraction but only addition.

The addition of two signed binary no. with -ve numbers represented in signed 2's complement form is obtained from the addition of the two numbers including their signed bit.

A carry out of the signed bit position is discarded.

(Case-1)

+6	00000110
+13	00001101
+19	00010011

(Case-2)

-6	(11111001 + 1)	= 11111010 (2's comp.)
+13	00001101	= 00001101
+7		1 00000111 ⇒ 00000111 (+7)

discarded carry

Note: (2's of +ve sign bit is same as itself)

(Case-3) +6 00000110 = 00000110

−13 (11110010)+1 = 11110011

−7 + 11111001

2's comp. of (11111001) = (10000110 + 1) [1's +1]

= 10000111

= −7

Sign bit.

(Case-4) −6 ~~1111~~ 2's of (−6) = ~~11111010~~

−13 2's of (−13) =

−19

2's of (−6) = (1's of (−6) + 1) = 11111001 + 1

2's of (−13) = (1's of (−13) + 1) = 11110010 + 1

2's of (−6) = 11111010 = 11111010

2's of (−13) = 111101101 = 11110011

+

1 11101101

discarded carry

Ans = 2's of (11101101)

= (10010010 + 1)

= (10010011)

= −9

Sign bit.

ration

not

ented

n of

mp.)

111

(i) −57
(ii) +57
(iii) −69

} Signed magnitude, signed 1's, signed 2's

(i) −57

Signed magnitude = 1111001

Signed 1's = 1000110

Signed 2's = 1000111

(iii) −69

Signed magnitude = 11000101

Signed 1's = 10111010

Signed 2's = 10111011

(ii) +57

Signed magnitude = 0111001

Signed 1's = 0111001

Signed 2's = 0111001

* BCD Subtraction:

The Subtraction of two BCD no. A and B can be performed by using.

(i) 9's complement method

(ii) 10's complement method

(Q) Subtract 3 from 7. i.e. $(7)_{10} - (3)_{10}$ in BCD.

$$\begin{array}{r} 7 \\ +6 \\ \hline 13 \\ 61 \\ \hline 4 \\ \hline \end{array}$$

BCD of 7 = 0111

9's complement of $(3)_{10} = (9-3)_{10} = (6)_{10}$

BCD of 6 = 0110

$$\begin{array}{r} 0111 \\ +\ 0110 \\ \hline 1101 \\ 0110 \\ \hline 10011 \\ +1 \\ \hline 0100 \\ \hline \end{array}$$

Invalid BCD add 6 (0110)

end around carry

Ans

(Q) Perform $(83)_{10} - (21)_{10}$ using 9's complement in BCD form.

9's complement of $(21)_{10} = (99-21)_{10} = (78)_{10}$

$$\begin{array}{r} 83 \\ +\ 78 \\ \hline 161 \\ 1 \\ \hline (62)_{10} \\ \hline \end{array}$$

$$\begin{array}{r|r} 1000 & 0011 \\ +\ 0111 & +\ 1000 \\ \hline 10000 & 1011 \\ 0110 & 0110 \\ \hline 1\underbrace{0110}_{6} & 10001 \\ & +1 \\ \hline & \underbrace{0010}_{2} \end{array}$$

$= (62)_{10}$

Ans

(Q) 54 − 28

9's complement of (28) = (99 − 28) = $(71)_{10}$

$$\begin{array}{r} 54 \\ +\ 71 \\ \hline 1\,25 \\ \hookrightarrow 1 \\ \hline 26 \end{array} \qquad \begin{array}{r} 0101 \\ 0111 \\ \hline 1100 \\ 0110 \\ \hline 10010 \end{array} \qquad \begin{array}{r} 0100 \\ 0001 \\ \hline 0101 \\ +\ 1 \\ \hline \underbrace{0110}_{6} \end{array}$$

$1\underbrace{0010}_{2}$

$= (26)_{10}$ Ans

$(54 - 28) = 26 = (0010\ 0110)_{BCD}$

Q) $(54 - 28)$ using 10's complement.

10's complement of $(28) = (9\text{'s comp.} + 1)$

$= (99 - 28) + 1$

$= 71 + 1 = 72$

$$\begin{array}{r} 54 \\ +\ 72 \\ \hline 1,26 \end{array} \qquad \begin{array}{r} 0101 \\ 0111 \\ \hline 1100 \\ 0110 \\ \hline 1\underbrace{0010} \end{array} \qquad \begin{array}{r} 0100 \\ 0010 \\ \hline \underbrace{0110} \end{array}$$

discard carry

discard carry

Q) $(21 - 83)$ using 9's complement.

9's complement of 83 $= (99 - 83) = (16)_{10}$

$$\begin{array}{r} 21 \\ +\ 16 \\ \hline 37 \end{array}$$

$$\boxed{\begin{aligned} \text{Ans} &= -(9\text{'s of } 37) \\ &= -(99-37) \\ &= -62 \end{aligned}}$$

$$\begin{array}{r} 0010 \\ 0001 \\ + \hline 0011 \end{array} \qquad \begin{array}{r} 0001 \\ 0110 \\ + \hline 0111 \end{array}$$

9's complement of 0011 0111 =

$$\begin{array}{r} 1001 \\ -0011 \\ \hline 0110 \end{array} \qquad \begin{array}{r} 1001 \\ -0111 \\ \hline 0010 \end{array}$$

Answer = − (0110 0010) BCD

---

* Boolean Algebra & logic Gates

⇒ Basic theorem and properties of Boolean Algebra :

* $x+0 = x$
* $x \cdot 1 = x$
* $x+x' = 1$
* $x \cdot x' = 0$
* $x+x = x$
* $x \cdot x = x$
* $x+1 = 1$
* $(x')' = x$
* $x+xy = x$
* $x+yz = (x+y)(x+z)$ Distributive Theorem
* $(x+y)(x+z) = x+yz$ Transposition Theorem.

Ques - 1) $AB + A\bar{B}C + A\bar{B}\bar{C}$

$= AB + A\bar{B}(C+\bar{C})$ $\quad (\because C+\bar{C} = 1)$

$= AB + A\bar{B}$

$= A(B+\bar{B})$ $\quad (\because B+\bar{B} = 1)$

$= A$

Ques - 2) $A\bar{B} + AB\bar{C} + A\bar{B}\bar{C}D$

$= A\bar{B}(1+\bar{C}D) + AB\bar{C}$

$= A\bar{B} + AB\bar{C}$ $\quad (\because 1+x = 1)$

$= A\bar{B} + AB\bar{C}$

$= A(\bar{B} + B\bar{C})$

$= A[\overset{1}{\cancel{(\bar{B}+B)}}(\bar{B}+\bar{C})]$

$= A(\bar{B}+\bar{C})$

$= A\bar{B} + A\bar{C}$

Ques - 3) $(A+B)(A+C)$ (Prove Transposition Theorem)

$= A\cdot A + A\cdot C + A\cdot B + B\cdot C$

$= A + A\cdot C + A\cdot B + B\cdot C$

$= A(1 + AC + AB) + B\cdot C$

$= (A + B\cdot C)$

(Ques - 4) $\overbrace{(A+B+C)(A+\overline{B}+C)}^{(A+C+B\overline{B})}(A+B+\overline{C}) = (A+BC)$

$$= (A\cdot A + A\cdot\overline{B} + A\cdot C + B\cdot A + B\cdot\overline{B} + BC + C\cdot A + C\overline{B} + C\cdot C)(A+B+\overline{C})$$

$$= (A + A(\cancel{\overline{B}+B})^{1} + A(\cancel{C+C})^{C} + C(\cancel{B+\overline{B}})^{1} + \cancel{B\cdot\overline{B}}^{0} + \cancel{C\cdot C}^{C})(A+B+\overline{C})$$

$$= ((\cancel{A+A})^{A} + AC + (\cancel{C+C})^{C})(A+B+\overline{C})$$

$$= [A(1+C) + C](A+B+C) \quad (\because (1+C) = 1)$$

$$= (A+C)(A+B+\overline{C}) \quad (\because x+y+1 = 1)$$

$$= (A+C)(A+(B+\overline{C})) \quad (\because (x+y)(x+z) = x+yz)$$

$$= A+C(B+\overline{C})$$

$$= A + CB + \cancel{C\overline{C}}^{0}$$

$$= (A+BC)$$

(Ques - 5) $AB + \overline{A}\overline{B} + A\overline{B}$

$$= AB + \overline{B}(\cancel{\overline{A}+A})^{1}$$

$$= AB + \overline{B}$$

$$= \overline{B} + AB \quad (\because (x+yz) = (x+y)(x+z))$$

$$= (\overline{B}+A)(\cancel{\overline{B}+B})^{1}$$

$$= \cancel{A+B}\ (\overline{B}+A)$$

(Ques - 6) $AB + \bar{A}C + BC = (AB + \bar{A}C)$

* Demorgan Theorem:

* $\overline{x+y} = \bar{x} \cdot \bar{y}$
* $(\overline{xy}) = \bar{x} + \bar{y}$

Canonical Form and Standard Form:

Minterms (m): Minterms are defined as the product terms which contains all the variables either in normal form or complement form.

Consider two binary variables x and y combined with an AND operation. Since each variable may appear in either form, there are four possible combinations – $\bar{X}\bar{Y}$, $\bar{X}Y$, $X\bar{Y}$ and XY. Each of these four AND terms is called minterms or a standard product. In a similar manner n variables can be combined to form 2^n minterms.

0 ⟶ represents complement variable

1 ⟶ represent normal variable.

Maxterms (M) : Max terms are defined as the sum terms which contains all the variables either in normal form or in complement form.

Consider two binary variables x and y combined with OR operations. Since each variable may appear in either form, there are four possible combinations - $\overline{X}+\overline{Y}$, $\overline{X}+Y$, $X+\overline{Y}$ and $X+Y$,

0 → represent normal variable
1 → represent complement variable

A	B	m	M
0	0	m_0	M_0
0	1	m_1	M_1
1	0	m_2	M_2
1	1	m_3	M_3

Note : Boolean Function expressed as sum of minterms or product ~~to~~ of maxterms are said to be in canonical form.

Sum of Minterms (sum of product terms) ;
The minterms whose sum defines the Boolean function. Boolean function are those, which give the 1's of the function in a truth table.

(Q) Express the Boolean function $F = A + \overline{B}C$ as a sum of minterms or canonical form of SOP.

$F = A + \overline{B}C$

$F = A + \bar{B}C$

1st term → B and C is missing.

$A \rightarrow A(B+\bar{B})(C+\bar{C})$

$A \rightarrow ABC + AB'C + ABC' + AB'C'$ ——— (i)

2nd term → A is missing

$B'C \rightarrow (A+A')B'C$

$\rightarrow AB'C + A'B'C$ ——— (ii)

$F = ABC + \underbrace{AB'C + ABC' + AB'C' + AB'C}_{\text{merging}} + A'B'C$

$F = \underbrace{ABC}_{\substack{111 \\ 7}} + \underbrace{AB'C}_{\substack{101 \\ 5}} + \underbrace{ABC'}_{\substack{110 \\ 6}} + \underbrace{AB'C'}_{\substack{100 \\ 4}} + \underbrace{A'B'C}_{\substack{001 \\ 1}}$

$\sum m(1,4,5,6,7)$

$\pi M(0,2,3)$

◉ Express the Boolean function $F = xy + \bar{x}z$ as a product of maxterms. (POS)

$F = xy + \bar{x}z$

$F = (xy + \bar{x})(xy + z)$

$F = \cancel{(x+\bar{x})}(\bar{x}+y)(x+z)(z+y)$ ①

$F = (\bar{x}+y)(x+z)(z+y)$

1st term → z is missing.

$(\bar{x}+y) \rightarrow (\bar{x}+y+z\bar{z}) = (\bar{x}+y+z)(\bar{x}+y+\bar{z})$

2nd term $\longrightarrow$ y is missing.

$(x+z) \rightarrow (x+z+y\bar{y}) \Rightarrow (x+z+y)(x+z+\bar{y})$

$\Rightarrow (x+y+z)(x+\bar{y}+z)$

3rd term $\longrightarrow$ x is missing.

$(z+y) \longrightarrow (y+z+x\bar{x}) = (y+z+x)(y+z+\bar{x})$

$= (x+y+z)(\bar{x}+y+z)$

$F = (\bar{x}+y+z)(\bar{x}+y+\bar{z})(x+y+z)(x+\bar{y}+z)(x+y+z)$

$(\bar{x}+y+z)$

merging

$= (\bar{x}+y+z)(\bar{x}+y+\bar{z})(x+y+z)(x+\bar{y}+z)$

$\underbrace{1\ 0\ 0}_{4} \quad \underbrace{1\ 0\ 1}_{5} \quad \underbrace{0\ 0\ 0}_{0} \quad \underbrace{0\ 1\ 0}_{2}$

$\Rightarrow \pi M(4,5,0,2)$

$\Rightarrow \pi M(0,2,4,5)$

(Q)(i) $F = A + B'C$ canonical form of POS.

(Q)(ii) $F = xy + \bar{x}z$ Canonical form of SOP.

(i) $F = A + B'C$ convert (in POS)

$= (A+B')(A+C)$

1st term $\longrightarrow$ C is missing

$(A+B') \longrightarrow (A+B'+C\bar{C}) = (A+B'+C)(A+B'+\bar{C})$

2^{nd} term $\rightarrow$ B is missing.

$$(A+C) \longrightarrow (A+C+B\bar{B}) = (A+C+B)(A+C+\bar{B})$$

$$F = (A+\bar{B}+C)(A+\bar{B}+\bar{C})(A+B+C)(A+\bar{B}+C)$$

$$F = \underbrace{(A+\bar{B}+C)}_{\underbrace{0\ 1\ 0}_{2}}\underbrace{(A+\bar{B}+\bar{C})}_{\underbrace{0\ 1\ 1}_{3}}\underbrace{(A+B+C)}_{\underbrace{0\ 0\ 0}_{0}}$$

$$= \pi M(0, 2, 3)$$

(ii) $F = xy + \bar{x}z$ (convert in SOP)

1^{st} term $\longrightarrow$ z is missing.

$$xy \rightarrow xy(z+\bar{z}) = xyz + xy\bar{z}$$

2^{nd} term $\rightarrow$ y is missing.

$$\bar{x}z \longrightarrow \bar{x}z(y+\bar{y}) = \bar{x}zy + \bar{x}\bar{y}z$$

$$F = \underbrace{xyz}_{\underbrace{111}_{7}} + \underbrace{xy\bar{z}}_{\underbrace{110}_{6}} + \underbrace{\bar{x}yz}_{\underbrace{011}_{3}} + \underbrace{\bar{x}\bar{y}z}_{\underbrace{001}_{1}}$$

$$\Sigma m(1, 3, 6, 7)$$

$\boxed{\text{Exercise}}$

(Q) Express the function in canonical form of POS:

$$F = (A+B)(A+C')$$

Canonical form of SOP and POS

(i) $F = A' + B'C'$

(ii) $F = (\bar{A} + \bar{B})(\bar{B} + \bar{C})$

(Q) Express $F = (A+C')(A'+B')$ in canonical for of POS or canonical form of product of Maxterms.

1st term missing 1 variable is B :

$(A+C') \longrightarrow (A+C'+BB') \Rightarrow (A+C+B)(A+C'+B')$

$\Rightarrow (A+B+C')(A+B'+C')$ —— ①

2nd term missing C :

$(A'+B') \longrightarrow (A'+B'+CC')$

$\longrightarrow (A'+B'+C)(A'+B'+C')$ —— ②

$F = \underbrace{(A+B+C')}_{\substack{0\ 0\ 1 \\ 1}}\underbrace{(A+B'+C')}_{\substack{0\ 1\ 1 \\ 3}}\underbrace{(A'+B'+C)}_{\substack{1\ 1\ 0 \\ 6}}\underbrace{(A'+B'+C')}_{\substack{1\ 1\ 1 \\ 7}}$

$= \pi M(1,3,6,7)$

* Karnaugh Map (K-map)

- It is used when output is zero, 1 and don't care (X)
- In K-map gray code representation is used.
- K-map is a graphical method.
- It is used when no. of variables are 2, 3, 4, 5.

⇒ Two Variable K-map:

A \ B	0	1
0	00 0	01 1
1	10 2	11 3

Possibilities $= 2^n = 2^2 = 4$.

where, n = no. of variables

⇒ Three-Variable K-map:

A \ BC	00	01	11	10
0	000 0	001 1	011 3	010 2
1	100 4	101 5	111 7	110 6

Possib. $= 2^n = 2^3 = 8$

⇒ Four-Variable K-map:

AB \ CD	00	01	11	10
00	0000 0	0001 1	0011 3	0010 2
01	0100 4	0101 5	0111 7	0110 6
11	1100 12	1101 13	1111 15	1110 14
10	1000 8	1001 9	1011 11	1010 10

Possibilities

$2^n = 2^4 = 16$

⇒ For two variable K-map:

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	$\bar{A}\cdot\bar{B}$ 00	$\bar{A}\cdot B$ 01
A 1	$A\cdot\bar{B}$ 10	$A\cdot B$ 11

⇑ SOP (Minterms)

A \ B	B 0	$\bar{B}$ 1
A 0	$A+B$ 0 0	$A+\bar{B}$ 0 1
$\bar{A}$ 1	$\bar{A}+B$ 1 0	$\bar{A}+\bar{B}$ 1 1

⇑ POS (Maxterms)

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	$\bar{A}\bar{B}\bar{C}$ 0	$\bar{A}\bar{B}C$ 1	$\bar{A}BC$ 3	$\bar{A}B\bar{C}$ 2
A 1	$A\bar{B}\bar{C}$ 4	$A\bar{B}C$ 5	ABC 7	$AB\bar{C}$ 6

(Minterms)

A \ BC	$B+C$ 00	$B+\bar{C}$ 01	$\bar{B}+\bar{C}$ 11	$\bar{B}+C$ 10
A 0	$A+B+C$ 0	$A+B+\bar{C}$ 1	$A+\bar{B}+\bar{C}$ 3	$A+\bar{B}+C$ 2
$\bar{A}$ 1	$\bar{A}+B+C$ 4	$\bar{A}+B+\bar{C}$ 5	$\bar{A}+\bar{B}+\bar{C}$ 7	$\bar{A}+\bar{B}+C$ 6

(Maxterms)

* Plotting a K-map:

Representation of Truth Table on K-map:

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

A \ B	0	1
0	0 (0)	1 (1)
1	1 (2)	1 (3)

Plotting of standard SOP and POS on K-map:

$$F = A'B'C + AB'C + A'BC$$

001   101   011

$$= \Sigma m(1,3,5)$$

A \ BC	00	01	11	10
0	(0)	1 (1)	1 (3)	(2)
1	(4)	1 (5)	(7)	(6)

$$F = (A+B+C)(\bar{A}+\bar{B}+C)$$
$$\quad\; 0\;\;0\;\;0 \qquad 1\;\;1\;\;0$$

$$= \pi M(0,6)$$

A \ BC	00	01	11	10
0	0 (0)	(1)	(3)	(2)
1	(4)	(5)	(7)	0 (6)

* Grouping of shells for simplification :

Grouping is nothing but combining adjacent shells with containing one's for SOP simplification and 0's for POS simplification.

Grouping shells are selected in the form of powers of 2 $(2^0, 2^1, 2^2, 2^3, ---)$.

⇒ Grouping of ~~at~~ two adjacent shells (Pair) :

SOP

	A \ B	$\bar{B}$ 0	B 1
$\bar{A}$	0	1	1
A	1		

POS

	A \ B	B 0	$\bar{B}$ 1
A	0	0	0
$\bar{A}$	1		

$$F = \bar{A}\bar{B} + \bar{A}B$$
$$= \bar{A}(\cancel{\bar{B}+B})^{1}$$
$$= \bar{A}$$

$$F = (A+B)(A+\bar{B})$$
$$= A + \cancel{B\bar{B}}^{0}$$
$$= A$$

SOP

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	(1)	(3)	1 (2)
A 1	(4)	(5)	(7)	(6)

$$\boxed{F = \bar{A}\bar{C}}$$

$$F = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C}$$
$$= \bar{A}\bar{C}(\underbrace{B + \bar{B}}_{1})$$
$$= \bar{A}\bar{C}$$

POS

AB \ CD	$C+D$ 00	$C+\bar{D}$ 01	$\bar{C}+\bar{D}$ 11	$\bar{C}+D$ 10
$A+B$ 00	0 (0)	(1)	(3)	(2)
$A+\bar{B}$ 01	(4)	(5)	(7)	(6)
$\bar{A}+\bar{B}$ 11	(12)	(13)	(15)	(14)
$\bar{A}+B$ 10	0 (8)	(9)	(11)	(10)

$$\boxed{F = B+C+D}$$

$$F = (A+B+C+D)(\bar{A}+B+C+D)$$
$$= (\underbrace{A\bar{A}}_{0} + B+C+D)$$
$$= (B+C+D).$$

⇒ Grouping of four adjacent ~~ed~~ shells (quad.) :

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	1 (0)	1 (1)
A 1	1 (2)	1 (3)

$$\boxed{F = 1}$$

$$= \bar{A}\bar{B} + \bar{A}B + A\bar{B} + AB$$
$$= \bar{A}(\bar{B}+B) + A(\bar{B}+B)$$
$$= (\bar{A}+A) = 1$$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	1 (1)	1 (3)	1 (2)
A 1	(4)	(5)	(7)	(6)

$$\boxed{F = \bar{A}}$$

A \ BC	$B+C$ 00	$B+\bar{C}$ 01	$\bar{B}+\bar{C}$ 11	$\bar{B}+C$ 10
A 0	0 (0)	(1)	(3)	0 (2)
$\bar{A}$ 1	0 (4)	(5)	(7)	0 (6)

$$\boxed{F = C}$$

AB \ CD	$C+D$ 00	$C+\bar{D}$ 01	$\bar{C}+\bar{D}$ 11	$\bar{C}+D$ 10
$A+B$ 00	0 (0)	(1)	(3)	0 (2)
$A+\bar{B}$ 01	(4)	(5)	(7)	(6)
$\bar{A}+\bar{B}$ 11	(12)	(13)	(15)	(14)
$\bar{A}+B$ 10	0 (8)	(9)	(11)	0 (10)

$$\boxed{F = B+D}$$

AB \ CD	$C+D$ 00	$C+\bar{D}$ 01	$\bar{C}+\bar{D}$ 11	$\bar{C}+D$ 10
$A+B$ 00	0_{0}	0_{1}	$_{3}$	$_{2}$
$A+\bar{B}$ 01	$_{4}$	$_{5}$	$_{7}$	$_{6}$
$\bar{A}+\bar{B}$ 11	$_{12}$	$_{13}$	$_{15}$	$_{14}$
$\bar{A}+B$ 10	0_{8}	0_{9}	$_{11}$	$_{10}$

$$\boxed{F = B + C}$$

$$F = (A+B+C+D)(A+B+C+\bar{D})(\bar{A}+B+C+D)(\bar{A}+B+C+\bar{D})$$

$$= (A+B+C+D\bar{D})(\bar{A}+B+C+D\bar{D})$$

$$= (\underbrace{A\bar{A}}_{0} + B + C + \underbrace{D\bar{D}}_{0})$$

$$= (B+C)$$

⇒ Grouping of eight adjacent shell :

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1_{0}	$_{1}$	$_{3}$	1_{2}
$\bar{A}B$ 01	1_{4}	$_{5}$	$_{7}$	1_{6}
AB 11	1_{12}	$_{13}$	$_{15}$	1_{14}
$A\bar{B}$ 10	1_{8}	$_{9}$	$_{11}$	1_{10}

$$\boxed{F = \bar{D}}$$

① Priority Order :

- Octet
- Quad
- Pair
- Single Term
- Redundant Group

⇒ Redundant Group:

It is a group in which all the elements in this group are covered by some other groups.

* $F(A,B,C) = \Sigma m(0,1,2,4,7)$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	1 (1)	(3)	1 (2)
A 1	1 (4)	(5)	1 (7)	(6)

Three pairs ~~are~~ and one single term are formed.

$$\boxed{F = \bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}\bar{C} + ABC}$$

* $F(A,B,C) = \Sigma m(0,1,5,6,7)$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	1 (1)	(3)	(2)
A 1	(4)	1 (5)	1 (7)	1 (6)

$$\boxed{F = \bar{A}\bar{B} + \bar{B}C + AB}$$

* $F(A,B,C,D) = \Sigma m(0,1,3,5,7,8,9,11,13,15)$

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 (0)	1 (1)	1 (3)	(2)
$\bar{A}B$ 01	(4)	1 (5)	1 (7)	(6)
AB 11	(12)	1 (13)	1 (15)	(14)
$A\bar{B}$ 10	1 (8)	1 (9)	1 (11)	(10)

$$\boxed{F = \underbrace{D}_{\text{(octet)}} + \underbrace{\bar{B}\bar{C}}_{\text{(quad)}}}$$

* $F(A,B,C) = \sum m(1,3,6,7)$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	0	1 (1)	1 (3)	2
A 1	4	5	1 (7)	1 (6)

$$\boxed{F = \bar{A}C + AB}$$

* $F(A,B,C,D) = \prod M(1,5,6,7,11,12,13,15)$

AB \ CD	$C+D$ 00	$C+\bar{D}$ 01	$\bar{C}+\bar{D}$ 11	$\bar{C}+D$ 10
$A+B$ 00	0	0 (1)	3	2
$A+\bar{B}$ 01	4	0 (5)	0 (7)	0 (6)
$\bar{A}+\bar{B}$ 11	0 (12)	0 (13)	0 (15)	14
$\bar{A}+B$ 10	8	9	0 (11)	10

Here quad is ~~there~~ redundant so, it is removed and there is four pair formed.

$$\boxed{F = (\bar{A}+\bar{B}+C)(A+\bar{B}+\bar{C})(\bar{A}+\bar{C}+\bar{D})(A+C+\bar{D})}$$

(Q) Simplify the Boolean function $F(A,B,C,D) = \prod M(3,4,5,7,9,13,14,15)$ using K-map.

AB \ CD	$C+D$ 00	$C+\bar{D}$ 01	$\bar{C}+\bar{D}$ 11	$\bar{C}+D$ 10
$A+B$ 00	0	1	0 (3)	2
$A+\bar{B}$ 01	0 (4)	0 (5)	0 (7)	6
$\bar{A}+\bar{B}$ 11	12	0 (13)	0 (15)	0 (14)
$\bar{A}+B$ 10	8	0 (9)	11	10

Here quad is redundant. so, it is removed and there is four pair formed.

$$F = (A+\bar{B}+C)(\bar{A}+C+\bar{D})(\bar{A}+\bar{B}+\bar{C})(\bar{C}+\bar{D}+A)$$

$$F = (A+\bar{B}+C)(\bar{A}+C+\bar{D})(\bar{A}+\bar{B}+\bar{C})(A+\bar{C}+\bar{D})$$

(Q)① Simplify the Boolean function $F(w,x,y,z) = \Sigma m(0,1,2,4,5,6,8,9,12,13,14)$

wx \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{w}\bar{x}$ 00	1 (0)	1 (1)	(3)	1 (2)
$\bar{w}x$ 01	1 (4)	1 (5)	(7)	1 (6)
wx 11	1 (12)	1 (13)	(15)	1 (14)
$w\bar{x}$ 10	1 (8)	1 (9)	(11)	(10)

1 octave, 2 quad

$$\boxed{F = \underbrace{\bar{y}}_{\text{oct.}} + \underbrace{\bar{w}\bar{z}}_{\text{quad.}} + \underbrace{x\bar{z}}_{\text{quad.}}}$$

(Q)② Find the canonical form of POS from the canonical form of given SOP.

(∵ SOP is the complement of POS (and) (∵ POS is the complement of SOP)

Soln:

$F(A,B,C,D) = \Sigma m(0,2,4,6)$

$\therefore F(A,B,C,D) = \Pi M(1,3,5,7,8,9,10,11,12,13,14,15)$

AB \ CD	C+D 00	C+$\bar{D}$ 01	$\bar{C}$+$\bar{D}$ 11	$\bar{C}$+D 10
A+B 00	(0)	0 (1)	0 (3)	(2)
A+$\bar{B}$ 01	(4)	0 (5)	0 (7)	(6)
$\bar{A}$+$\bar{B}$ 11	0 (12)	0 (13)	0 (15)	0 (14)
$\bar{A}$+B 10	0 (8)	0 (9)	0 (11)	0 (10)

2 octave

$$\boxed{F = (\bar{A})(\bar{D})}$$

$$F = \Sigma m(0,2,4,6)$$

$$= \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}B\bar{C}\bar{D} + \bar{A}BC\bar{D}$$

$$F(A,B,C,D) = \pi M(1,3,5,7,8,9,10,11,12,13,14,15)$$

$$= (A+B+C+\bar{D})(A+B+\bar{C}+\bar{D})(A+\bar{B}+C+\bar{D})(A+\bar{B}+\bar{C}+\bar{D})$$
$$(\bar{A}+B+C+D)(\bar{A}+B+C+\bar{D})(\bar{A}+B+\bar{C}+D)(\bar{A}+B+\bar{C}+\bar{D})$$
$$(\bar{A}+\bar{B}+C+D)(\bar{A}+\bar{B}+C+\bar{D})(\bar{A}+\bar{B}+\bar{C}+D)(A+\bar{B}+\bar{C}+\bar{D})$$

(Q)③

		$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$	00	1 (0)	(1)	(3)	1 (2)
$\bar{A}B$	01	(4)	1 (5)	1 (7)	1 (6)
AB	11	(12)	1 (13)	1 (15)	1 (14)
$A\bar{B}$	10	1 (8)	(9)	(11)	1 (10)

Three quad are formed :

$$\boxed{F = BD + C\bar{D} + \bar{B}\bar{D}}$$

(Q)④

		$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$	00	1 (0)	(1)	1 (3)	1 (2)
$\bar{A}B$	01	1 (4)	(5)	(7)	1 (6)
AB	11	(12)	1 (13)	1 (15)	(14)
$A\bar{B}$	10	1 (8)	1 (9)	1 (11)	1 (10)

4 quads are formed :

$$\boxed{F = \bar{A}\bar{D} + \bar{B}C + AD + A\bar{B}}$$

(Q-5)

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1_{0}	1_{1}	$_{3}$	1_{2}
$\bar{A}B$ 01	1_{4}	1_{5}	$_{7}$	1_{6}
AB 11	1_{12}	1_{13}	1_{15}	1_{14}
$A\bar{B}$ 10	1_{8}	1_{9}	1_{11}	$_{10}$

1 octet, 3 quad are formed

$$\boxed{F = \bar{C} + AD + \bar{A}\bar{D} + B\bar{D}}$$

$\bar{B} + \bar{C} + \bar{D})$

$\bar{B} + \bar{C} + \bar{D})$

$\bar{C} + \bar{D})$

(Q-6) Simplify $AB + A'C + BC$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	$_{0}$	1_{1}	1_{3}	$_{2}$
A 1	$_{4}$	$_{5}$	1_{7}	1_{6}

$$\boxed{F = \bar{A}C + AB}$$

(Q-7) Simplify the Boolean Function:

$$F = \bar{A}\bar{B}\bar{C} + \bar{B}C\bar{D} + \bar{A}BC\bar{D} + A\bar{B}\bar{C} \quad \text{—— (1)}$$

$$F(A,B,C,D) = ACD + \bar{A}B + D'$$

from (1), we can draw:

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1_{0}	1_{1}	$_{3}$	1_{2}
$\bar{A}B$ 01	$_{4}$	$_{5}$	$_{7}$	1_{6}
AB 11	$_{12}$	$_{13}$	$_{15}$	$_{14}$
$A\bar{B}$ 10	1_{8}	1_{9}	$_{11}$	1_{10}

2 quad and one pair are formed:

$$\boxed{F = \bar{B}\bar{D} + \bar{B}\bar{C} + \bar{A}C\bar{D}}$$

(Q-8)

AB \ CD	$\bar C\bar D$ 00	$\bar C D$ 01	CD 11	$C\bar D$ 10
$\bar A\bar B$ 00	1 (0)	(1)	(3)	1 (2)
$\bar A B$ 01	1 (4)	1 (5)	1 (7)	1 (6)
AB 11	1 (12)	(13)	1 (15)	1 (14)
$A\bar B$ 10	1 (8)	(9)	1 (11)	1 (10)

1 Octet and 2 quad are formed !

$$\boxed{F = \bar D + \bar A B + AC}$$

(1) ⇒ Simplify the following Boolean function into :

(a) SOP

(b) POS

$$F(A,B,C,D) = \sum m(0,1,2,5,8,9,10)$$

Soln:

$$F(A,B,C,D) = \sum m(0,1,2,5,8,9,10)$$

$$= \bar A\bar B\bar C\bar D + \bar A\bar B\bar C D + \bar A\bar B C\bar D + \bar A B\bar C D + A\bar B\bar C\bar D$$

$$+ A\bar B\bar C D + A\bar B C\bar D$$

$$F(A,B,C,D) = \pi M(3,4,6,7,11,12,13,14,15)$$

$$= (A+B+\bar C+\bar D)(A+\bar B+C+D)(A+\bar B+\bar C+D)$$

$$(A+\bar B+\bar C+\bar D)(\bar A+B+\bar C+\bar D)(\bar A+\bar B+C+D)(\bar A+\bar B+C+\bar D)$$

$$(\bar A+\bar B+\bar C+D)(\bar A+\bar B+\bar C+\bar D)$$

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 (0)	1 (1)	(3)	1 (2)
$\bar{A}B$ 01	(4)	1 (5)	(7)	(6)
AB 11	(12)	(13)	(15)	(14)
$A\bar{B}$ 10	1 (8)	1 (9)	(11)	1 (10)

$$\boxed{F = \underbrace{\bar{B}\bar{D}}_{\text{quad.}} + \underbrace{\bar{A}\bar{C}D}_{\text{pair}} + \underbrace{A\bar{B}\bar{C}}_{\text{pair}}}$$

AB \ CD	$C+D$ 00	$C+\bar{D}$ 01	$\bar{C}+\bar{D}$ 11	$\bar{C}+D$ 10
$A+B$ 00	(0)	(1)	0 (3)	(2)
$A+\bar{B}$ 01	0 (4)	(5)	0 (7)	0 (6)
$\bar{A}+\bar{B}$ 11	0 (12)	0 (13)	0 (15)	0 (14)
$\bar{A}+B$ 10	(8)	(9)	0 (11)	(10)

$$\boxed{F = \underbrace{(\bar{B}+D)}_{\text{quad.}}\underbrace{(\bar{C}+\bar{D})}_{\text{quad.}}\underbrace{(\bar{A}+\bar{B})}_{\text{quad.}}}$$

② ⇒ Minimize the following expression using K-map.

$$F = \Sigma m(0,1,2,5,13,15).$$

Soln:

Given: $F = \Sigma m(0,1,2,5,13,15)$

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 (0)	1 (1)	(3)	1 (2)
$\bar{A}B$ 01	(4)	1 (5)	(7)	(6)
AB 11	(12)	1 (13)	1 (15)	(14)
$A\bar{B}$ 10	(8)	(9)	(11)	(10)

$$\boxed{F = \underbrace{\bar{A}\bar{B}\bar{D}}_{\text{pair}} + \underbrace{\bar{A}\bar{C}D}_{\text{pair}} + \underbrace{ABD}_{\text{pair}}}$$

* Don't Care Conditions (X) :

Minimum number of don't care are used.

$$F(A,B) = \Sigma m(1,3) + \Sigma d(2)$$

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	(0)	1 (1)
A 1	X (2)	1 (3)

$$\boxed{F = B}$$

$$F(A,B) = \sum m(0,3) + \sum d(2)$$

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	1_{0}	$_{1}$
A 1	X_{2}	1_{3}

$$\boxed{F = A + \bar{B}}$$

$$F(A,B,C) = \sum(0,1,2,5,7) + \sum d(3,6)$$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1_{0}	1_{1}	X_{3}	1_{2}
A 1	$_{4}$	1_{5}	1_{7}	X_{6}

$$\boxed{F = \bar{A} + C}$$

$$F(A,B,C) = \prod M(0,1,6,7) \cdot \prod d(3,5)$$

A \ BC	$B+C$ 00	$B+\bar{C}$ 01	$\bar{B}+\bar{C}$ 11	$\bar{B}+C$ 10
A 0	0_{0}	0_{1}	X_{3}	$_{2}$
$\bar{A}$ 1	$_{4}$	X_{5}	0_{7}	0_{6}

$$F = (A + B)(\bar{A} + \bar{B})$$

$F(A,B,C,D) = \sum m(1,3,7,11,15) + \sum d(0,2,5)$

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	X (0)	1 (1)	1 (3)	X (2)
$\bar{A}B$ 01	(4)	X (5)	1 (7)	(6)
AB 11	(12)	(13)	1 (15)	(14)
$A\bar{B}$ 10	(8)	(9)	1 (11)	(10)

$$\boxed{F = \bar{A}D + CD}$$

$F(A,B,C) = \sum m(0,1,6,7) + \sum d(3,4,5)$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	1 (1)	X (3)	(2)
A 1	X (4)	X (5)	1 (7)	1 (6)

$$\boxed{F = A + \bar{B}}$$

(Q)

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	(0)	(1)	(3)	(2)
$\bar{A}B$ 01	(4)	1 (5)	1 (7)	1 (6)
AB 11	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$ 10	1 (8)	1 (9)	X (11)	X (10)

1 octet, 2 quad :

$$\boxed{F = A + BD + BC}$$

(Q)

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0	1 (1)	1 (3)	1 (2)
$\bar{A}B$ 01	1 (4)	5	7	6
AB 11	X (12)	X (13)	X (15)	14
$A\bar{B}$ 10	8	1 (9)	X (11)	X (10)

2 quad, 1 pair :

$$\boxed{F = \bar{B}D + \bar{B}C + B\bar{C}\bar{D}}$$

(Q)

$F(A,B,C,D,E) = \Sigma m(0,5,6,8,9,10,11,16,20,24,25,26,27,29,31)$

$\bar{A} = 0$

BC \ DE	$\bar{D}\bar{E}$ 00	$\bar{D}E$ 01	DE 11	$D\bar{E}$ 10
$\bar{B}\bar{C}$ 00	1 (0)	1	3	2
$\bar{B}C$ 01	4	1 (5)	7	1 (6)
BC 11	12	13	15	14
$B\bar{C}$ 10	1 (8)	1 (9)	1 (11)	1 (10)

$A = 1$

BC \ DE	$\bar{D}\bar{E}$ 00	$\bar{D}E$ 01	DE 11	$D\bar{E}$ 10
$\bar{B}\bar{C}$ 00	1 (16)	17	19	18
$\bar{B}C$ 01	1 (20)	21	23	22
BC 11	28	1 (29)	1 (31)	30
$B\bar{C}$ 10	1 (24)	1 (25)	1 (27)	1 (26)

Octet $(8,9,10,11,24,25,26,27) = B\bar{C}$

Single term $(5) = \bar{A}\bar{B}C\bar{D}E$

" $(6) = \bar{A}\bar{B}CD\bar{E}$

quad $(25,27,29,31) = ABE$

Pair $(16,20) = A\bar{B}\bar{D}\bar{E}$

2^{nd} quad $(0,8,16,24) = \bar{C}\bar{D}\bar{E}$

$\therefore F = B\bar{C} + \bar{A}\bar{B}C\bar{D}E + \bar{A}\bar{B}CD\bar{E} + ABE + A\bar{B}\bar{D}\bar{E} + \bar{C}\bar{D}\bar{E}$

(Q) $F(A,B,C,D,E) = \pi M(6,9,11,13,14,17,20,25,28,29,30)$

$A = 0$

BC \ DE	D+E 00	D+$\bar{E}$ 01	$\bar{D}$+$\bar{E}$ 11	$\bar{D}$+E 10
B+C 00	0	1	3	2
B+$\bar{C}$ 01	4	5	7	0 6
$\bar{B}$+$\bar{C}$ 11	12	0 13	15	0 14
$\bar{B}$+C 10	8	0 9	0 11	10

$\bar{A} = 1$

BC \ DE	D+E 00	D+$\bar{E}$ 01	$\bar{D}$+$\bar{E}$ 11	$\bar{D}$+E 10
B+C 00	16	0 17	19	18
B+$\bar{C}$ 01	0 20	21	23	22
$\bar{B}$+$\bar{C}$ 11	0 28	0 29	31	0 30
$\bar{B}$+C 10	24	0 25	27	26

27,

quad $(9,13,25,29) = (\bar{B}+D+\bar{E})$

Pair $(9,11) = (A+\bar{B}+C+\bar{E})$

Pair $(20,28) = (\bar{A}+\bar{C}+D+E)$

Pair $(14,6) = (A+\bar{C}+\bar{D}+E)$

Pair $(28,30) = (\bar{A}+\bar{B}+\bar{C}+E)$

Pair $(17,25) = (\bar{A}+C+D+\bar{E})$

$$F = (\bar{B}+D+\bar{E})(A+\bar{B}+C+\bar{E})(\bar{A}+\bar{C}+D+E)(A+\bar{C}+\bar{D}+E)(\bar{A}+\bar{B}+\bar{C}+E)(\bar{A}+C+D+\bar{E})$$

Questions

(1) Minimize the Boolean function ~~for~~ $F(A,B,C,D,E) = \sum m(3,4,5,7,9,13,14,15)$

(2) $F(A,B,C,D,E) = \sum m(0,1,3,5,6,7,9,11,16,18,19,20,21,22,24,26)$

(3) To a minimum number of literals (variable):

(i) $\bar{A}\bar{C} + ABC + A\bar{C} + A\bar{B}$

(ii) $(\bar{X}\bar{Y} + Z) + Z + XY + WZ$

Soln (1) $F(A,B,C,D,E) = \sum m(3,4,5,7,9,13,14,15)$

$\bar{A} = 0$

BC \ DE	$\bar{D}\bar{E}$ 00	$\bar{D}E$ 01	DE 11	$D\bar{E}$ 10
$\bar{B}\bar{C}$ 00	(0)	(1)	1 (3)	(2)
$\bar{B}C$ 01	1 (4)	1 (5)	1 (7)	(6)
BC 11	(12)	1 (13)	1 (15)	1 (14)
$B\bar{C}$ 10	(8)	1 (9)	(11)	(10)

A = 1

BC \ DE	00	01	11	10
00	(16)	(17)	(19)	(18)
01	(20)	(21)	(23)	(22)
11	(28)	(29)	(31)	(30)
10	(24)	(25)	(27)	(26)

$F = \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}DE + \bar{A}BCD + \bar{A}B\bar{D}E$

Soln: (2)

$F(A,B,C,D,E) = \sum m(0,1,3,5,6,7,9,11,16,18,19,20,21,22,24,26)$

$\bar{A} = 0$

BC \ DE	$\bar{D}\bar{E}$ 00	$\bar{D}E$ 01	DE 11	$D\bar{E}$ 10
$\bar{B}\bar{C}$ 00	1 (0)	1 (1)	1 (3)	(2)
$\bar{B}C$ 01	(4)	1 (5)	1 (7)	1 (6)
BC 11	(12)	(13)	(15)	(14)
$B\bar{C}$ 10	(8)	1 (9)	1 (11)	(10)

A = 1

BC \ DE	$\bar{D}\bar{E}$ 00	$\bar{D}E$ 01	DE 11	$D\bar{E}$ 10
$\bar{B}\bar{C}$ 00	1 (16)	(17)	1 (19)	1 (18)
$\bar{B}C$ 01	1 (20)	1 (21)	(23)	1 (22)
BC 11	(28)	(29)	(31)	(30)
$B\bar{C}$ 10	1 (24)	(25)	(27)	1 (26)

quad(1,3,9,11) = $\bar{A}\bar{C}E$

quad(1,3,5,7) = $\bar{A}\bar{B}E$

quad(16,18,24,26) = $A\bar{C}\bar{E}$

quad(16, 20, 18, 22) = $A\bar{B}\bar{E}$

pair(0,1) = $\bar{A}\bar{B}\bar{C}\bar{D}$

pair(6,7) = $\bar{A}\bar{B}CD$

pair(20,21) = $A\bar{B}C\bar{D}$

pair(18,19) = $A\bar{B}\bar{C}D$

$$F = \bar{A}\bar{C}E + \bar{A}\bar{B}E + A\bar{C}\bar{E} + A\bar{B}\bar{E} + \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}CD + A\bar{B}C\bar{D} + A\bar{B}\bar{C}D$$

Soln: (3)

(i) $\bar{A}\bar{C} + ABC + A\bar{C} + A\bar{B}$ → $\bar{A}\bar{C}(B+\bar{B}) + ABC + A\bar{C}(B+\bar{B}) + A\bar{B}(C+\bar{C})$

$= \bar{A}\bar{C}B + \bar{A}\bar{C}\bar{B} + ABC + A\bar{C}B + A\bar{C}\bar{B} + A\bar{B}C + A\bar{B}\bar{C}$

$= \underbrace{\bar{A}\bar{C}B}_{2} + \underbrace{\bar{A}\bar{C}\bar{B}}_{0} + \underbrace{ABC}_{7} + \underbrace{AB\bar{C}}_{6} + \underbrace{A\bar{B}\bar{C}}_{4} + \underbrace{A\bar{B}C}_{5}$

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	(1)	(3)	1 (2)
A 1	1 (4)	1 (5)	1 (7)	1 (6)

$\boxed{F = A + \bar{C}}$

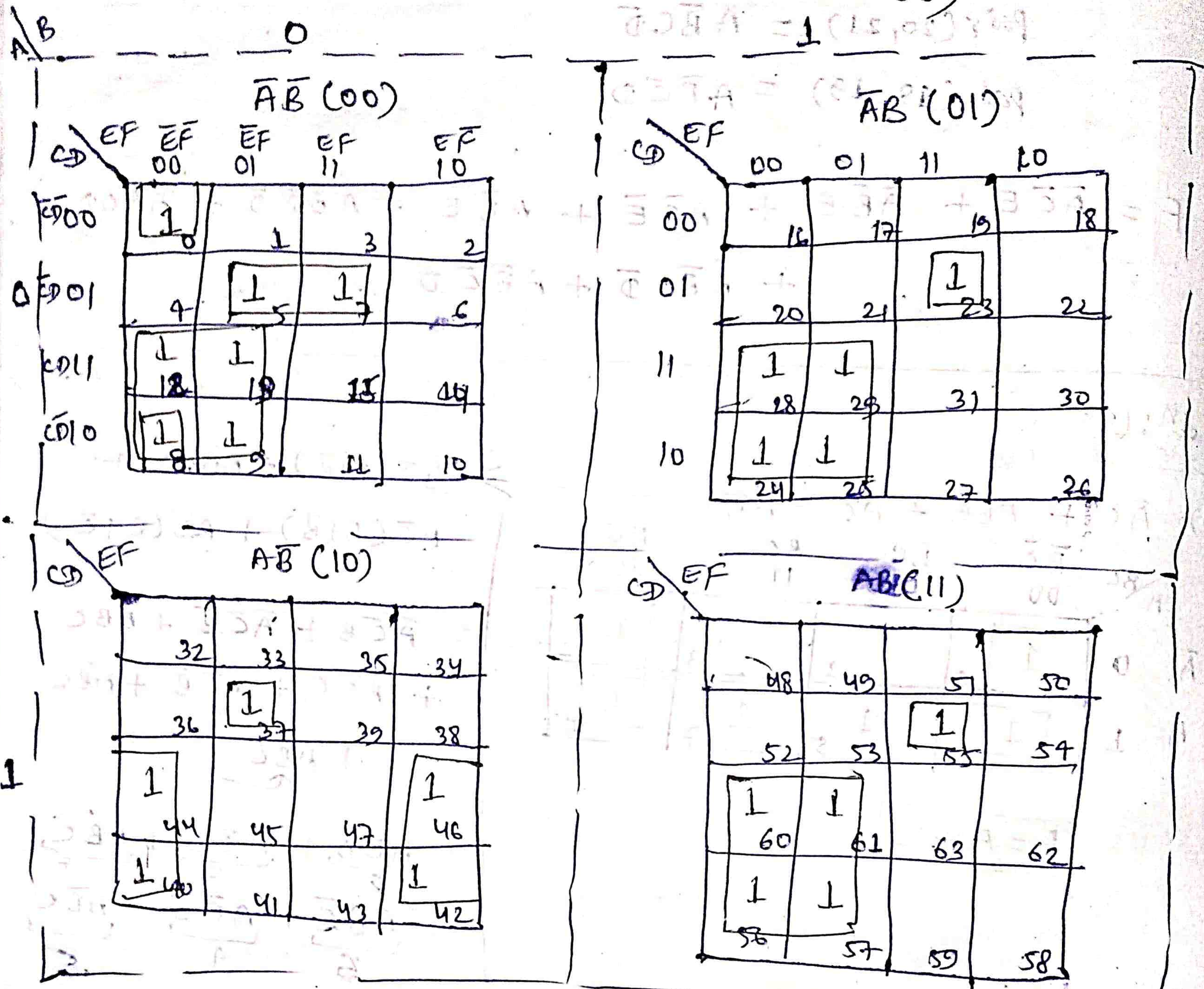
(ii) $(\bar{X}\bar{Y} + Z) + Z + XY + WZ$ ⇒ $\bar{X}\bar{Y} + Z(\overset{1}{\cancel{1+1+W}}) + XY$

$= \bar{X}\bar{Y} + Z + XY$

wx \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{w}\bar{x}$ 00	1 (0)	1 (1)	1 (3)	(2)
$\bar{w}x$ 01	(4)	1 (5)	1 (7)	1 (6)
wx 11	(12)	1 (13)	1 (15)	1 (14)
$w\bar{x}$ 10	1 (8)	1 (9)	1 (11)	(10)

$\boxed{F = Z + \bar{X}\bar{Y} + XY}$

(Q) $F(A,B,C,D,E,F) = \sum m(0,5,7,8,9,12,13,23,24,25,28,29,37,40,42,44,46,55,56,57,60,68)$



1st octet$(8,9,12,13,24,25,28,29) = \bar A C \bar E$

1st pair $(0,8) = \bar A \bar B \bar D \bar E \bar F$

2nd octet$(24,25,28,29,56,57,60,61) = BC\bar E$

1st quad $(40,42,44,46) = A\bar B C \bar F$

2nd pair$(5,7) = \bar A \bar B \bar C D F$

3rd pair$(23,55) = \bar A B \bar C D E F + A B \bar C D E F$

$= B\bar C D E F(\bar A + A) = B \bar C D E F$

4th pair (5,37) $= \overline{A}\overline{B}\,\overline{C}D\overline{E}F + A\overline{B}\,\overline{C}DEF$

$= \overline{B}\,\overline{C}D\overline{E}F(\overline{A}+A)$

$= \overline{B}\,\overline{C}DEF$

$F = \overline{A}C\overline{E} + \overline{A}\,\overline{B}\,\overline{D}\,\overline{E}F + BCE + A\overline{B}CF + \overline{A}B\overline{C}DF + B\overline{C}DEF$
$+ \overline{B}\,\overline{C}DEF$

(Q) $F(A,B,C,D,E,F) = \sum m(6,9,13,18,19,25,27,29,41,45,57,61)$

Soln:

A = 0, B = 0 ($\overline{A}\,\overline{B}$)

CD \ EF	00 ($\overline{E}\,\overline{F}$)	01 ($\overline{E}F$)	11 (EF)	10 ($E\overline{F}$)
00 ($\overline{C}\,\overline{D}$)	0	1	3	2
01 ($\overline{C}D$)	4	5	7	1 (6)
11 (CD)	12	1 (13)	15	14
10 ($C\overline{D}$)	8	1 (9)	11	10

A = 0, B = 1 ($\overline{A}B$)

CD \ EF	00	01	11	10
00	16	17	1 (19)	1 (18)
01	20	21	23	22
11	28	1 (29)	31	30
10	24	1 (25)	1 (27)	26

A = 1, B = 0 ($A\overline{B}$)

CD \ EF	00	01	11	10
00	32	33	35	34
01	36	37	39	38
11	44	1 (45)	47	46
10	40	1 (41)	43	42

A = 1, B = 1 (AB)

CD \ EF	00	01	11	10
00	48	49	51	50
01	52	53	55	54
11	60	1 (61)	63	62
10	56	1 (57)	59	58

1st octet (9, 13, 25, 29, 41, 45, 57, 61) $= C\overline{E}F$

Pair (18, 19) $= \overline{A}B\overline{C}\,\overline{D}E$

Pair (25, 27) $= \overline{A}BC\overline{D}F$

$$\text{Single}(B) = \bar{A}\bar{B}\bar{C}DE\bar{F}$$

$$F = C\bar{E}F + \bar{A}B\bar{C}\bar{D}E + \bar{A}BC\bar{D}F + \bar{A}\bar{B}\bar{C}DE\bar{F}$$

Unit - 2

Combinational logic Circuit

* **Design Procedure :**

(i) Identify the number of inputs and outputs.
(ii) Draw the truth table
(iii) Minize the function using K-map.
(iv) Design the circuit using logic gates.

Question : Design a combinational logic circuit that has four inputs and one output produce 1 when an input is greater than 1000.

Soln : Let the inputs are A, B, C, D and O/P = F

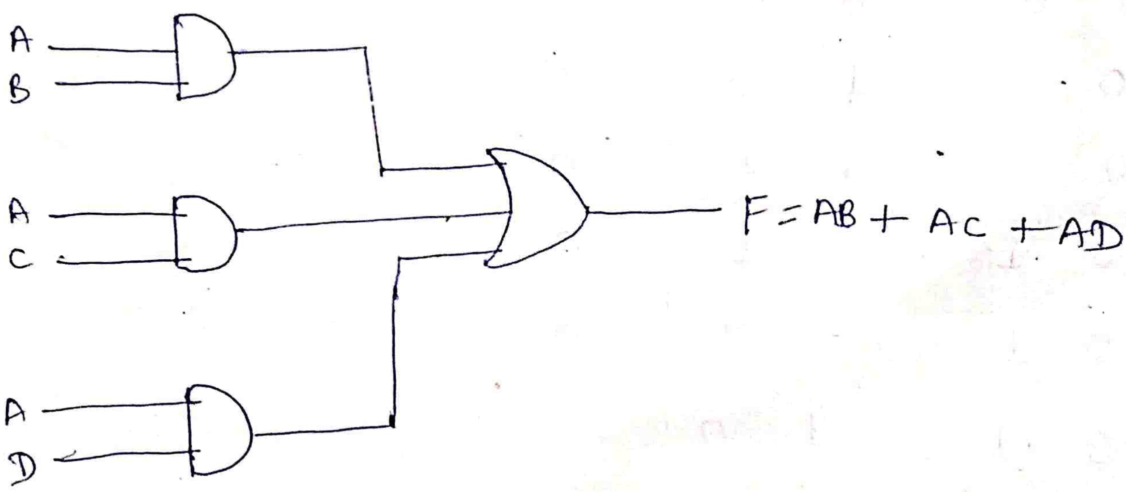
I/P				O/P
A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1

1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

	AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$	00	0	1	3	2
$\bar{A}B$	01	4	5	7	6
AB	11	1 (12)	1 (13)	1 (15)	1 (14)
$A\bar{B}$	10	8	1 (9)	1 (11)	1 (10)

3 quads are formed !

$$F = AB + AC + AD$$



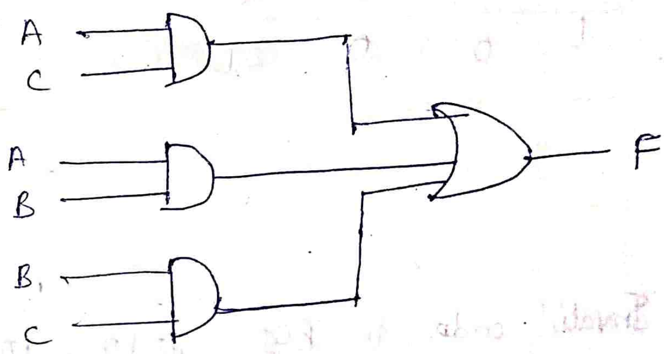
Question : Design a combinational logic ckt. with three input that will produce a logic one output, when more than one input variable is at logic one.

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0			1	
A 1		1	1	1

3 pairs are formed

$F = AC + AB + BC$



---

* Code Converter :

Ques : Design a combinational logic ckt that converts BCD to Excess-3 Code.

Let the inputs are A, B, C, D

and Outputs are W, X, Y, Z

Inputs (BCD)				Output (Excess-3)			
A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

Invalid code in BCD $\underbrace{1010}_{10}$, $\underbrace{1011}_{11}$, $\underbrace{1100}_{12}$, $\underbrace{1101}_{13}$, $\underbrace{1110}_{14}$ and $\underbrace{1111}_{15}$

So, place X in these positions.

For – W

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	(0)	(1)	(3)	(2)
$\bar{A}B$ 01	(4)	1 (5)	1 (7)	1 (6)
AB 11	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$ 10	1 (8)	1 (9)	X (11)	X (10)

1 octet, 2 quad

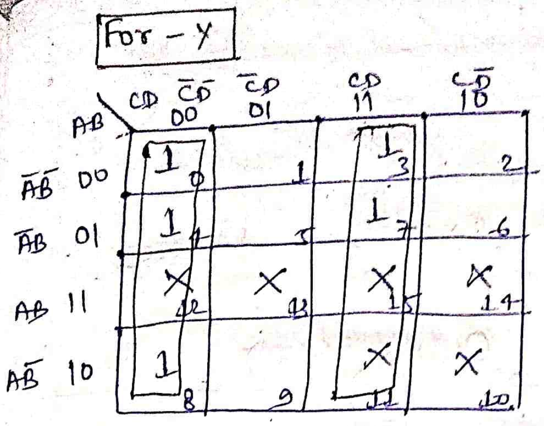
$$W = A + BD + BC$$

For – X

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	(0)	1 (1)	1 (3)	1 (2)
$\bar{A}B$ 01	1 (4)	(5)	(7)	(6)
AB 11	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$ 10	(8)	1 (9)	X (11)	X (10)

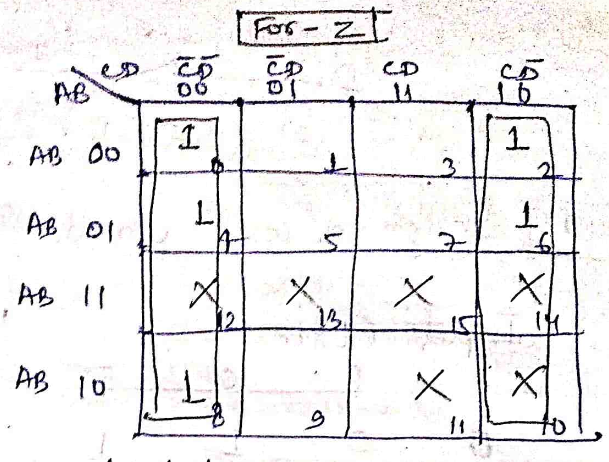
2 quad, 1 pair

$$X = \bar{B}D + \bar{B}C + B\bar{C}\bar{D}$$



2 quad,

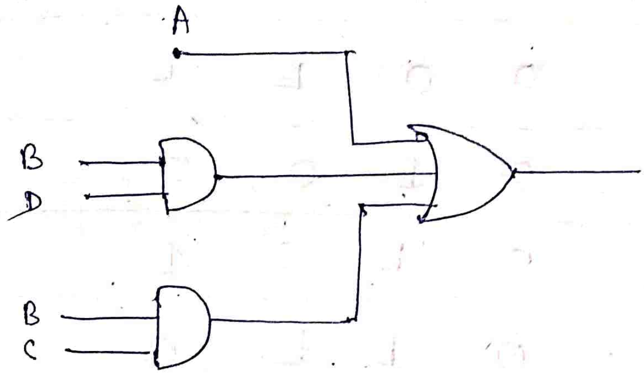
$Y = \bar{C}\bar{D} + CD$



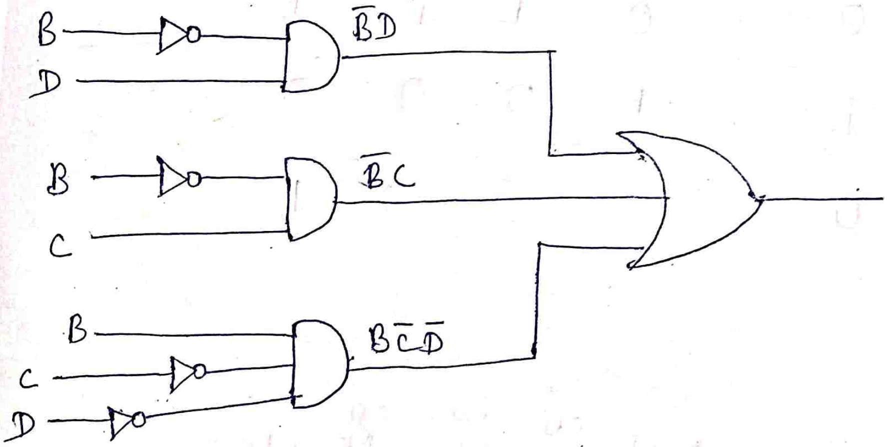
1 octet,

~~$Z = \bar{C}\bar{D} +$~~

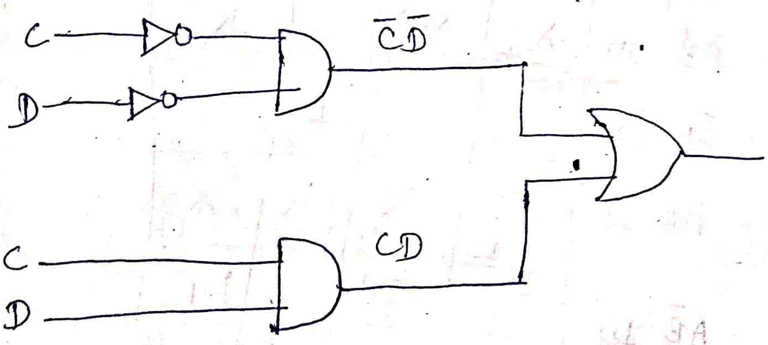
$Z = \bar{D}$



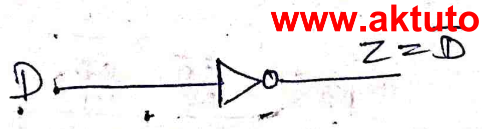
$W = A + BD + BC$



$X = \bar{B}D + \bar{B}C + B\bar{C}\bar{D}$



$Y = \bar{C}\bar{D} + CD$



Ques: Design a combinational logic ckt Excess-3 to BCD.

Inputs (Excess-3) A	B	C	D	Output (BCD) W	X	Y	Z
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1

For - W

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	X 0	X 1	3	X 2
$\bar{A}B$ 01	4	5	7	6
AB 11	1 12	X 13	X 15	X 14
$A\bar{B}$ 10	8	9	1 11	10

1 quad, 1 pair $W = AB + ACD$

For - X

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	X 0	X 1	3	X 2
$\bar{A}B$ 01	4	5	1 7	6
AB 11	12	X 13	X 15	X 14
$A\bar{B}$ 10	1 8	1 9	11	1 10

2 quad, 1 pair $X = \bar{B}\bar{C} + \bar{B}\bar{D} + BCD$

For – Y

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	X 0	X 1	3	X 2
$\bar{A}B$ 01	4	1 5	7	1 6
AB 11	12	X 13	X 15	X 14
$A\bar{B}$ 10	8	1 9	11	1 10

2 quads, $Y = \bar{C}D + C\bar{D}$

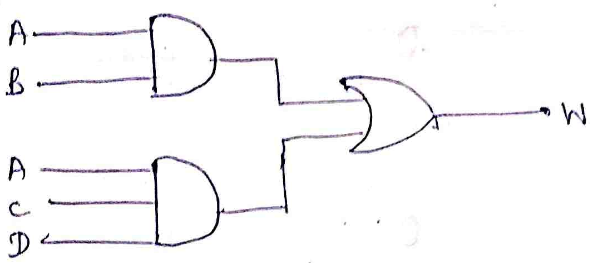
For – Z

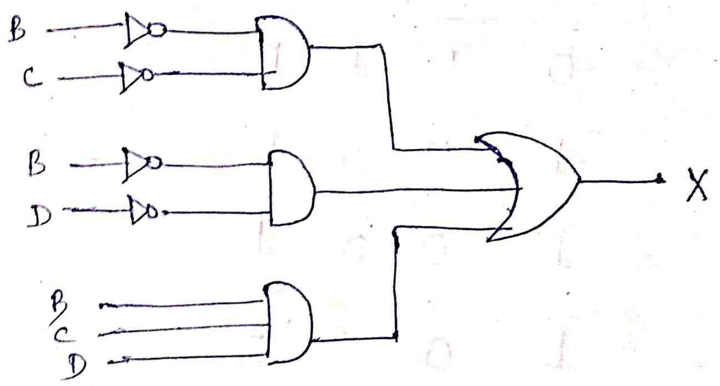
AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	X 0	X 1	3	X 2
$\bar{A}B$ 01	1 4	5	7	1 6
AB 11	1 12	X 13	X 15	X 14
$A\bar{B}$ 10	1 8	9	11	1 10

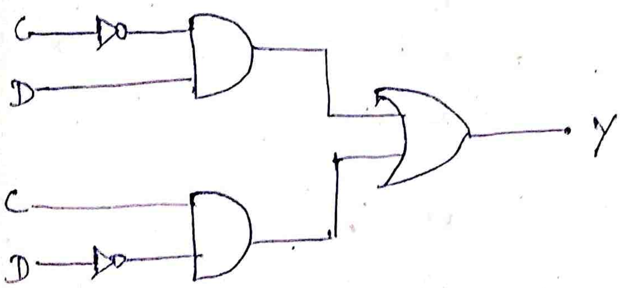
1 octet, $Z = \bar{D}$

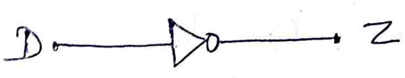
$W = AB + ACD$, $X = \bar{B}\bar{C} + \bar{B}\bar{D} + BCD$

$Y = \bar{C}D + C\bar{D}$, $Z = \bar{D}$









<u>Question – 1</u>: A circuit has 4 inputs and 2 outputs. One of the output is high when majority of input are high. The second output is high when all inputs are of same type.

Sol^n!

I/P				O/P	
A	B	C	D	X	Y
0	0	0	0	0	1
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	0
0	1	0	1	0	0
0	1	1	0	0	0
0	1	1	1	1	0
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	0	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	1	1

For – X :

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0	1	3	2
$\bar{A}B$ 01	4	5	1 (7)	6
AB 11	12	1 (13)	1 (15)	1 (14)
$A\bar{B}$ 10	8	9	1 (11)	10

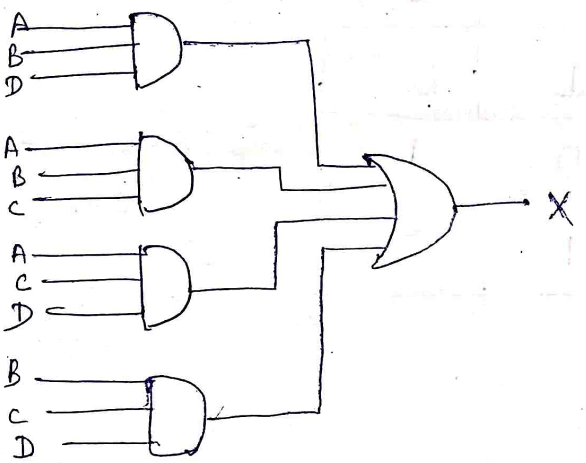
4 pairs,

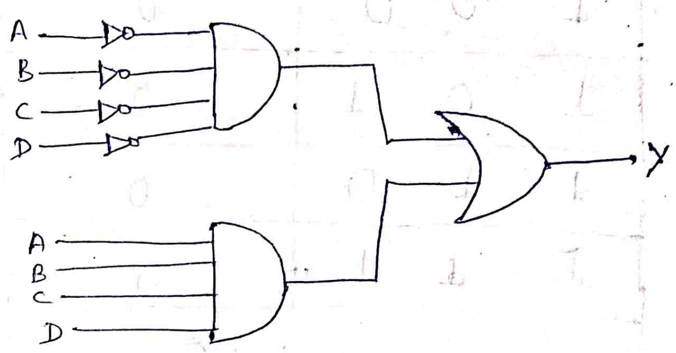
$X = ABD + ABC + ACD$
$+ BCD$

For – Y

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 (0)	1	3	2
$\bar{A}B$ 01	4	5	7	6
AB 11	12	13	1 (15)	14
$A\bar{B}$ 10	8	9	11	10

$Y = \bar{A}\bar{B}\bar{C}\bar{D} + ABCD$





Question – 2 : Design a combinational ckt with 3 inputs and 1 output.

(a) the output is 1 when binary value of the inputs is less than or equal to 3.

(b) the output is 1 when the binary value of the input is an odd number.

(c) the output is 1 when the binary value of the input is an even number.

Soln!

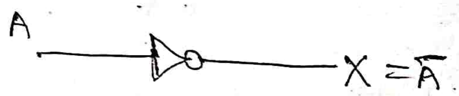
(2)

I/P			O/P	O/P	O/P
A	B	C	X	Y	Z
0	0	0	1	0	1
0	0	1	1	1	0
0	1	0	1	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	0	1	0
1	1	0	0	0	1
1	1	1	0	1	0

(a) For X :-

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	1 (1)	1 (3)	1 (2)
A 1	(4)	(5)	(7)	(6)

$X = \bar{A}$



(b) for Y :-

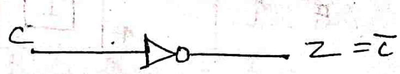
A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	(0)	1 (1)	1 (3)	(2)
A 1	(4)	1 (5)	1 (7)	(6)

$Y = C$

(c) for Z :-

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	1 (0)	(1)	(3)	1 (2)
A 1	1 (4)	(5)	(7)	1 (6)

$Z = \bar{C}$



C → Z = $\bar{C}$

(Q-1) Convert three bit binary to gray code.

Soln:

I/P			O/P		
A	B	C	X	Y	Z
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

$=\bar{A}$

For – X :

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	(0)	(1)	(3)	(2)
A 1	1 (4)	1 (5)	1 (7)	1 (6)

$X = A$

For – Y :

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	(0)	(1)	1 (3)	1 (2)
A 1	1 (4)	1 (5)	(7)	(6)

$Y = A\bar{B} + \bar{A}B$

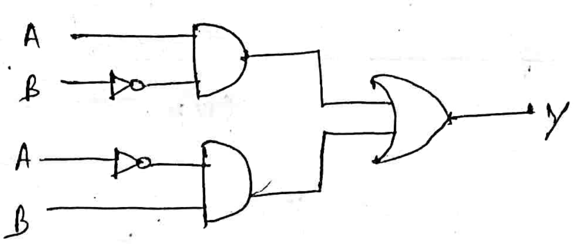
For – Z :

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	0	1 (1)	3	1 (2)
A 1	4	1 (5)	7	1 (6)

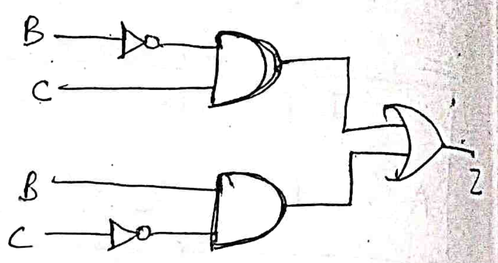
$Z = \bar{B}C + B\bar{C}$

For X :

For Y :



For Z :



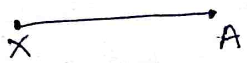
(Q-2) Convert 3-bit gray to binary :

Gray			Binary		
X	Y	Z	A	B	C
0	0	0	0	0	0
0	0	1	0	0	1
0	1	1	0	1	0
0	1	0	0	1	1
1	1	0	1	0	0
1	1	1	1	0	1
1	0	1	1	1	0
1	0	0	1	1	1

For - A :-

X \ YZ	$\bar{Y}\bar{Z}$ 00	$\bar{Y}Z$ 01	YZ 11	$Y\bar{Z}$ 10
$\bar{X}$ 0	$_0$	$_1$	$_3$	$_2$
X 1	$\boxed{1}_4$	$\boxed{1}_5$	$\boxed{1}_7$	$\boxed{1}_6$

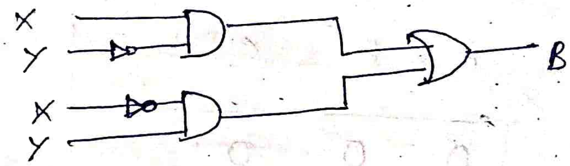
$$A = X$$

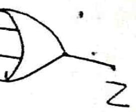


For - B :-

X \ YZ	$\bar{Y}\bar{Z}$ 00	$\bar{Y}Z$ 01	YZ 11	$Y\bar{Z}$ 10
$\bar{X}$ 0	$_0$	$_1$	$\boxed{1}_3$	$\boxed{1}_2$
X 1	$\boxed{1}_4$	$\boxed{1}_5$	$_7$	$_6$

$$B = X\bar{Y} + \bar{X}Y$$

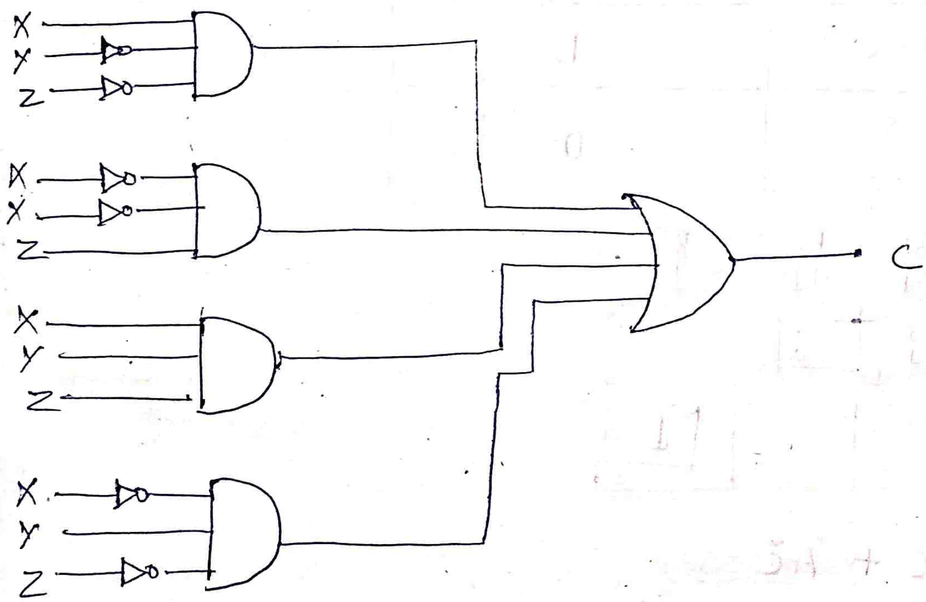




For - C :

X \ YZ	$\bar{Y}\bar{Z}$ 00	$\bar{Y}Z$ 01	YZ 11	$Y\bar{Z}$ 10
$\bar{X}$ 0	$_0$	$\boxed{1}_1$	$_3$	$\boxed{1}_2$
X 1	$\boxed{1}_4$	$_5$	$\boxed{1}_7$	$_6$

$$C = X\bar{Y}\bar{Z} + \bar{X}\bar{Y}Z + XYZ + \bar{X}Y\bar{Z}$$



---

* ~~Adder~~ !

~~Binary Adder~~

Question - 3 : Design binary code to gray code converter.

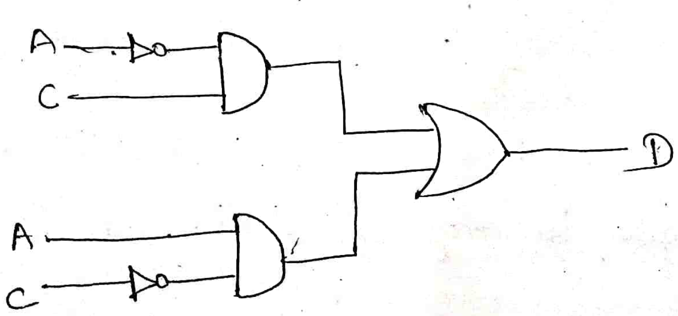
Question-4 Write the steps for combinational circuit designing and design a circuit of three inputs which gives an high output whenever the sum of LSB and MSB bit is 1.

Soln:

I/P			O/P
A	B	C	D
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	0	1 1	1 3	2
A 1	1 4	5	7	1 6

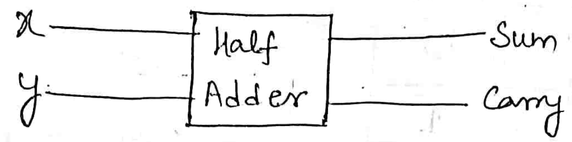
$$D = \bar{A}C + A\bar{C}$$



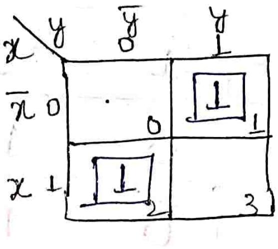
* Adder :

(0) Binary Adder :

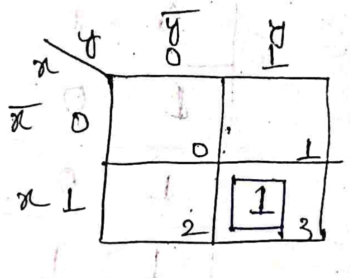
(1) Half Adder : Half adder is a combinational logic circuit with two inputs and ~~the~~ two outputs. The two inputs are augend and addend bits and two binary output sum and carry. An half adder circuit is designed to add two single bit binary numbers.



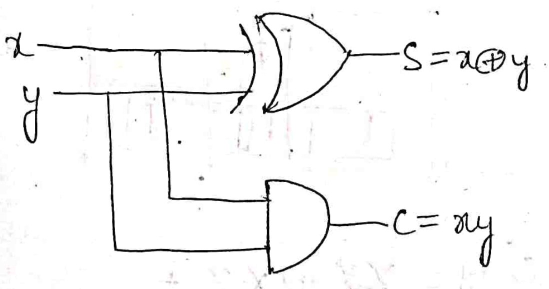
x	y	S (Sum)	C (Carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

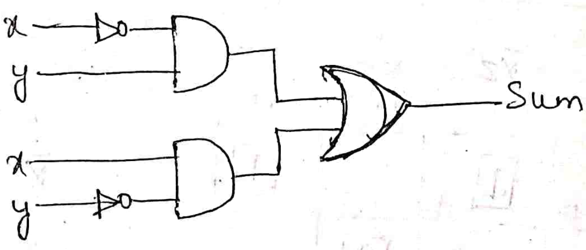


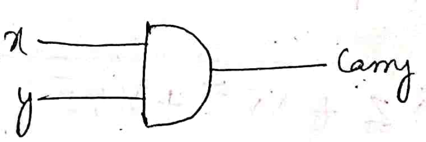
$$\text{Sum} = \bar{x}y + x\bar{y}$$
$$= x \oplus y$$



$$\text{Carry} = xy$$

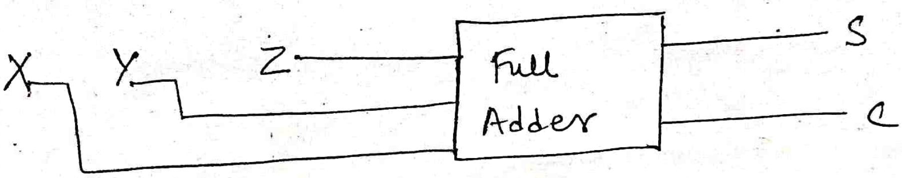






(2) Full Adder : (One bit)

To overcome the drawback of an half adder ckt, we develop a three single bit-adder ckt called full adder. It can add two one bit numbers x and y and carry z represents the carry from the previous lower significant position.



I/p

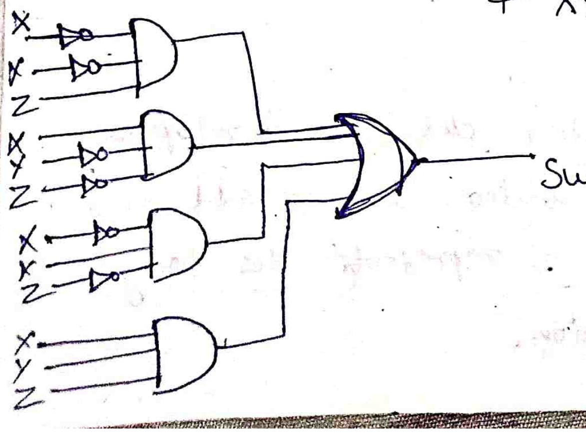
X	Y	Z	S	C
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

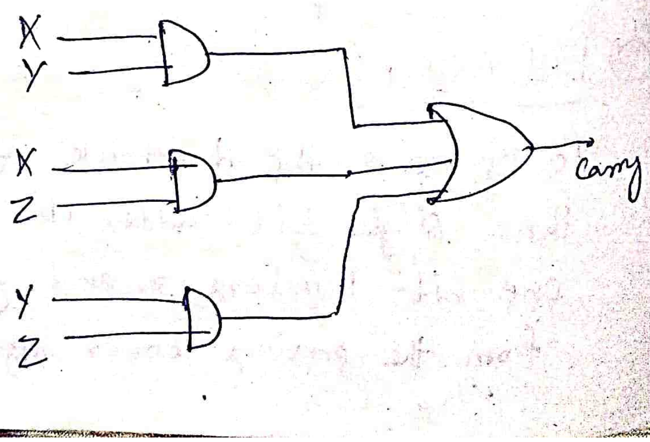
X \ YZ	$\bar{Y}\bar{Z}$ 00	$\bar{Y}Z$ 01	YZ 11	$Y\bar{Z}$ 10
$\bar{X}$ 0	0	$\boxed{1}$ 1	3	$\boxed{1}$ 2
X 1	$\boxed{1}$ 4	5	$\boxed{1}$ 7	6

X \ YZ	$\bar{Y}\bar{Z}$ 00	$\bar{Y}Z$ 01	YZ 11	$Y\bar{Z}$ 10
$\bar{X}$ 0	0	1	$\boxed{1}$ 3	2
X 1	4	$\boxed{1}$ 5	$\boxed{1}$ 7	$\boxed{1}$ 6

$$\text{Sum} = \bar{X}\bar{Y}Z + X\bar{Y}\bar{Z} + \bar{X}Y\bar{Z} + XYZ$$

$$\text{Carry} = XY + XZ + YZ$$





(Q) Design Full adder using half adder :

$$\text{Sum} = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + xyz$$

$$= z(\bar{x}\bar{y} + xy) + \bar{z}(\bar{x}y + x\bar{y})$$

$$= z(\overline{x \oplus y}) + \bar{z}(x \oplus y)$$

$\because (x \odot y = \overline{x \oplus y})$

EXNOR EX-OR

$$= z\bar{X} + \bar{z}X \qquad [\because \text{ let } (x \oplus y) = X]$$

$$= z \oplus X$$

$$\text{Sum} = x \oplus y \oplus z \qquad [\because X = (x \oplus y)]$$

$$\text{Carry} = xy(z+\bar{z}) + xz(y+\bar{y}) + yz(x+\bar{x}) = xyz + xy\bar{z} + \cancel{xyz} + x\bar{y}z + \cancel{xyz} + \bar{x}yz$$

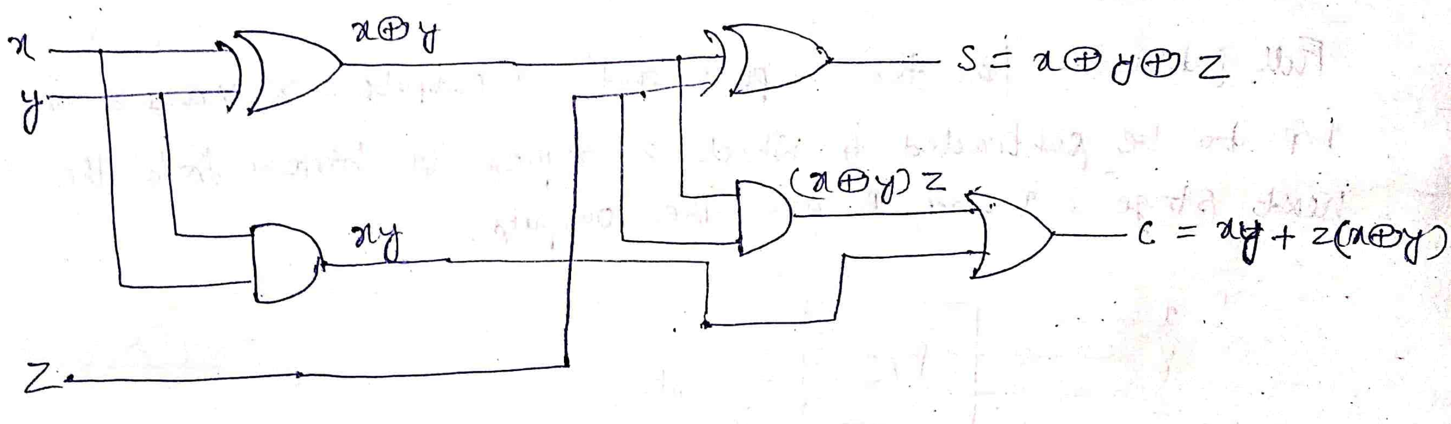
(4) (3) (2) (1)

$$\text{Carry} = \bar{x}yz + x\bar{y}z + xy\bar{z} + xyz$$

$$= z(\bar{x}y + x\bar{y}) + xy(\bar{z} + z)$$

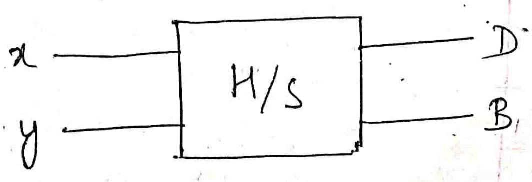
$$= z(x \oplus y) + xy$$

$$\text{Carry} = xy + z(x \oplus y)$$



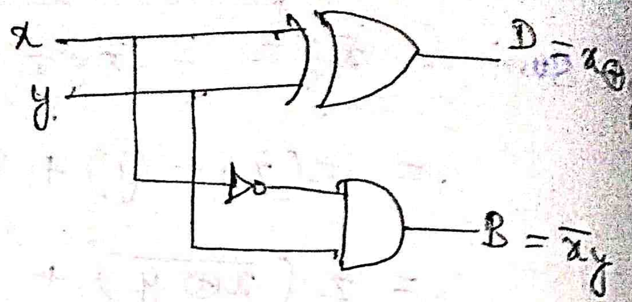
---

* Half Subtractor :



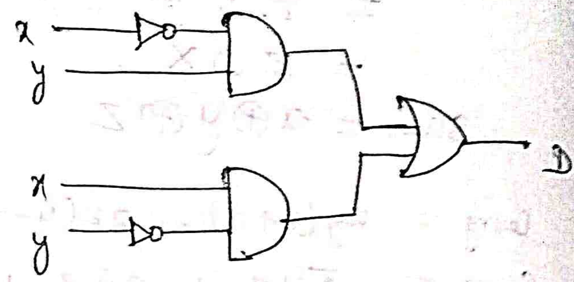
Half subtractor is a combinational logic ckt that subtracts two single bit binary numbers. The half subtractor needs two inputs (minuend and subtrahend) and two ~~inputs borrow~~ outputs (borrow and difference).

I/P		O/P	
x	y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



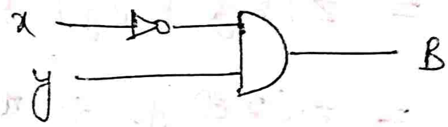
x \ y	$\bar{y}$ 0	y 1
$\bar{x}$ 0	0	$\boxed{1}$ 1
x 1	$\boxed{1}$ 2	3

$D = \bar{x}y + x\bar{y} \Leftarrow$



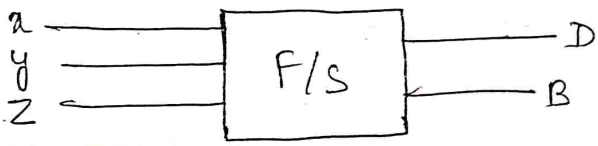
$B = \bar{x}y \Leftarrow$

x \ y	$\bar{y}$ 0	y 1
$\bar{x}$ 0	0	$\boxed{1}$ 1
x 1	2	3



* Full Subractor :

Full subtractor has three inputs and 2 outputs. X, Y and Z are I/p to be subtracted in which Z represents borrow from the next stage. D and B are the outputs.



I/P			O/P	
X	Y	Z	D (x−y−z)	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$D = \sum m(1,2,4,7)$

$D = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + xyz$

$= z(\bar{x}\bar{y} + xy) + \bar{z}(\bar{x}y + x\bar{y})$

$= z(x \odot y) + \bar{z}(x \oplus y)$

$= z(\overline{x \oplus y}) + \bar{z}(x \oplus y)$

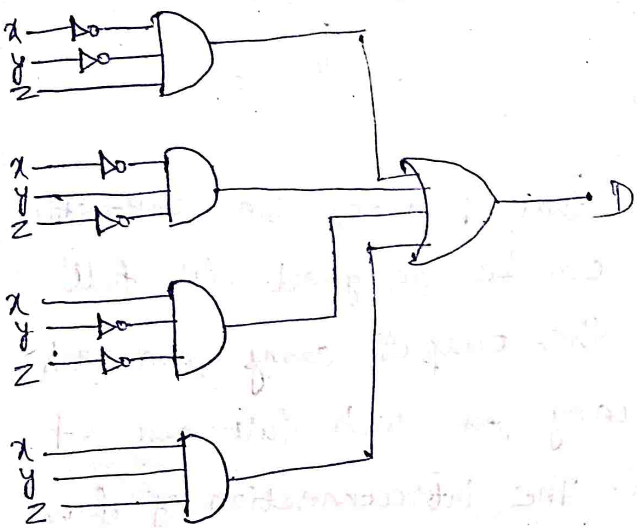
$= x \oplus y \oplus z$

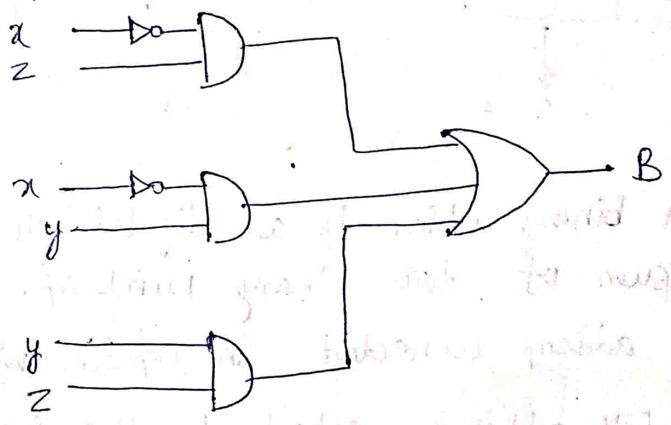
$\boxed{\text{For-B}}$:-

x \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	0	1 (1)	1 (3)	1 (2)
x 1	4	5	1 (7)	6

3 pair:

$B = \bar{x}z + \bar{x}y + yz$





(Q) Design full subtractor using half subtractor.

$D = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + xyz$

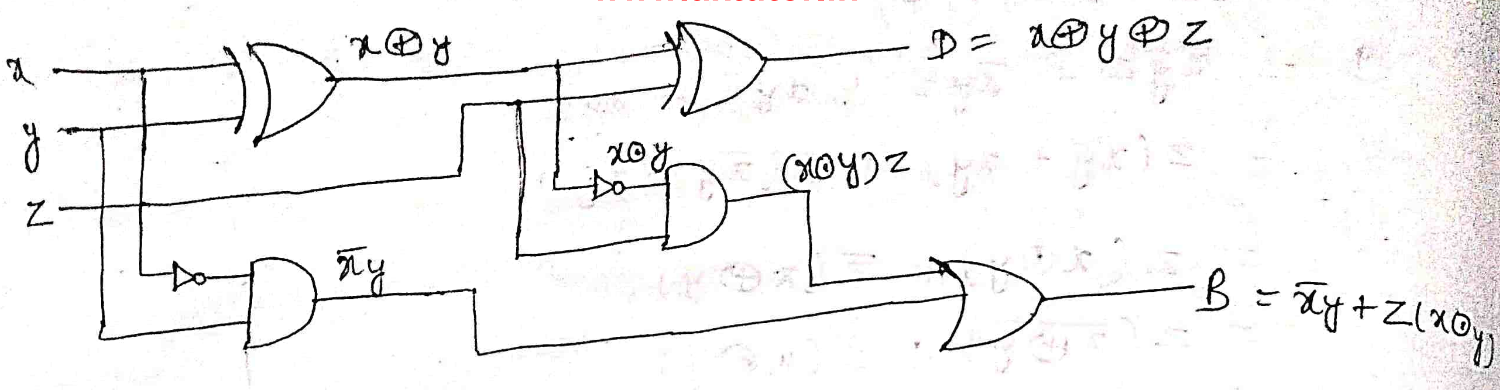
$= x \oplus y \oplus z$

$B = \bar{x}\bar{y}z + \bar{x}y\bar{z} + \bar{x}yz + xyz$

$= \bar{x}\bar{y}z + \bar{x}y(\bar{z}+z) + xyz$

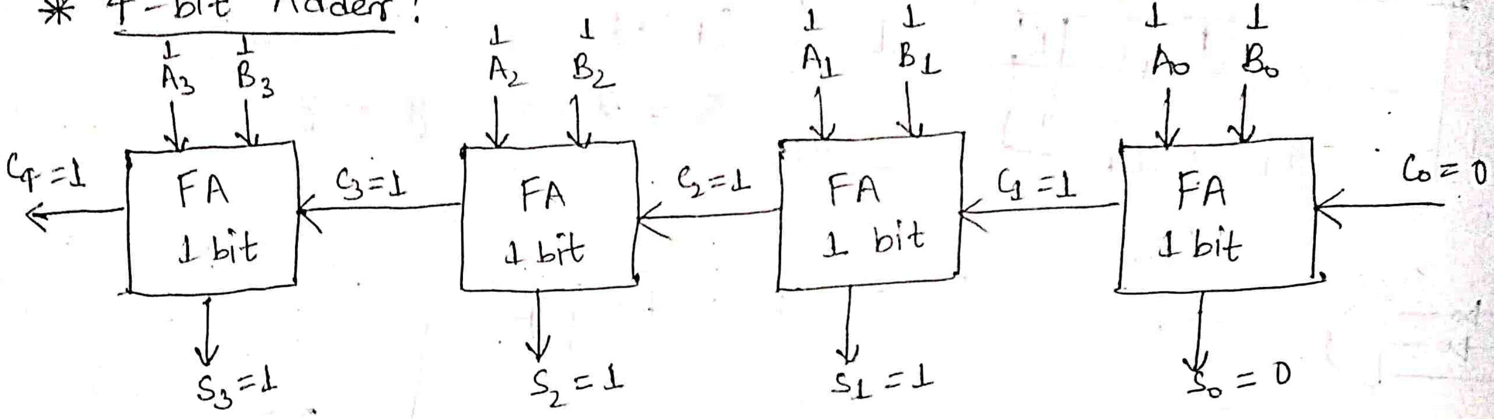
$= z(\bar{x}\bar{y} + xy) + \bar{x}y$

$= \bar{x}y + z(x \odot y)$



⊙ Binary Adder (Parallel Adder) :

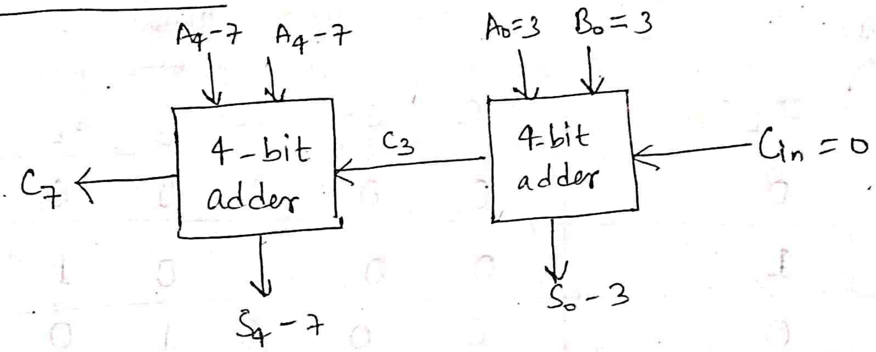
* 4-bit Adder :



A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers. It can be designed with full adders connected in cascade with the output carry from each full adder connected to the input carry ~~from each full add~~ of the next full adder in the chain. The Interconnection of four full adder circuits to provide a four bit binary repel carry adder. The augend bits of A and addend bits of B are designated by subscript numbers from right to left with subscript 0 denoting the LSB. The input carry to the adder is C_0 and it repels through the full adders to the output carry C_4. The S output generate the required sum bits. An n-bit adder requires n full adders with each output carry connected to the input carry of the next higher order full adder. e.g.

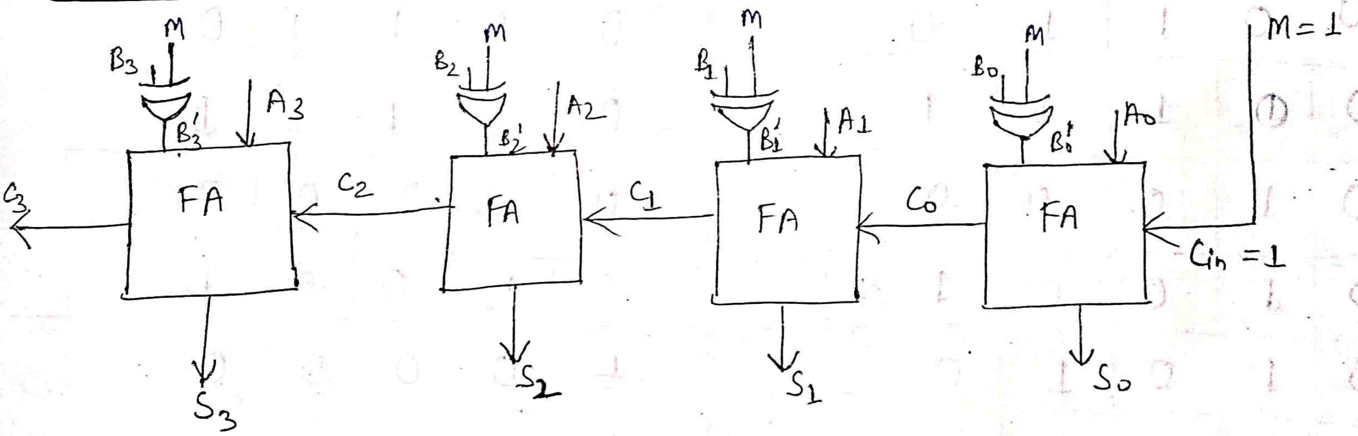
Subscript	3	2	1	0	
I/P carry	0	0	0	0	C_i
Augend	1	1	0	0	A_i
Addend	1	0	1	1	B_i
	0	1	1	1	S_i
	1	0	0	0	C_{i+1}

* 8-bit Adder:



$$B \oplus 0 = B$$
$$B \oplus 1 = B'$$

* 4-Adder/Subtractor Ckt:



A 4-bit adder-subtractor circuit shown in figure. The mode M controls the operation, When $M=0$, the circuit is an adder and when $M=1$, the circuit becomes a subtractor. Each EX-OR receives input M and one of the inputs of B .. When $M=0$, then $B \oplus 0 = B$, The Full adder receives the values of B, the input carry is zero and the ckt perform $A+B$.

When $M=1$, then $B \oplus 1 = \overline{B}$ and input carry $C_{in} = 1$ the B inputs are all complemented and a 1 is added through the input carry. The ckt performs the operation A + 2's complement of B.

Imp * Decimal Adder !

Binary Sum					BCD Sum				
K	Z_8	Z_4	Z_2	Z_1	C	S_8	S_4	S_2	S_1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1
0	0	0	1	0	0	0	0	1	0
0	0	0	1	1	0	0	0	1	1
0	0	1	0	0	0	0	1	0	0
0	0	1	0	1	0	0	1	0	1
0	0	1	1	0	0	0	1	1	0
0	0	1	1	1	0	0	1	1	1
0	1	0	0	0	0	1	0	0	0
0	1	0	0	1	0	1	0	0	1
0	1	0	1	0	1	0	0	0	0
0	1	0	1	1	1	0	0	0	1
0	1	1	0	0	1	0	0	1	0
0	1	1	0	1	1	0	0	1	1
0	1	1	1	0	1	0	1	0	0
0	1	1	1	1	1	0	1	0	1
1	0	0	0	0	1	0	1	1	0

1	0	0	0	1	1	0	1	1	1
1	0	0	1	0	1	1	0	0	0
1	0	0	1	1	1	1	0	0	1

$\bar{K} = 0$

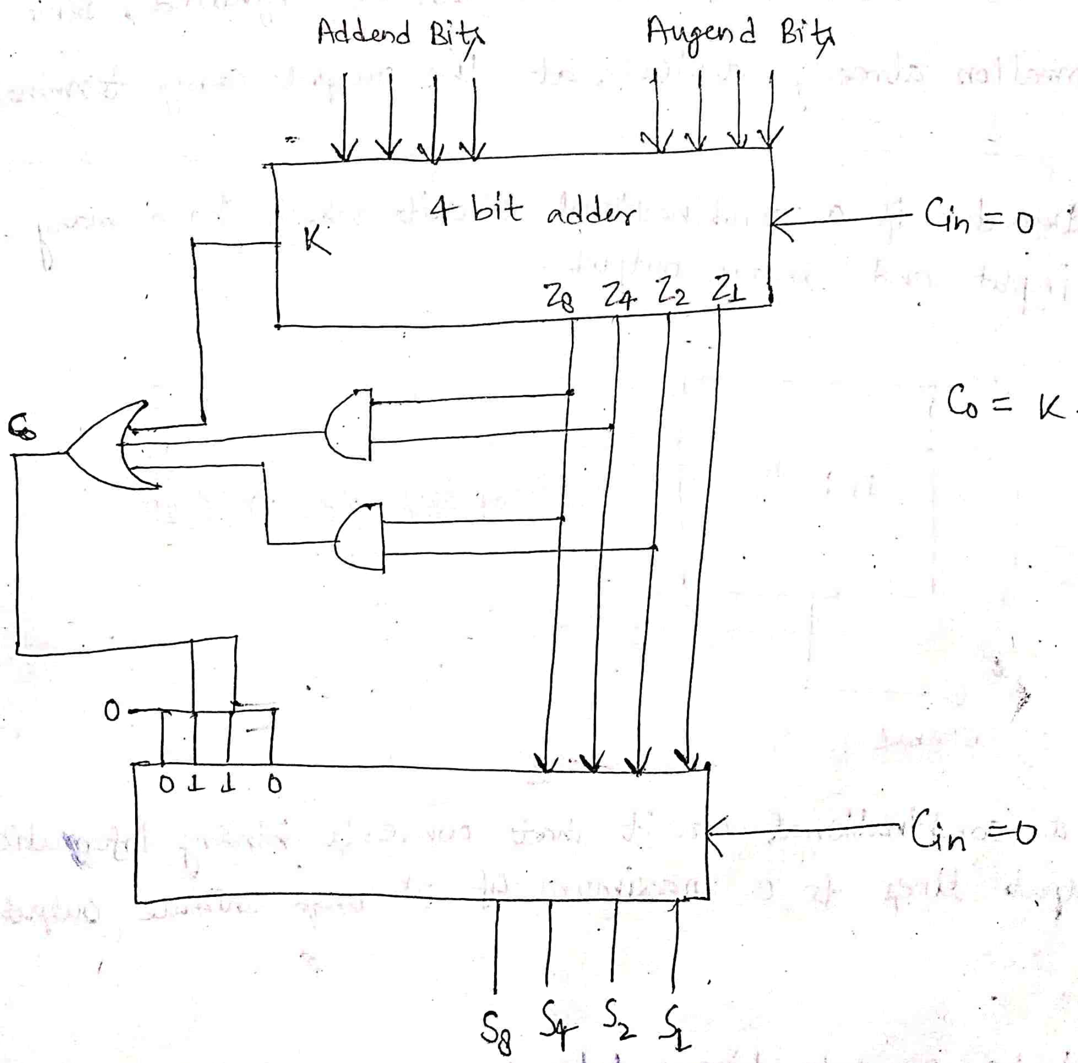
$Z_8Z_4 \backslash Z_2Z_1$				
	0	1	3	2
	4	5	7	6
	1 (12)	1 (13)	1 (15)	1 (14)
	8	9	1 (11)	1 (10)

$K = 1$

$Z_8Z_4 \backslash Z_2Z_1$				
	1 (16)	1 (17)	1 (18)	1 (19)
	X (20)	X (21)	X (23)	X (22)
	X (28)	X (29)	X (31)	X (30)
	X (24)	X (25)	X (27)	X (26)

~~2 octet and 1~~

$\therefore C = K + Z_8Z_4 + Z_8Z_2$



$C_0 = K + Z_8Z_4 + Z_8Z_2$

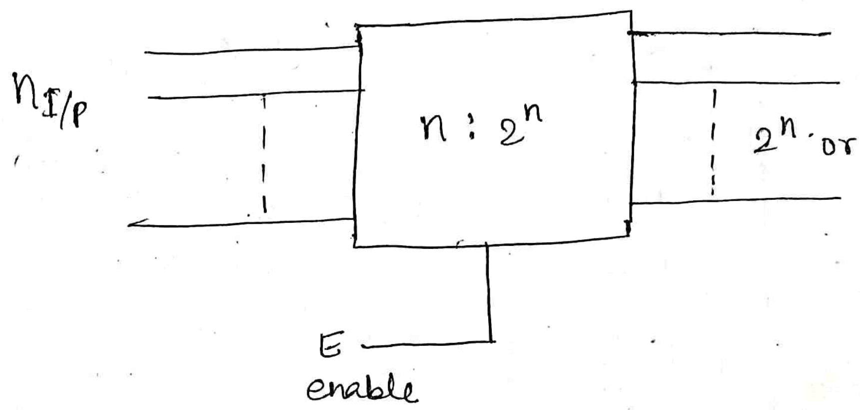
The condition for a correction and an output carry can be expressed by the boolean function:

$$C_o = K + Z_8 Z_4 + Z_8 Z_2$$

A BCD adder that adds two BCD digits and produces a sum digit in BCD is shown in figure. The two decimal digits together with the input carry are first added in the top four bit adder to produce the binary sum. When the output carry is equal to zero nothing is added to the binary sum. When it is equal to one. Binary 0110 is added to the binary sum through the bottom four bit adder. The output carry generates from the bottom adder can be ignored, since supply information already available at the output carry terminal.

---

* Decoder: Decoder is a combinational circuit which have many input and many output.



- Decoder is a combinational circuit that converts binary information from n-inputs lines to a maximum of 2^n ~~unqs~~ unique output lines.

- It is used to convert binary data to other code. e.g. Binary to octal, BCD to decimal, BCD to seven segment display, Binary to hexadecimal.

- If the n-bit coded information has unused combinations the decoder may have fewer than 2^n output.
- Some decoders are designed to produce active low output, while all the other outputs remain high.
- Decoder generate 2^n (or less) minterms of n input variables.
- Decoder include one or more enable inputs to control the circuit operation.

2 to 4 line decoder:

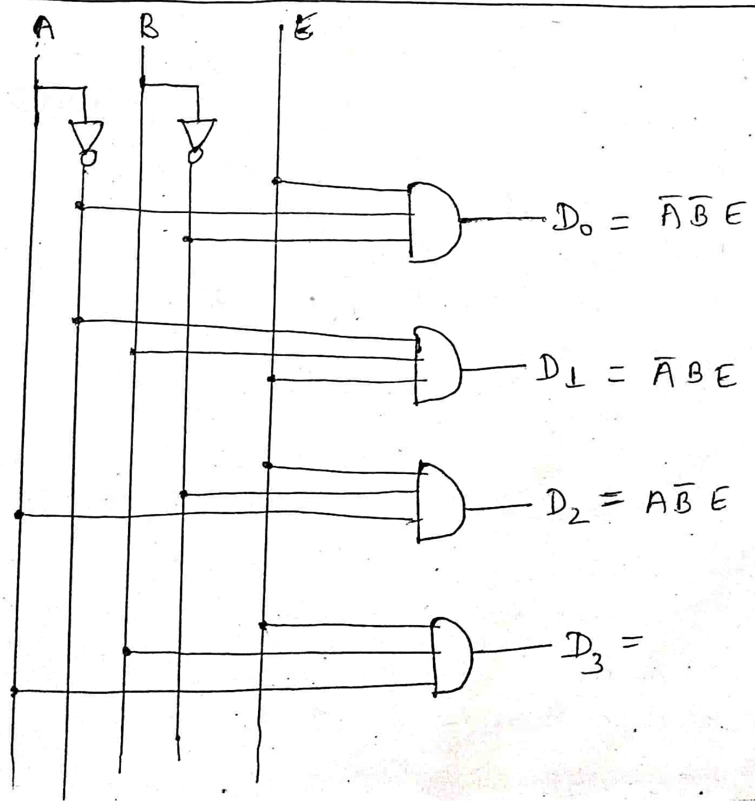
E	A	B	D_0	D_1	D_2	D_3
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

$D_0 = \bar{A}\bar{B}E$

$D_1 = \bar{A}BE$

$D_2 = A\bar{B}E$

$D_3 = ABE$



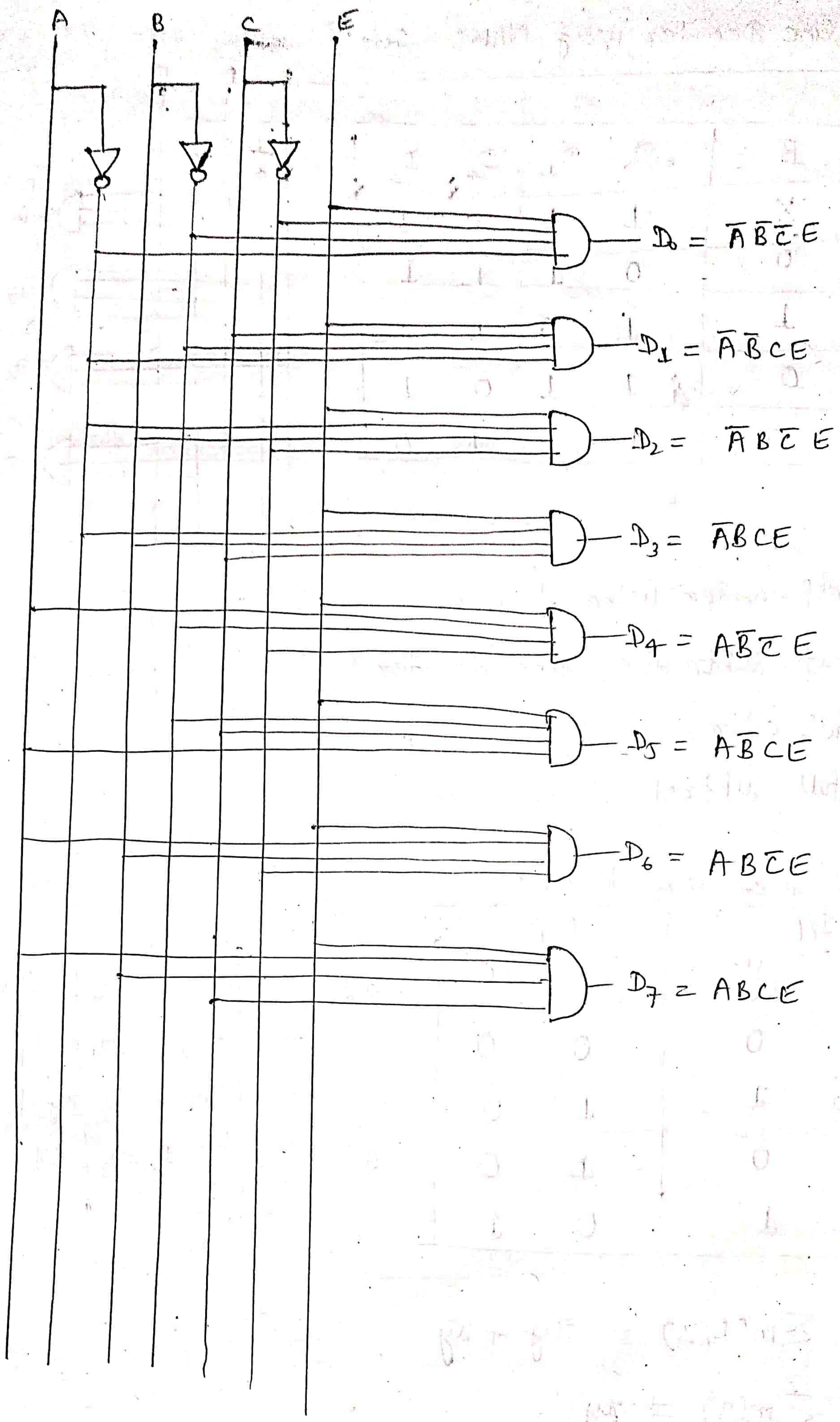
* 3 to 8 line Decoder :

E	A	B	C	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	X	X	X	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
1	0	1	0	0	0	1	0	0	0	0	0
1	0	1	1	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	0	0	1	0	0
1	1	1	0	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	0	1

$D_0 = \bar{A}\bar{B}\bar{C}E$ $D_1 = E\bar{A}\bar{B}C$ $D_2 = E\bar{A}B\bar{C}$, $D_3\ E\bar{A}BC$

$D_4 = EA\bar{B}\bar{C}$ $D_5 = EA\bar{B}C$ $D_6 = EAB\bar{C}$.

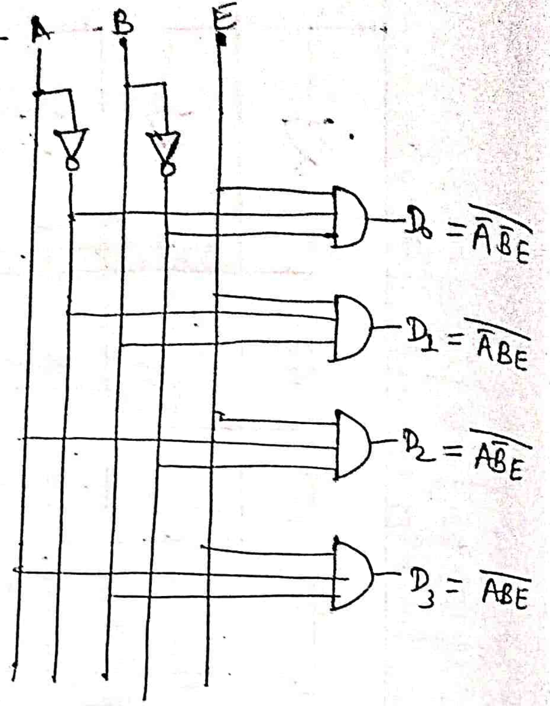
$D_7 = EABC$



A
B
C
E
$D_0 = \bar{A}\bar{B}\bar{C}E$
$D_1 = \bar{A}\bar{B}CE$
$D_2 = \bar{A}B\bar{C}E$
$D_3 = \bar{A}BCE$
$D_4 = A\bar{B}\bar{C}E$
$D_5 = A\bar{B}CE$
$D_6 = AB\bar{C}E$
$D_7 = ABCE$

* 2 to 4 line Decoder using NAND - gate :

E	A	B	D_0	D_1	D_2	D_3
0	X	X	1	1	1	1
1	0	0	0	1	1	1
1	0	1	1	0	1	1
1	1	0	1	1	0	1
1	1	0	1	1	1	0

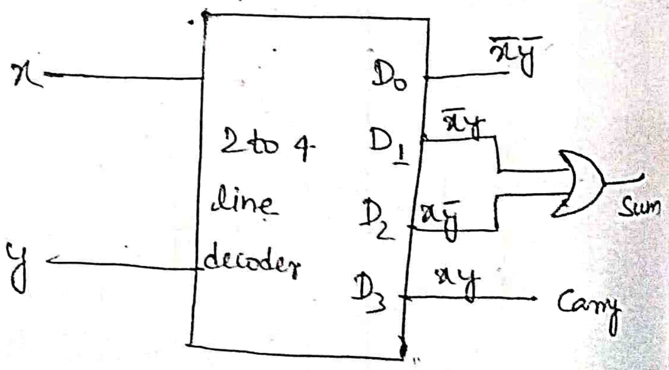


Questions!

1. Design half-adder using decoder.
2. Design half-subtractor using decoder.
3. Design full adder, " " .
4. Design full subtractor, " " .

Soln ①: Half adder using decoder :

I/P		O/P	
x	y	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



$$Sum = \Sigma m(1,2) = \bar{x}y + x\bar{y}$$

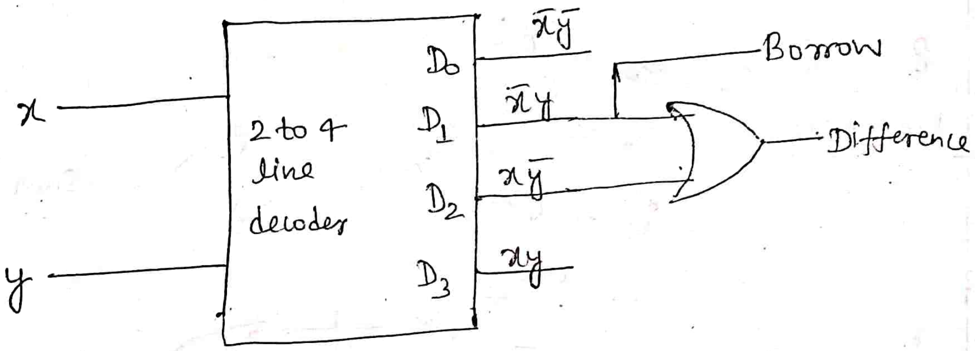
$$Carry = \Sigma m(3) = xy$$

Sol^n ②: Half subtractor using decoder :

I/P		O/P	
x	y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Difference $= \sum m(1,2) = \bar{x}y + x\bar{y}$

Borrow $= \sum m(1) = \bar{x}y$

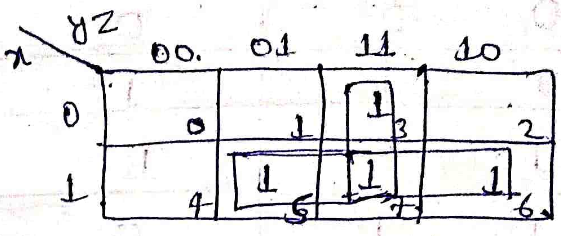


Sol^n ③: Full adder using decoder !

I/P			O/P	
x	y	z	S	C
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\text{Sum} = \sum m(1, 2, 4, 7)$$

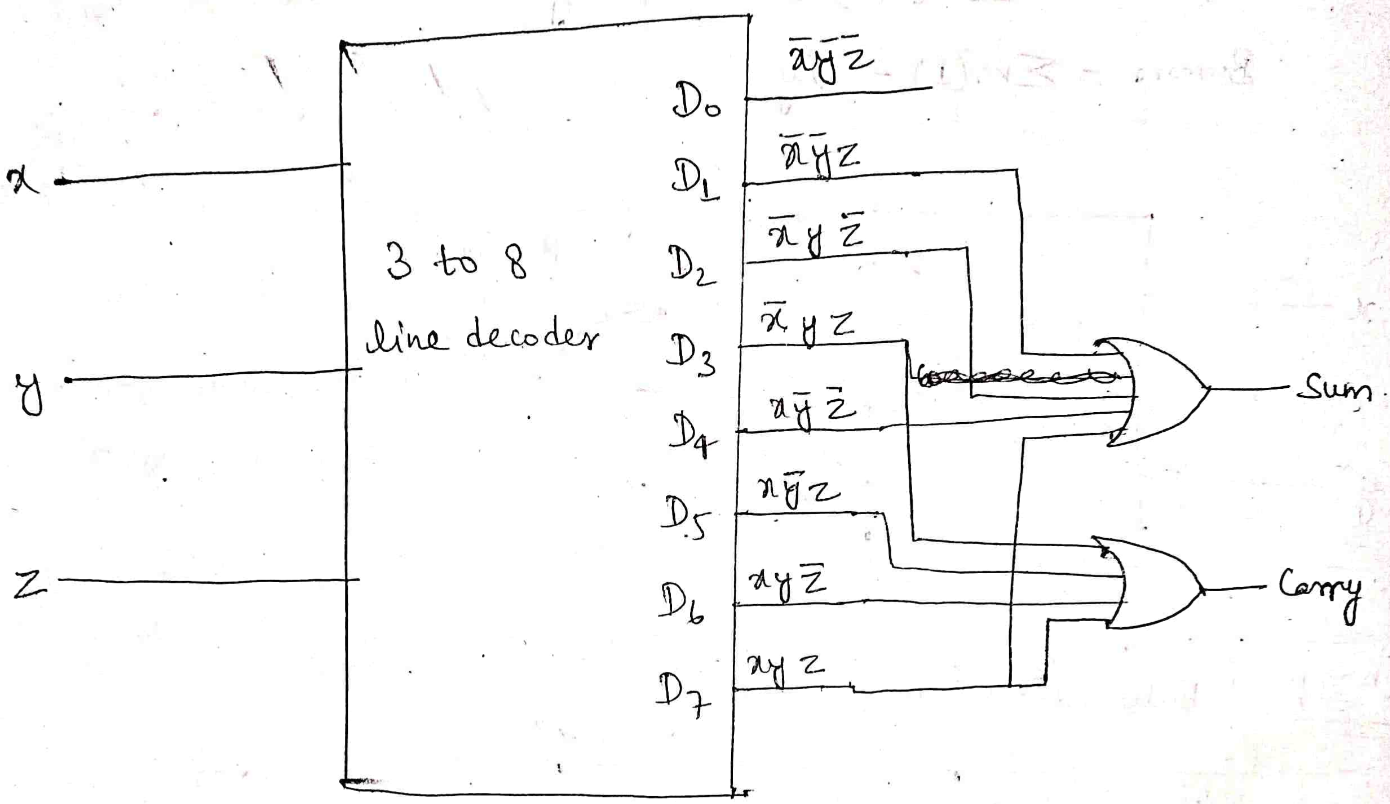
$$\text{Sum} = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + xyz$$



$$\text{Carry} = \sum(3, 5, 6, 7)$$

$$\text{Carry} = \bar{x}yz + x\bar{y}z + xy\bar{z} + xyz$$

$$\text{Carry} = xy + yz + zx$$



Soln ⑧ ! Full subtractor using decoder !

I/P			O/P	
x	y	z	D	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$D = \Sigma m(1,2,4,5)$

$D = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + x\bar{y}z$

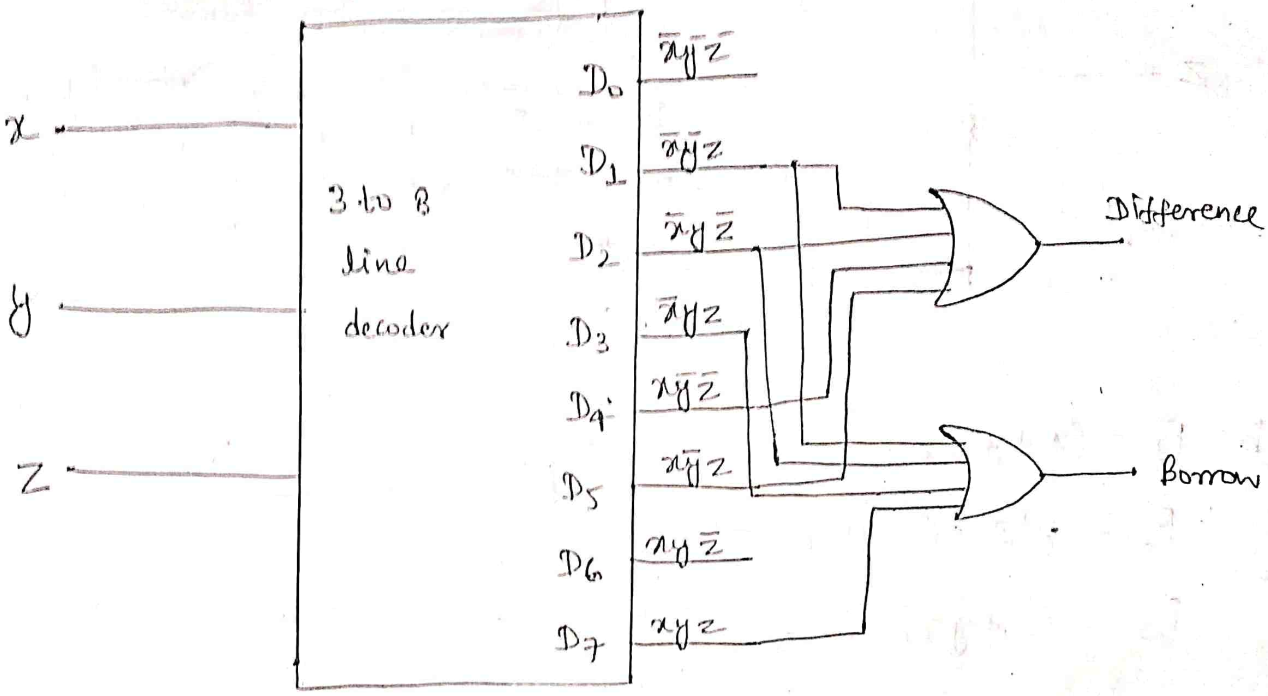
$B = \Sigma m(1,2,3,7)$

$B = \bar{x}\bar{y}z + \bar{x}y\bar{z} + \bar{x}yz + xyz$

$B = \bar{x}z + \bar{x}y + yz$

$x \backslash yz$	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	0	1 (1)	1 (3)	1 (2)
x 1	4	5	1 (7)	6

$B = \bar{x}z + \bar{x}y + yz$



Question: Using a decoder and external gates design the combinational circuit. Defined by the following three Boolean function.

(a) $F_1 = \bar{x}y\bar{z} + xz$

$F_2 = xy'z' + x'y$

$F_3 = x'y'z' + xy$

Soln:

$F_1 = \bar{x}y\bar{z} + xz(y+\bar{y})$

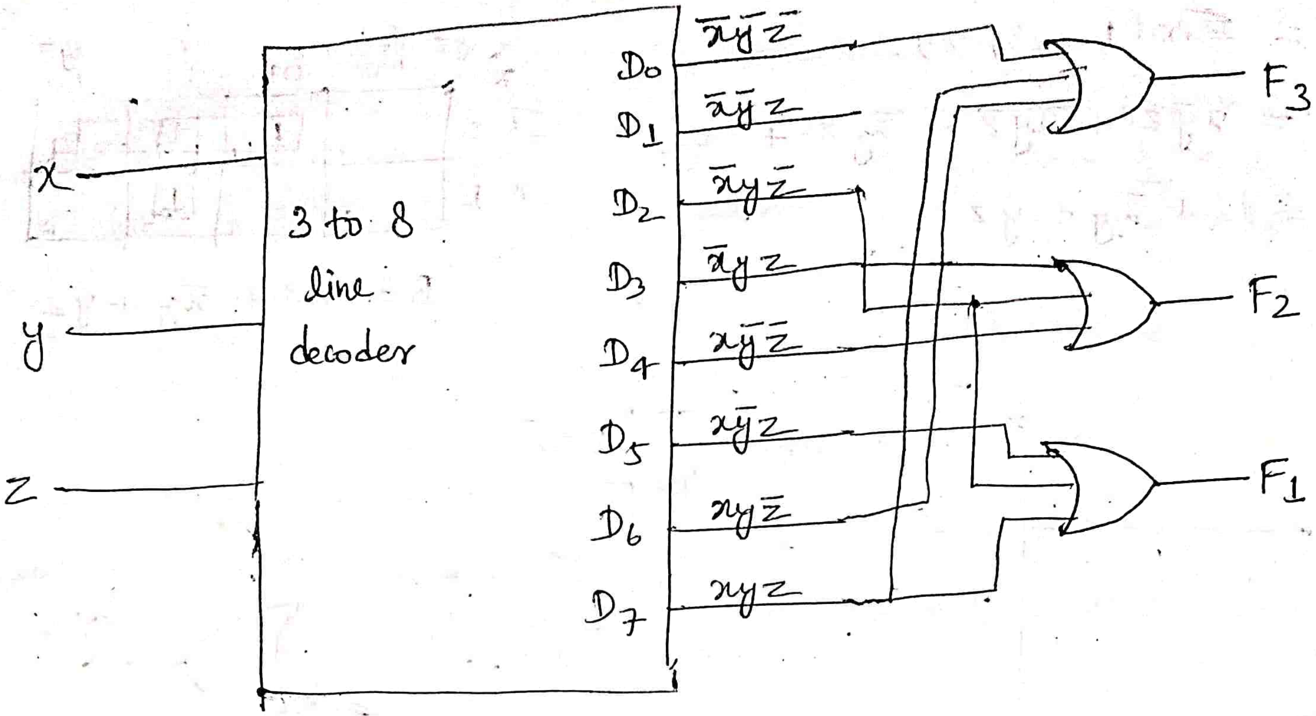
$= \underbrace{\bar{x}y\bar{z}}_{2} + \underbrace{xyz}_{7} + \underbrace{x\bar{y}z}_{5}$

$F_2 = x\bar{y}\bar{z} + \bar{x}y(z+\bar{z})$

$= x\bar{y}\bar{z}_{(4)} + \bar{x}yz_{(3)} + xy\bar{z}_{(6)}$

$F_3 = \bar{x}\bar{y}\bar{z} + xy(z+\bar{z})$

$= \underbrace{\bar{x}\bar{y}\bar{z}}_{0} + \underbrace{xyz}_{7} + \underbrace{xy\bar{z}}_{6}$



(b) $F_1 = (x+y')z$

$F_2 = y'z' + xy' + xz'$

$F_3 = (x'+y)z$

Soln:

$F_1 = xz + \bar{y}z$

$= xz(y+\bar{y}) + \bar{y}z(x+\bar{x})$

$= xyz + \underbrace{x\bar{y}z + x\bar{y}z}_{\text{merging}} + \bar{x}\bar{y}z$

$= xyz + x\bar{y}z + \bar{x}\bar{y}z \Rightarrow F_1 = \Sigma m(7,5,1)$

$F_2 = \bar{y}\bar{z}(x+\bar{x}) + x\bar{y}(z+\bar{z}) + x\bar{z}(y+\bar{y})$

$= \underbrace{x\bar{y}\bar{z} + \bar{x}\bar{y}\bar{z} + x\bar{y}z + x\bar{y}\bar{z} + xy\bar{z} + x\bar{y}\bar{z}}$

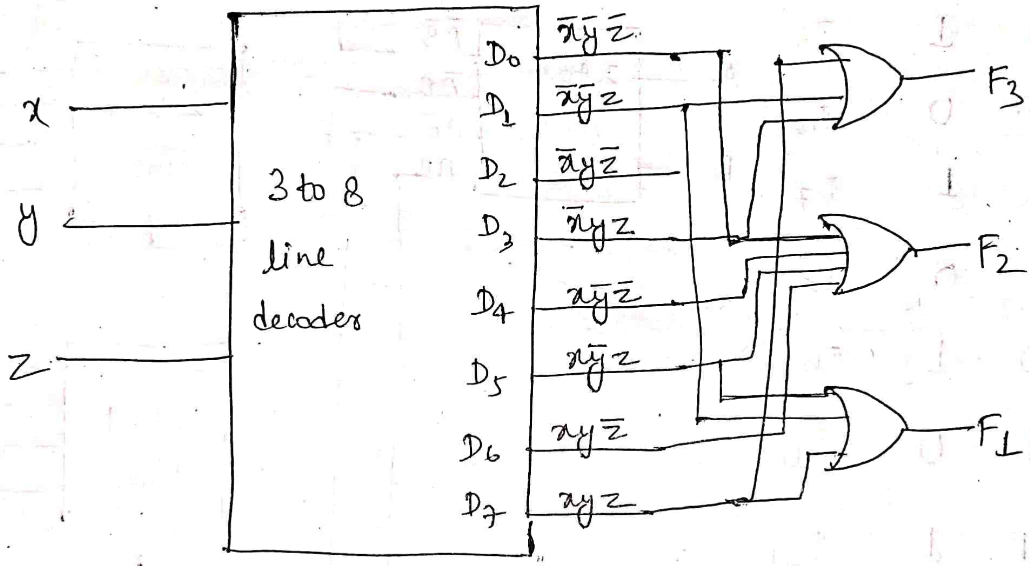
$= x\bar{y}\bar{z} + \bar{x}\bar{y}\bar{z} + x\bar{y}z + xy\bar{z} \Rightarrow F_2 = \Sigma m(4,0,5,6)$

$F_3 = \bar{x}z + yz$

$= \bar{x}z(y+\bar{y}) + yz(x+\bar{x})$

$= \bar{x}yz + \bar{x}\bar{y}z + xyz + \bar{x}yz$

$= \bar{x}yz + \bar{x}\bar{y}z + xyz \Rightarrow F_3 = \Sigma m(3,1,7)$



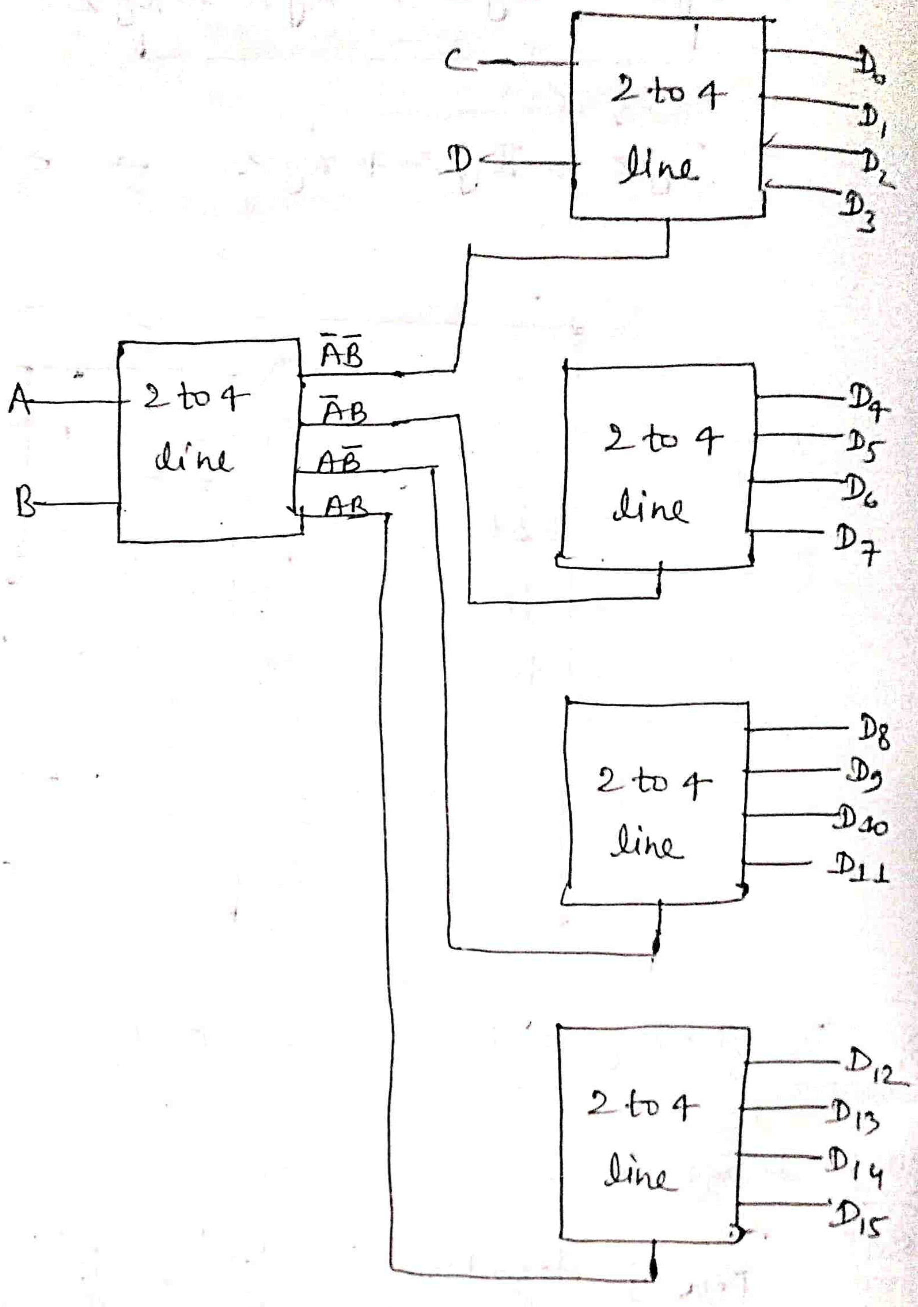
* Design Higher Order Decoder using lower order!

Ques! Design 4 to 16 line decoder using 2-4 line decoder.

Sol^n! No. of decoder $= \frac{\cancel{16}^{4}}{\cancel{4}} + \frac{\cancel{4}^{1}}{\cancel{4}} = 5$ decoder.

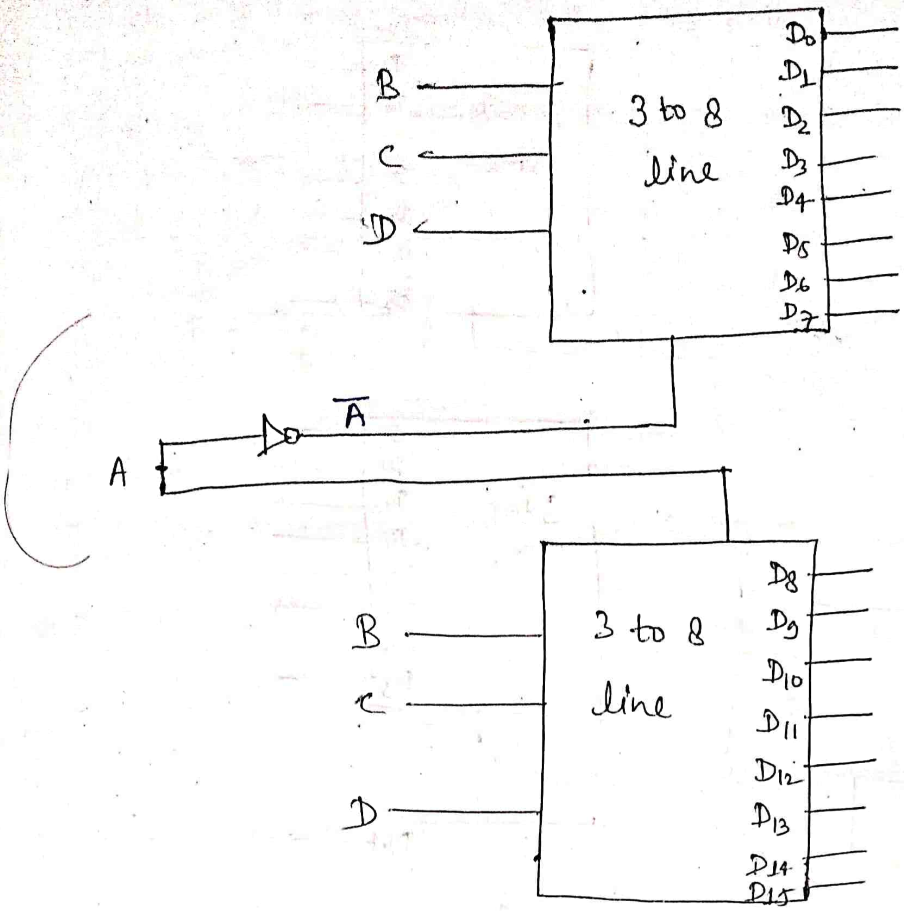
P.T.O

A	B	C	D	
0	0	0	0	D_0
0	0	0	1	D_1
0	0	1	0	D_2
0	0	1	1	D_3
0	1	0	0	D_4
0	1	0	1	D_5
0	1	1	0	D_6
0	1	1	1	D_7
1	0	0	0	D_8
1	0	0	1	D_9
1	0	1	0	D_{10}
1	0	1	1	D_{11}
1	1	0	0	D_{12}
1	1	0	1	D_{13}
1	1	1	0	D_{14}
1	1	1	1	D_{15}



$\boxed{\text{Ques}}$: Design 4 - 16 line decoder using 3-8 line.

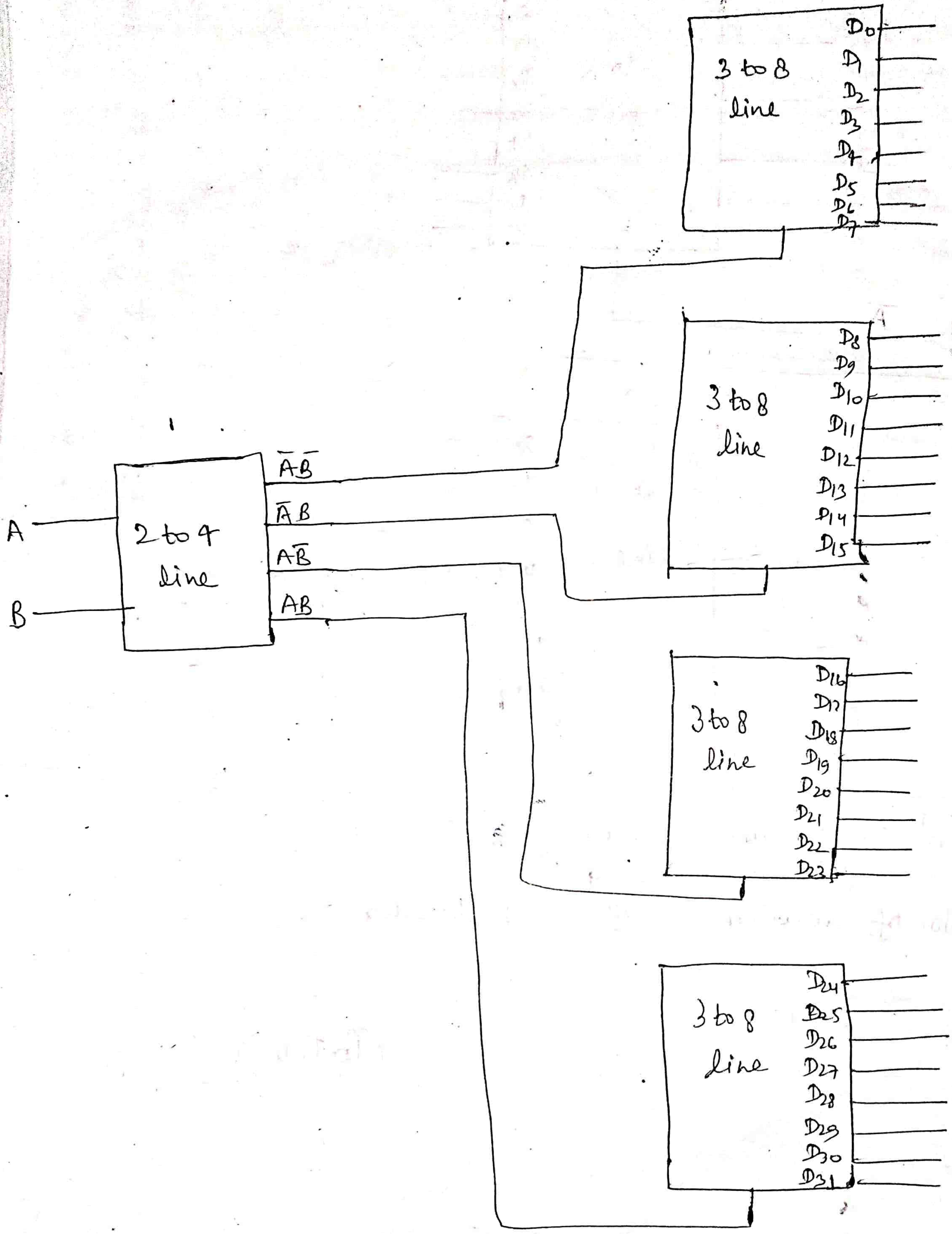
<u>Soln</u>: No. of decoder $= \frac{\cancel{16}^{2}}{\cancel{8}}$ = 2 decoder.



$\boxed{\text{Ques}}$! Design 5-32 line decoder using 3-8 line.

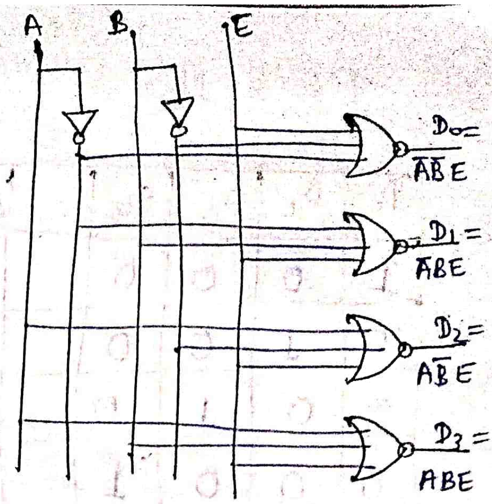
Solⁿ! No. of decoder $= \dfrac{\cancel{32}^{4}}{\cancel{8}} = 4$ decoder

(P.T.O)



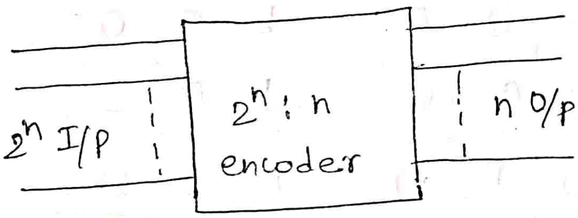
Ques! Design 2-4 line decoder using NOR-Gate.

A	B	E	D_0	D_1	D_2	D_3
0	0	1	1	0	0	0
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	1



* Encoder :

- An encoder is a digital circuit that performs the reverse operation of a decoder.
- Encoder is a combinational circuit which have ~~may~~ many input and many output.
- Encoder is used to convert other code to binary.
  e.g. Octal to Binary, Hexadecimal to Binary, Decimal to BCD.

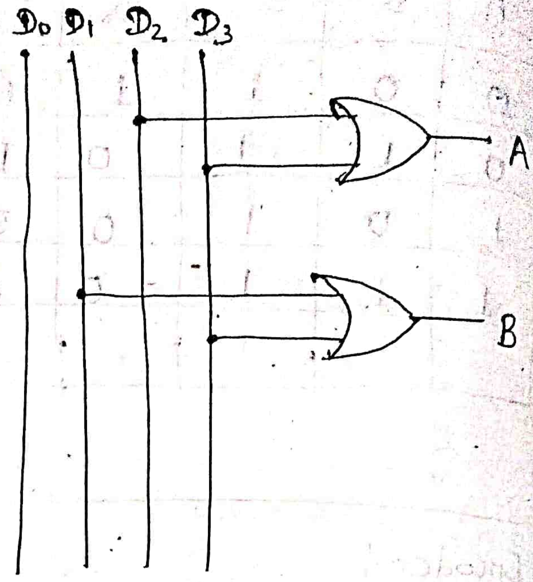


- An encoder has 2^n (or fewer) input lines and n output lines.
- In normal encoder one of the input line is high and corresponding binary available at the O/P.

$\boxed{\text{Note}}$: In priority encoder, number of I/P is high only highest priority number corresponding binary is available at the output.

4 to 2 line

D_0	D_1	D_2	D_3	A	B
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1



$A = D_2 + D_3$, $B = D_1 + D_3$

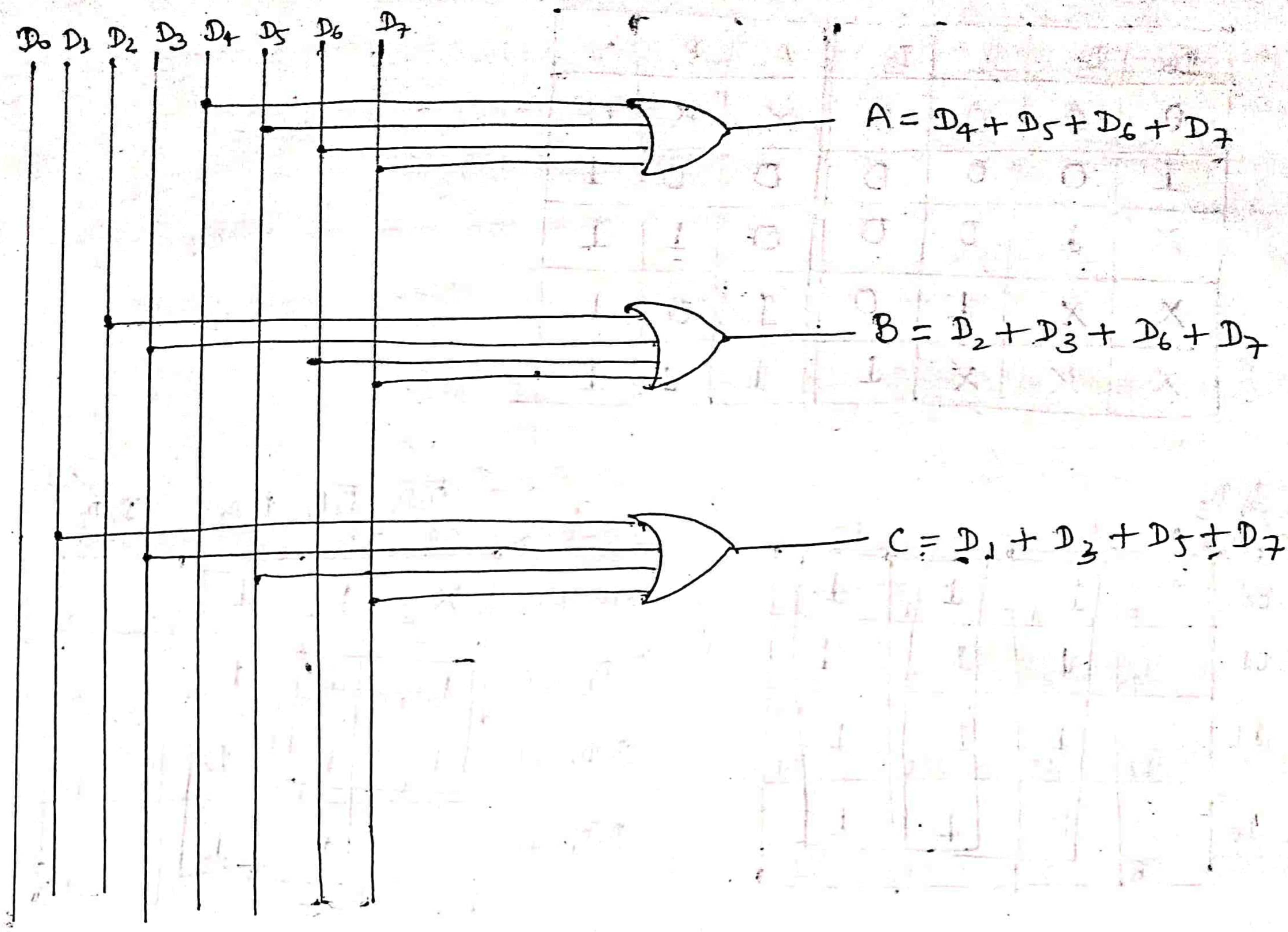
Octal to Binary Encoder :

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	A	B	C
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$A = D_4 + D_5 + D_6 + D_7$

$B = D_2 + D_3 + D_6 + D_7$

$C = D_1 + D_3 + D_5 + D_7$

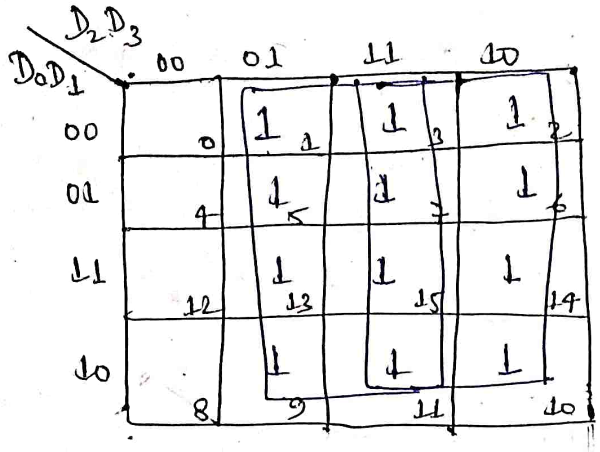


The limitation of encoder that only one input can be active at any given time. If two inputs are active simultaneously, the output produces an undefined combination. e.g. If D_3 and D_6 are high simultaneously, the output of the encoder will be 111 because all three outputs are equal to 1. The output 111 does not represent either binary 3 or 6. If both D_3 and D_6 are 1 at the same time, the output will be 110 because D_6 has higher ~~prioority~~ priority than D_3.

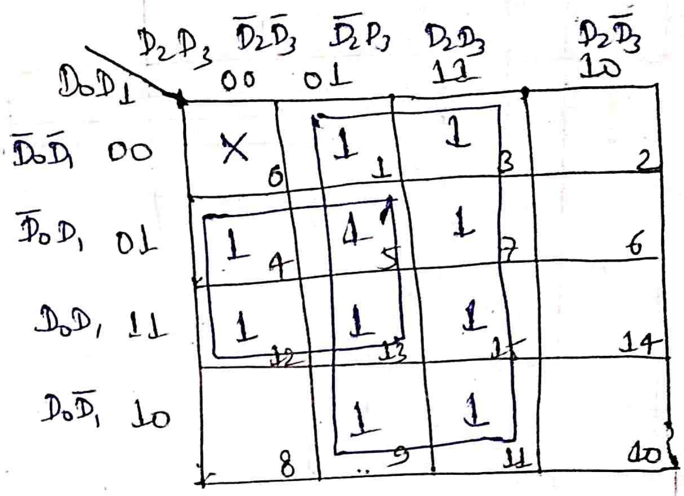
* Priority Encoders!

4 to 2 line encoder!

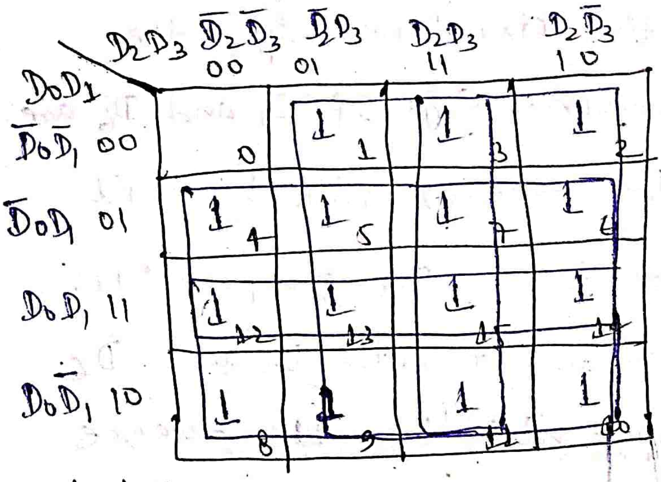
D_0	D_1	D_2	D_3	A	B	V
0	0	0	0	X	X	0
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1



$A = D_3 + D_2$



$B = D_1\bar{D}_2 + D_3$



4 octave,

$V = D_0 + D_1 + D_2 + D_3$

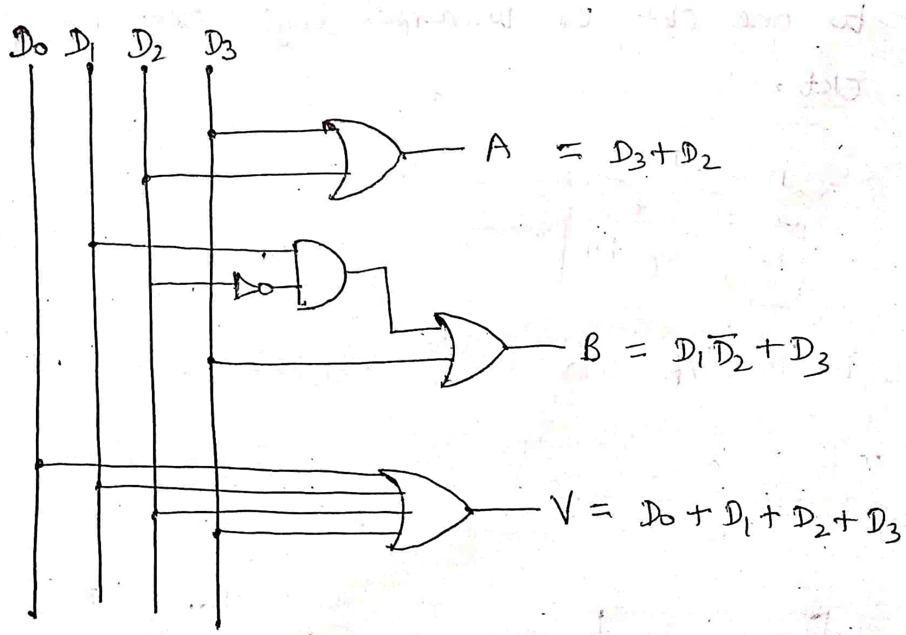
Priority encoder is an encoder circuit that includes priority function. If two or more inputs are equal to 1 at the same time, the input having the highest priority.

The truth table of a 4 input priority encoder and the two outputs

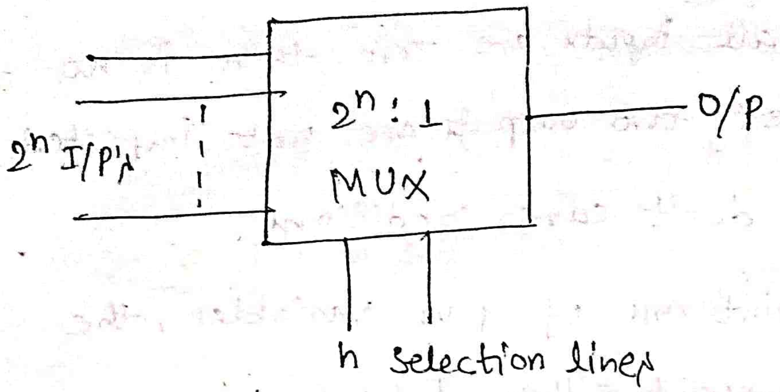
A and B, the circuit has 3^{rd} output designated by V, this is valid bit indicator. i.e. said to 1 when one or more inputs are equal to 1. If all inputs are zero, there is no valid output and $V=0$. The other two outputs are not inspected when $V=0$. and are specified as don't care conditions.

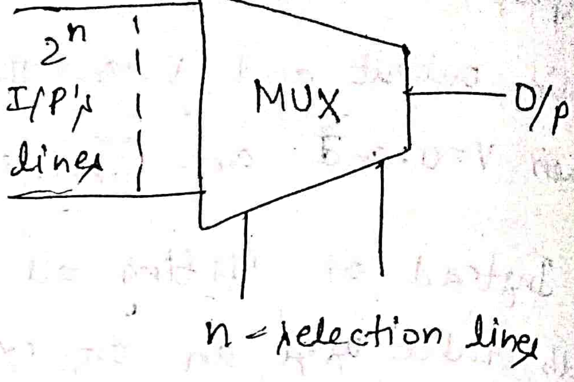
Instead of listing all 16 minterms of four variables, the truth table uses an cross(X) represent either 1 or 0.

The higher the subscript no., the higher the priority of the I/P. I/P D_3 has the highest priority so regardless of the values of the other I/P's, when this I/P is 1 the O/P is AB (11), D_2 has the next priority level. The output is (10) if $D_2=1$, provided that $D_3=0$, regardless of the values of other two lower priority I/P. The O/P for D_1 is generated only if higher priority I/P's are zero and so on down the priority levels.



* Multiplexer :





A multiplexer is a combinational circuit, that selects binary information from one of the many I/P lines and directs it to a single O/P line. The selection of a particular I/P line is controlled by a set of selection lines.

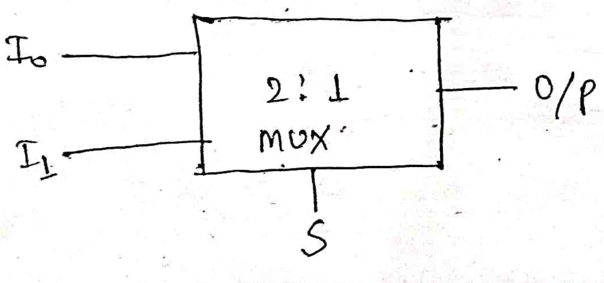
- It have many inputs and one O/P.
- Depending on control or select I/P's, one of the I/p line is transferred to the O/P line.
- It is a select input then also called as data selector or many to one ckt or universal logic ckt or parallel to serial ckt.

$$m = 2^n$$

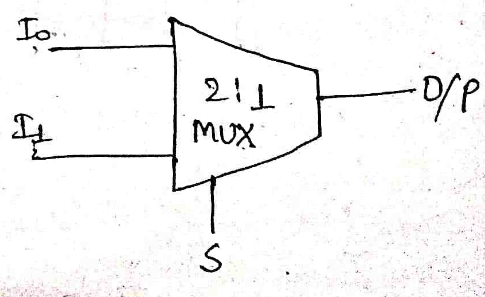
$$\boxed{n = \log_2 m}$$

Where m is I/p and n is selection line.

2:1 MUX : —



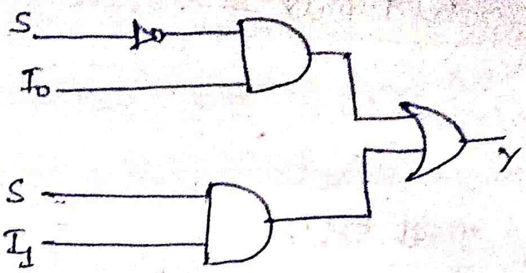
or



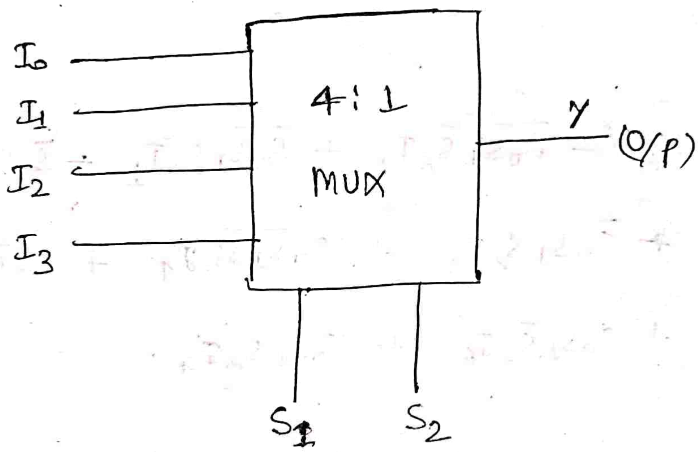
Truth Table :

S	Y
0	I_0
1	I_1

O/P, $Y = \bar{S} I_0 + S I_1$

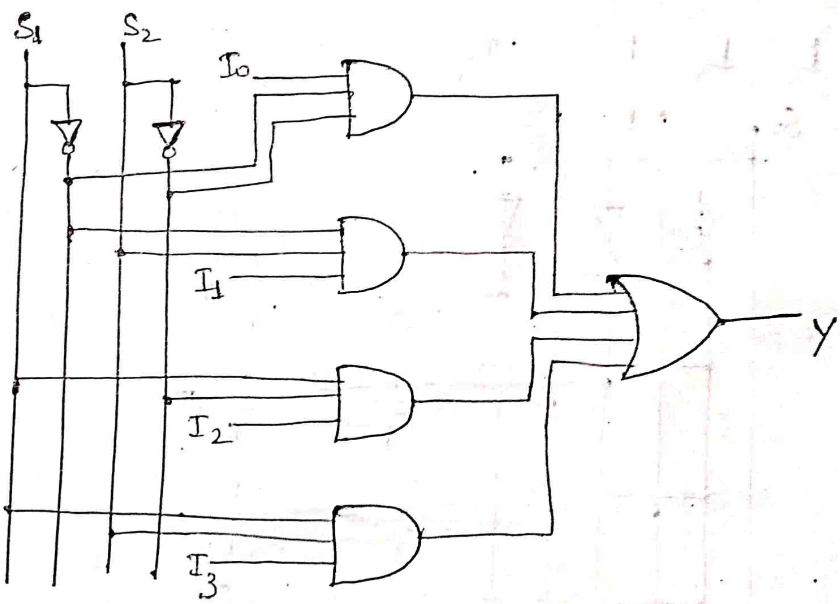


4:1 MUX :-



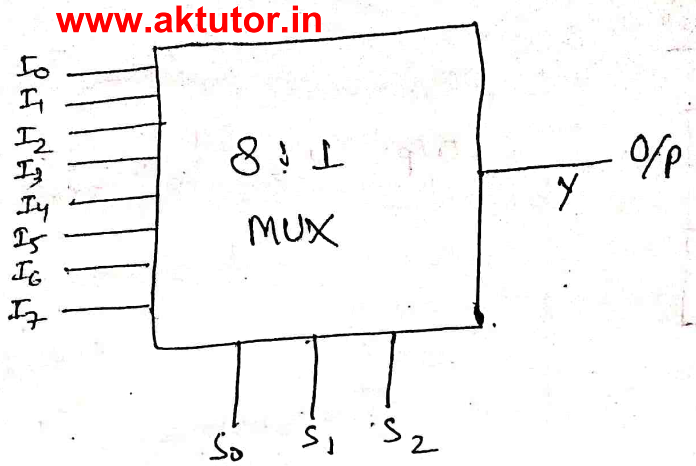
Truth Table :

S_1	S_2	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3



$$Y = \bar{S_1}\bar{S_2} I_0 + \bar{S_1} S_2 I_1 + S_1 \bar{S_2} I_2 + S_1 S_2 I_3$$

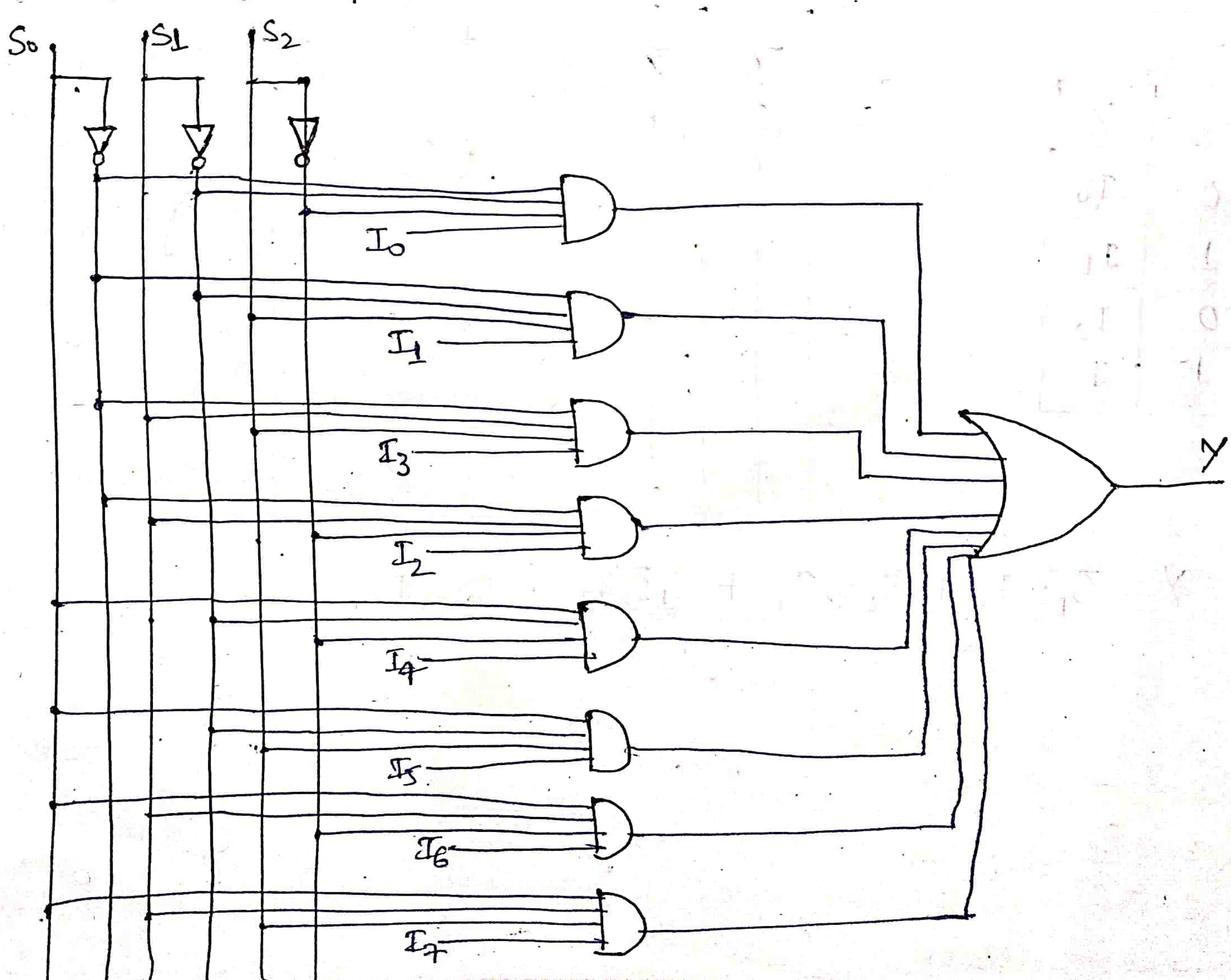
8:1 MUX :-



Truth Table :

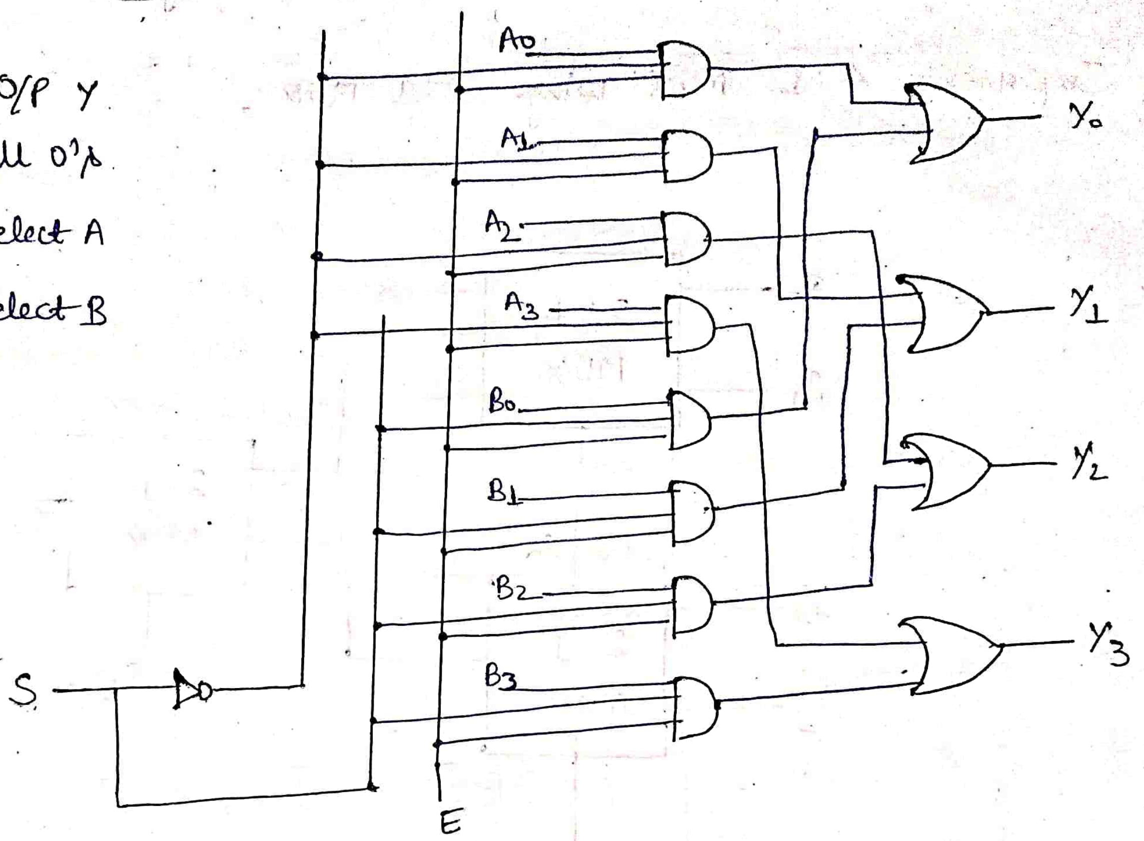
S_0	S_1	S_2	Y
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

$$Y = \bar{S}_0\bar{S}_1\bar{S}_2 I_0 + \bar{S}_0\bar{S}_1 S_2 I_1 + \bar{S}_0 S_1 \bar{S}_2 I_2 + \bar{S}_0 S_1 S_2 I_3 + S_0\bar{S}_1\bar{S}_2 I_4 + S_0\bar{S}_1 S_2 I_5 + S_0 S_1 \bar{S}_2 I_6 + S_0 S_1 S_2 I_7$$



* Quadruple two to one line Multiplexer !

E	S	O/P Y
0	X	all 0's
1	0	Select A
1	1	Select B

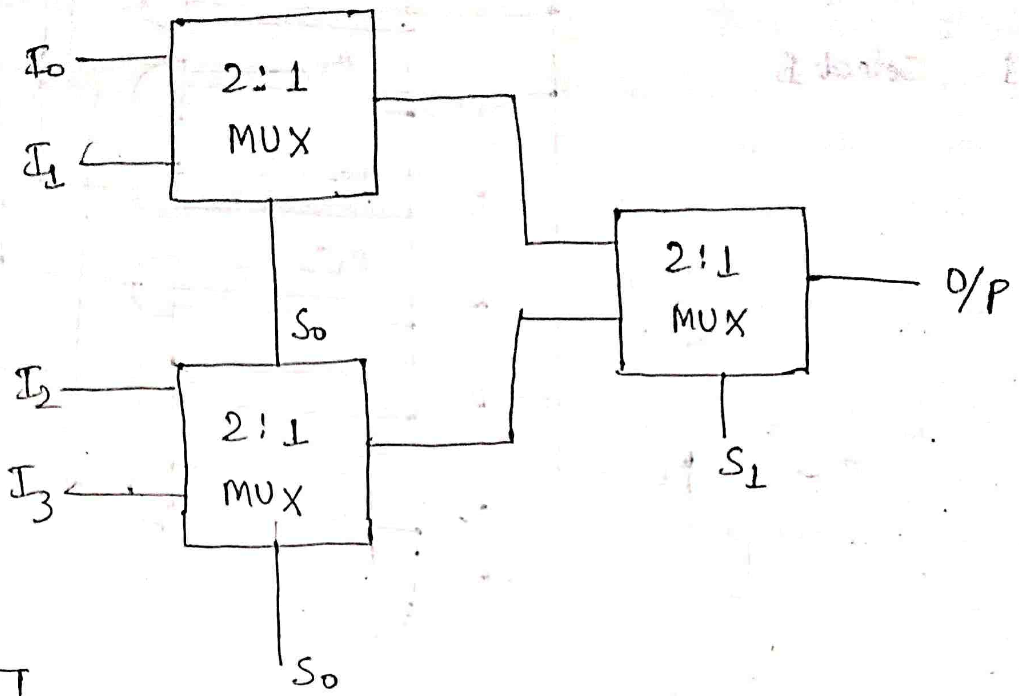


Multiplexer circuits can be combined with common selection I/P's to provide multiple bit selection logic. The ckt has 4 multiplexers, each capable of selecting one of two I/P lines. Output ~~Y~~ can be selected to come from either I/P's A_0 or I/P B_0. Similarly, Output Y_1 may have the value of A_1 or B_1 and so on..

The enable when $E=1$, then if $S=0$, the four A I/P's have a path to the four O/P's. If $S=1$ the four B I/P's are applied to the O/P's. The O/P's have all zeroes ~~be~~ when $E=0$, regardless value of the values of S.

* Implementation of Higher Multiplexer using Lower multiplexer :

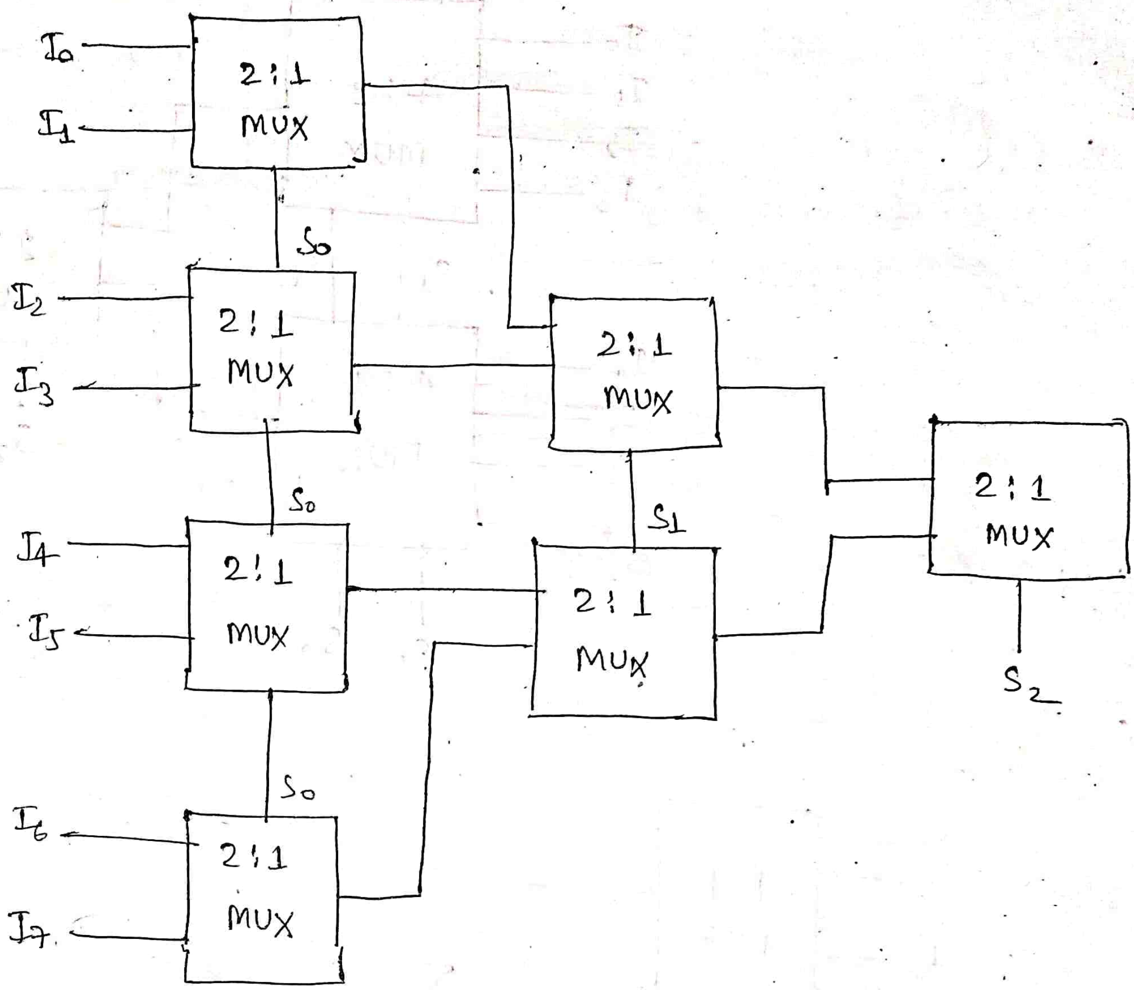
Implement 4:1 MUX using 2:1 MUX.



S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

(d) 8:1 MUX using 2:1 MUX

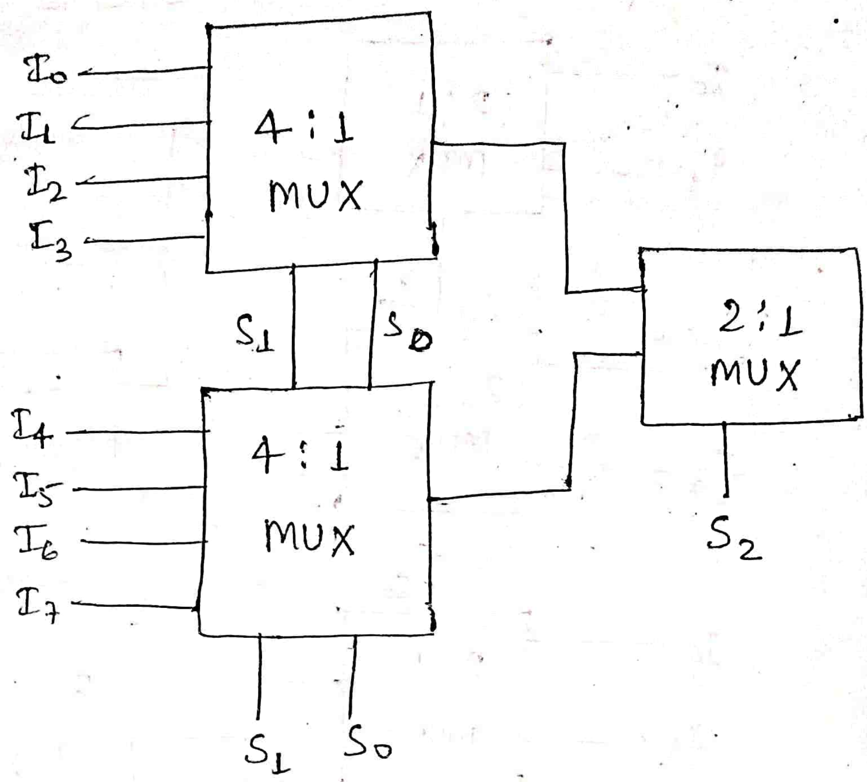
$$\text{No. of MUX} = \frac{\cancel{8}^{4}}{\cancel{2}} + \frac{\cancel{4}^{2}}{\cancel{2}} + \frac{\cancel{2}^{1}}{\cancel{2}} = 7$$

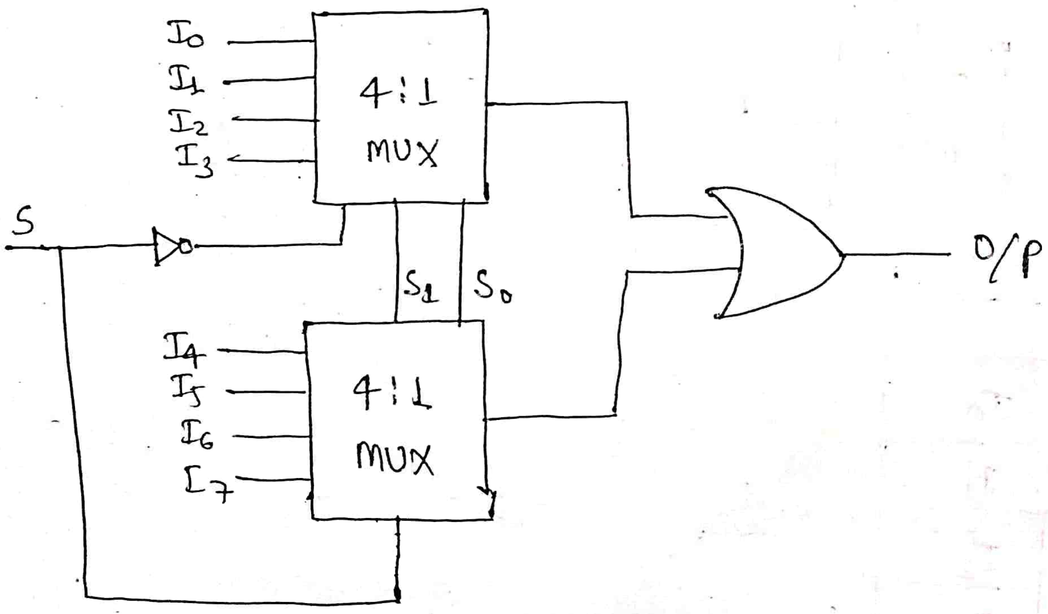


S_2	S_1	S_0	O/P
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

(8) 8:1 MUX using 4:1 MUX.

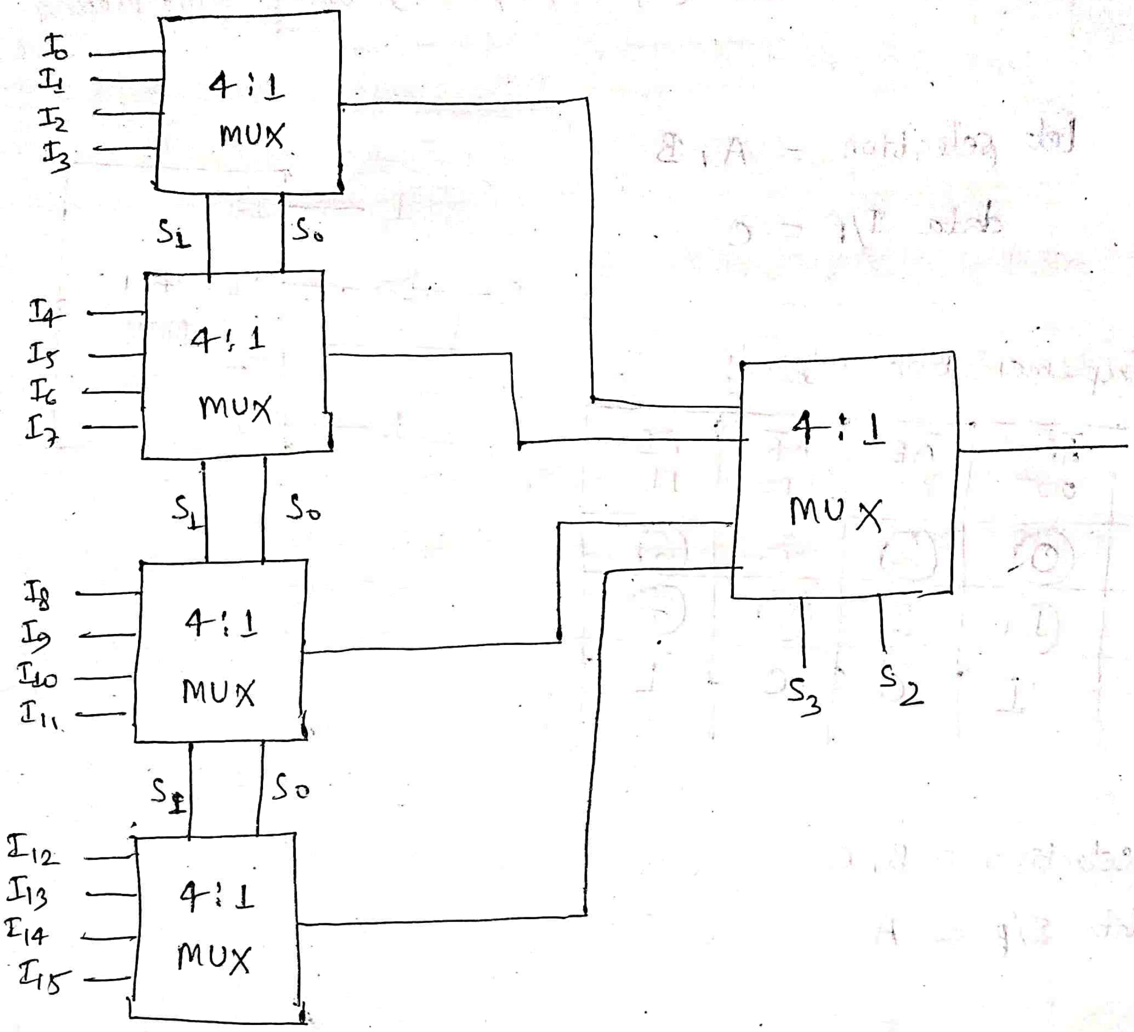
Sol^n: No. of MUX = $\frac{\cancel{8}^{2}}{\cancel{4}}$ = 2 MUX



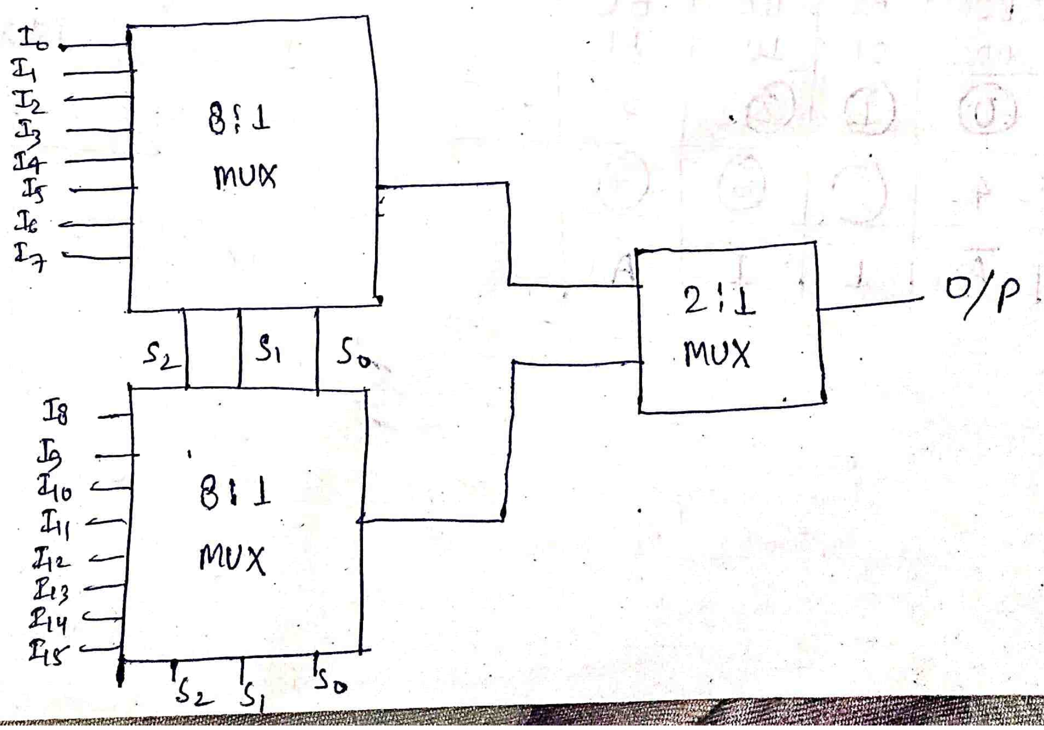


(Q) 16 : 1 MUX using 4 : 1 MUX.

No. of MUX $= \frac{\cancel{16}^{4}}{\cancel{4}} + \frac{4^{1}}{4} = 5$ mux.



(Q) 16 : 1 mux using 8 : 1 mux.



* Implementation of Boolean Function :

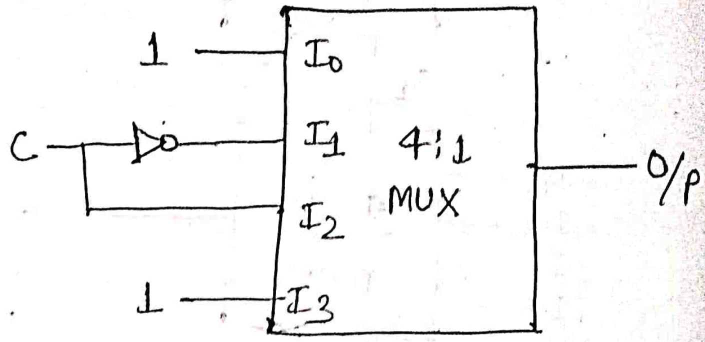
Ques: Implement the function

$F(A,B,C) = \sum m(0,1,2,5,6,7)$ using multiplexer.

Soln:

Let selection = A, B

data I/P = C



Implementation Table :

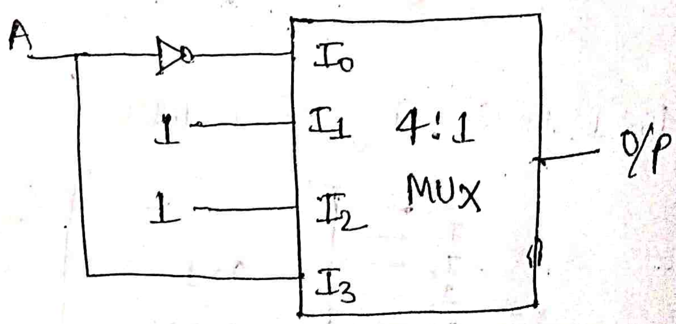
	$\bar{A}\bar{B}$ 00	$\bar{A}B$ 01	$A\bar{B}$ 10	AB 11
$\bar{C}$ 0	0	2	4	6
C 1	1	3	5	7
	1	$\bar{C}$	C	1

Let selection = B, C

data I/P = A

Implementation Table :

	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	$B\bar{C}$ 10	BC 11
$\bar{A}$ 0	0	1	2	3
A 1	4	5	6	7
	$\bar{A}$	1	1	A

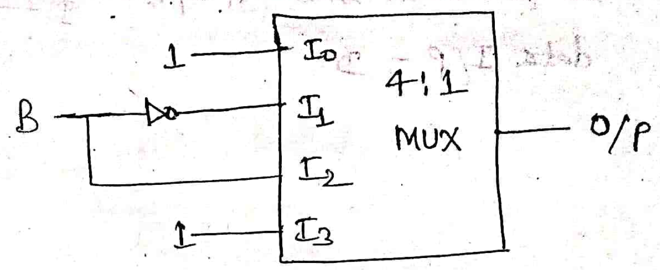


let Selection = A, C

data I/P = B

Implementation Table :

		$\bar{A}\bar{C}$ 00	$\bar{A}C$ 01	$A\bar{C}$ 10	AC 11
$\bar{B}$	0	0	1	4	5
B	1	2	3	6	7
		1	$\bar{B}$	B	1

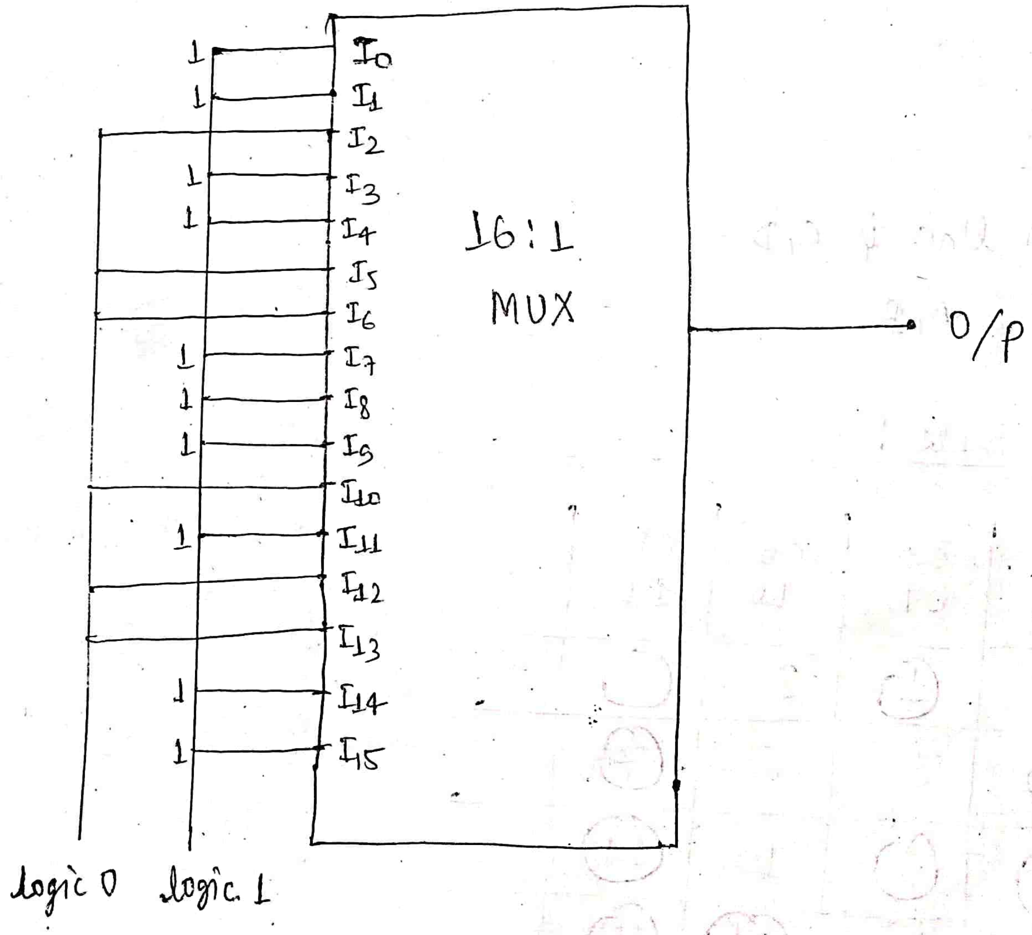


Ques: Implement the following function.

$$F(A, B, C, D) = \sum m(0, 1, 3, 4, 7, 8, 9, 11, 14, 15)$$

using 16:1 MUX, 8:1 MUX, 4:1 MUX, 2:1 MUX.

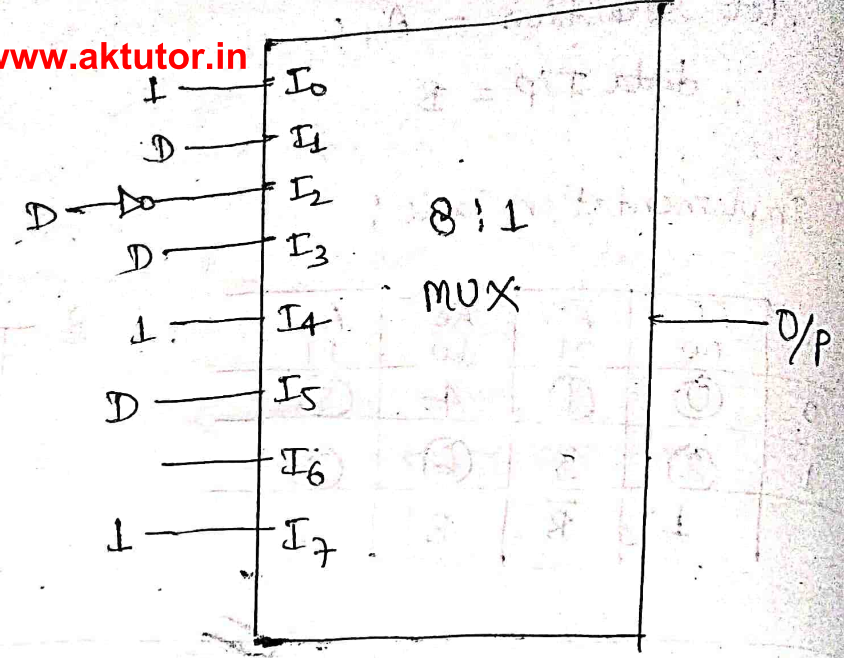
(i) Using 16:1 :



(ii) using 8:1 :-

let selection line = A, B, C

data I/P = D



Implementation Table :

	$\bar{A}\bar{B}\bar{C}$ 000	$\bar{A}\bar{B}C$ 001	$\bar{A}B\bar{C}$ 010	$\bar{A}BC$ 011	$A\bar{B}\bar{C}$ 100	$A\bar{B}C$ 101	$AB\bar{C}$ 110	ABC 111
$\bar{D}$ 0	0	2	4	6	8	10	12	14
D 1	1	3	5	7	9	11	13	15
	1	D	$\bar{D}$	D	1	D	0	1

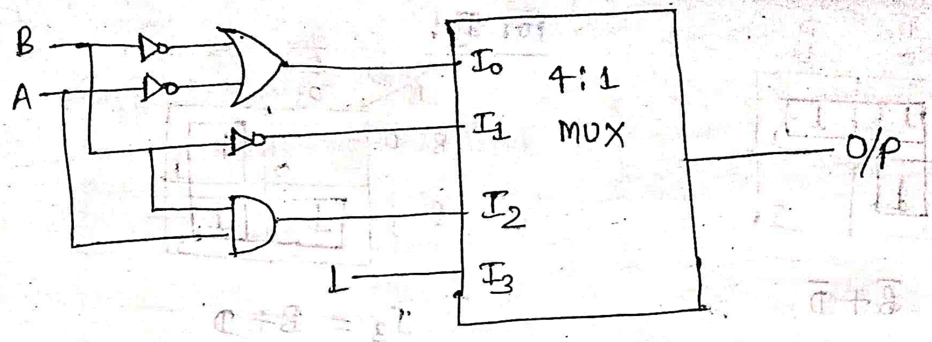
(iii) Using 4:1 :

① let selection line is C, D

Data I/P = A, B

Implementation Table :

	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	$C\bar{D}$ 10	CD 11
$\bar{A}\bar{B}$ 00	0	1	2	3
$\bar{A}B$ 01	4	5	6	7
$A\bar{B}$ 10	8	9	10	11
AB 11	12	13	14	15
	$(\bar{B}+\bar{A})$	$\bar{B}$	AB	1

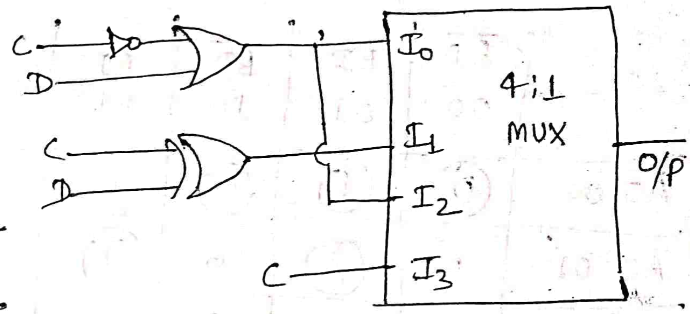


(2) let selection line = A, B

Data I/P = C, D

Implementation Table :

	$\bar{A}\bar{B}$ 00	$\bar{A}B$ 01	$A\bar{B}$ 10	AB 11
$\bar{C}\bar{D}$ 00	0	4	8	12
$\bar{C}D$ 01	1	5	9	13
$C\bar{D}$ 10	2	6	10	14
CD 11	3	7	11	15
	$(\bar{C}+D)$	$(C\odot D)$	$(\bar{C}+D)$	C

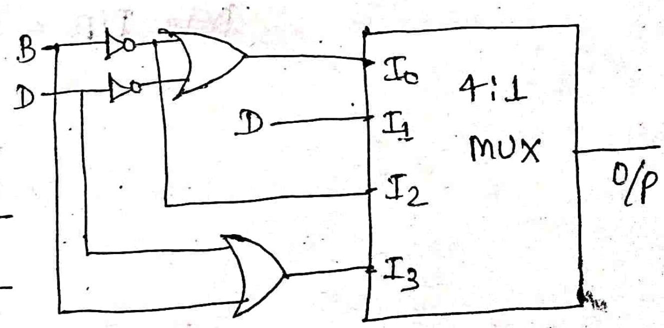


(3) let selection line = A, C

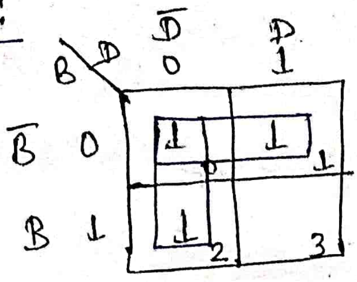
Data I/P = B, D

Implementation Table :

	$\bar{A}\bar{C}$ 00	$\bar{A}C$ 01	$A\bar{C}$ 10	AC 11
$\bar{B}\bar{D}$ 00	0	2	8	10
$\bar{B}D$ 01	1	3	9	11
$B\bar{D}$ 10	4	6	12	14
BD 11	5	7	13	15
	$(\bar{B}+\bar{D})$	D	$\bar{B}$	$(B+D)$

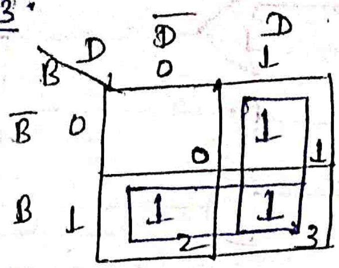


For I_0 :



$I_0 = \bar{B} + \bar{D}$

For I_3 :

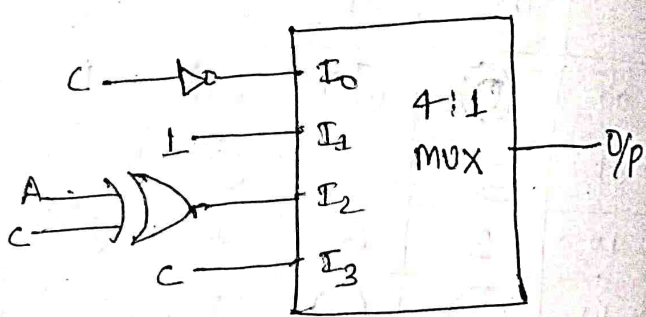


$I_3 = B + D$

④ let selection line = B, D
Data I/P = A, C.

Implementation Table :

	$\bar{B}\bar{D}$ 00	$\bar{B}D$ 01	$B\bar{D}$ 10	BD 11
$\bar{A}\bar{C}$ 00	0	1	4	5
$\bar{A}C$ 01	2	3	6	7
$A\bar{C}$ 10	8	9	12	13
AC 11	10	11	14	15
	$\bar{C}$	1	$A \odot C$	C



(iv) Using 2:1 : ① let the selection = D
Data I/P = A, B, C

	$\bar{D}$ 0	D 1
$\bar A \bar B \bar C$ 000	0	1
$\bar A \bar B C$ 001	2	3
$\bar A B \bar C$ 010	4	5
$\bar A B C$ 011	6	7
$A \bar B \bar C$ 100	8	9
$A \bar B C$ 101	10	11
$A B \bar C$ 110	12	13
ABC 111	14	15

For I_0 :-

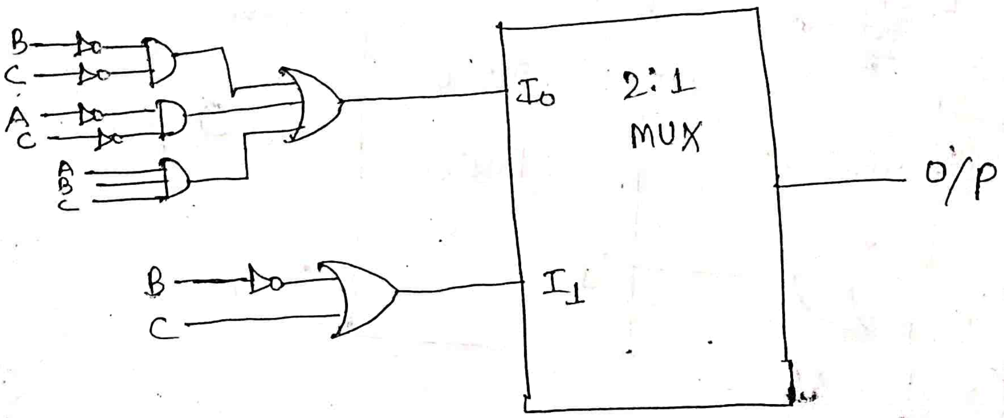
A \ BC	$\bar B \bar C$ 00	$\bar B C$ 01	BC 11	$B \bar C$ 10
$\bar A$ 0	1 (0)	(1)	(3)	1 (2)
A 1	1 (4)	(5)	1 (7)	(6)

$$I_0 = \bar B \bar C + \bar A \bar C + ABC$$

For I_1 :-

A \ BC	$\bar B \bar C$ 00	$\bar B C$ 01	BC 11	$B \bar C$ 10
$\bar A$ 0	1 (0)	1 (1)	1 (3)	(2)
A 1	1 (4)	1 (5)	1 (7)	(6)

$$I_1 = \bar B + C$$



(2) Let selection line A

Data I/P B, C, D

	$\bar{A}$ 0	A 1
$\bar{B}\bar{C}\bar{D}$ 000	0	8
$\bar{B}\bar{C}D$ 001	1	9
$\bar{B}C\bar{D}$ 010	2	10
$\bar{B}CD$ 011	3	11
$B\bar{C}\bar{D}$ 100	4	12
$B\bar{C}D$ 101	5	13
$BC\bar{D}$ 110	6	14
BCD 111	7	15

For I_0 :-

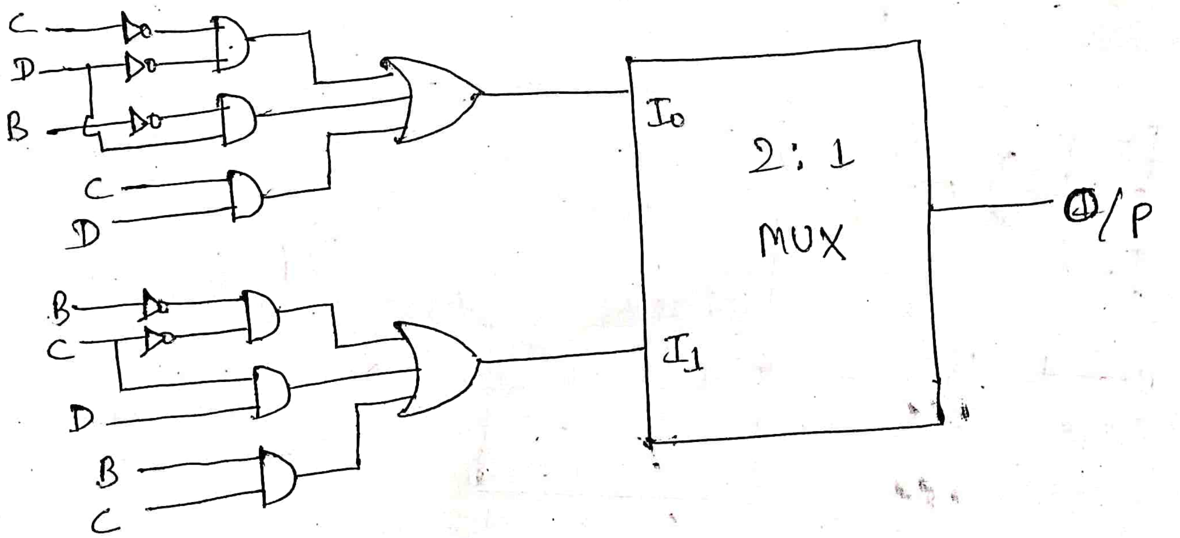
B \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{B}$ 0	1 (0)	1 (1)	1 (3)	(2)
B 1	1 (4)	(5)	1 (7)	(6)

$$I_0 = \bar{C}\bar{D} + \bar{B}D + CD$$

For I_1 :-

B \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{B}$ 0	1 (0)	1 (1)	1 (3)	(2)
B 1	(4)	(5)	1 (7)	1 (6)

$$I_1 = \bar{B}\bar{C} + CD + BC$$



(3) let the selection line B
Data I/P A, C, D

	$\bar{B}$ 0	B 1
$\bar{A}\bar{C}\bar{D}$ 000	0	4
$\bar{A}\bar{C}D$ 001	1	5
$\bar{A}C\bar{D}$ 010	2	6
$\bar{A}CD$ 011	3	7
$A\bar{C}\bar{D}$ 100	8	12
$A\bar{C}D$ 101	9	13
$AC\bar{D}$ 110	10	14
ACD 111	11	15

For I_0 :-

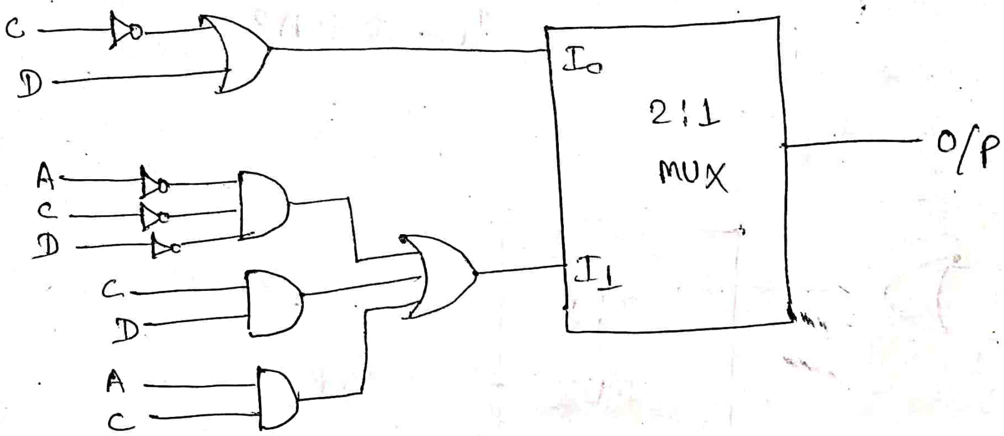
A \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}$ 0	1 ₀	1 ₁	1 ₃	2
A 1	1 ₄	1 ₅	1 ₇	6

$$I_0 = \bar{C} + D$$

For I_1 :-

A \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}$ 0	1 ₀	1	1 ₃	2
A 1	4	5	1 ₇	1 ₆

$$I_1 = \bar{A}\bar{C}\bar{D} + CD + AC$$



(4) Let selection line is C.

Data I/P A, B, D

		$\bar{C}$ 0	C 1
$\bar{A}\bar{B}\bar{D}$	000	0	2
ABD	001	1	3
ABD	010	4	6
ABD	011	5	7
ABD	100	8	10
ABD	101	9	11
ABD	110	12	14
ABD	111	13	15

For I_0 :-

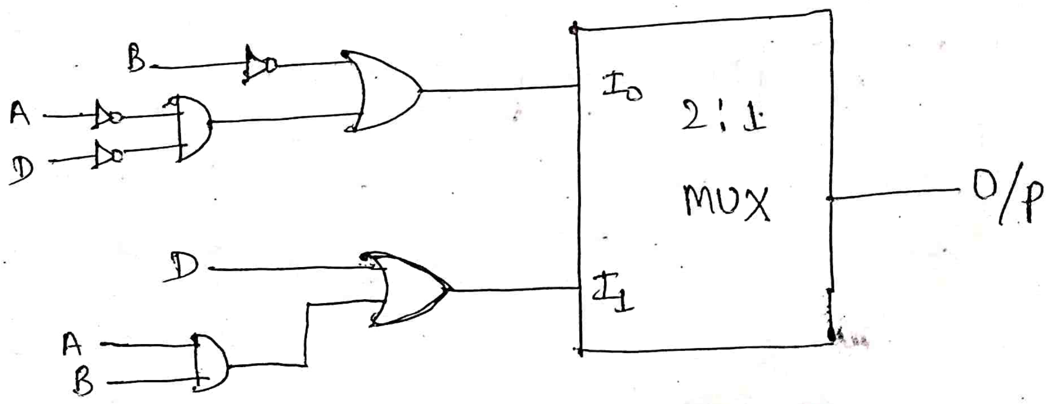
A \ BD	$\bar{B}\bar{D}$ 00	$\bar{B}D$ 01	BD 11	$B\bar{D}$ 10
$\bar{A}$ 0	1 (0)	1 (1)	(3)	1 (2)
A 1	1 (4)	1 (5)	0 (7)	0 (6)

$I_0 = \bar{B} + \bar{A}\bar{D}$

For I_1 :-

A \ BD	$\bar{B}\bar{D}$ 00	$\bar{B}D$ 01	BD 11	$B\bar{D}$ 10
$\bar{A}$ 0	(0)	1 (1)	1 (3)	(2)
A 1	(4)	1 (5)	1 (7)	1 (6)

$I_1 = D + AB$

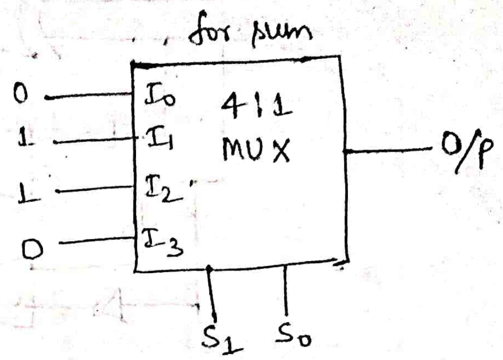


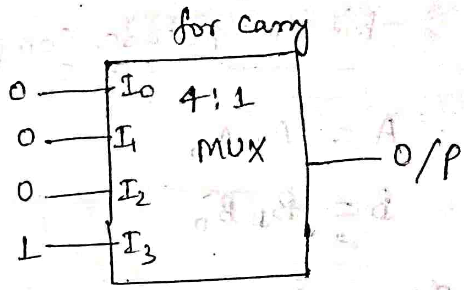
* Half-Adder using Multiplexer:

Selection line A, B

Truth Table:

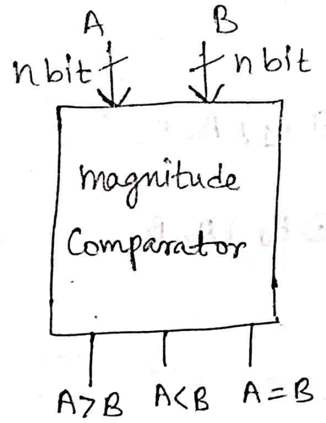
I/P		O/P	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1





* Magnitude Comparator: -

A magnitude comparator is a combinational circuit that compare two numbers A and B and determines their relative magnitude. It receives two n-bit numbers and gives the output $A<B$, $A>B$ and $A=B$.



Possibility $= 2^{2n}$.

* 1-bit magnitude Comparator:

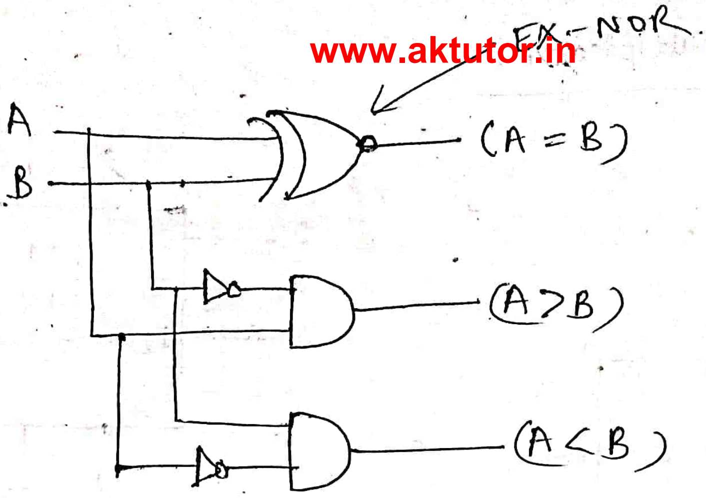
A	B	A < B	A = B	A > B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

$(A>B) = A\bar{B}$

$(A<B) = \bar{A}B$

$(A=B) = \bar{A}\bar{B} + AB$

$= A \odot B$



* 2-bit magnitude Comparator :-

$A = A_1 A_0$

$B = B_1 B_0$

$x_i = \bar{A}_i \bar{B}_i + A_i B_i \quad , \quad i = 0, 1, 2, ---$

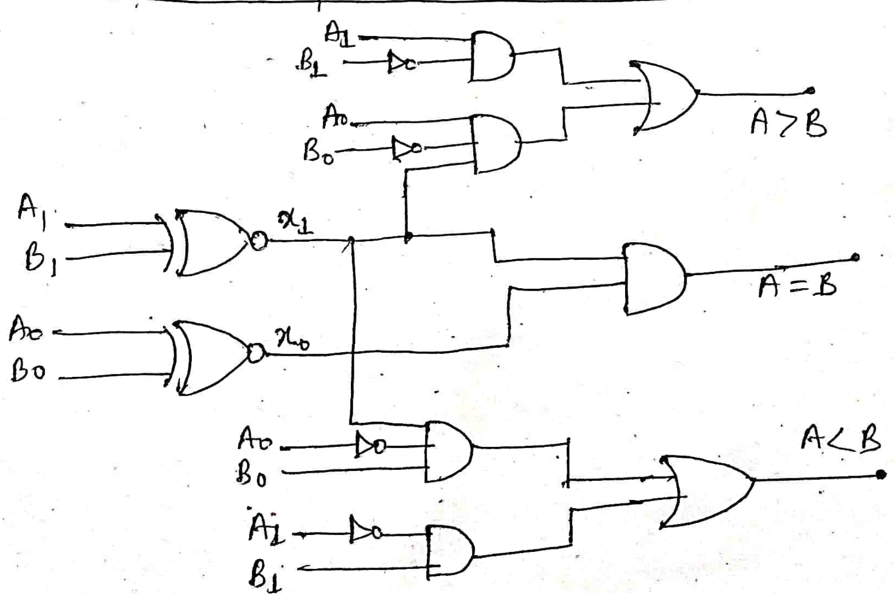
$(A = B) \quad (A_1 = B_1 \ \& \ A_0 = B_0)$

$$\left.\begin{aligned} x_0 &= \bar{A}_0 \bar{B}_0 + A_0 B_0 = A_0 \odot B_0 \\ x_1 &= \bar{A}_1 \bar{B}_1 + A_1 B_1 = A_1 \odot B_1 \end{aligned}\right\} \begin{aligned} x &= x_1 x_0 \\ &= (A_1 \odot B_1)(A_0 \odot B_0) \end{aligned}$$

$(A > B) = A_1 \bar{B}_1 + (A_1 \odot B_1) A_0 \bar{B}_0$

$(A < B) = \bar{A}_1 B_1 + (A_1 \odot B_1) \bar{A}_0 B_0$

A_1	A_0	B_1	B_0	$A>B$	$A<B$	$A=B$
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	0	0	1
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	0	1



* 3-bit magnitude Comparator :-

$A = A_2 A_1 A_0$

$B = B_2 B_1 B_0$

$A_2 = B_2$, $A_1 = B_1$, $A_0 = B_0$.

$x_i = \bar{A}_i \bar{B}_i + A_i B_i$, $i = 0, 1, 2, 3, --$

$x_0 = \bar{A}_0 \bar{B}_0 + A_0 B_0 = A_0 \odot B_0$

$x_1 = \bar{A}_1 \bar{B}_1 + A_1 B_1 = A_1 \odot B_1$

$x_2 = \bar{A}_2 \bar{B}_2 + A_2 B_2 = A_2 \odot B_2$

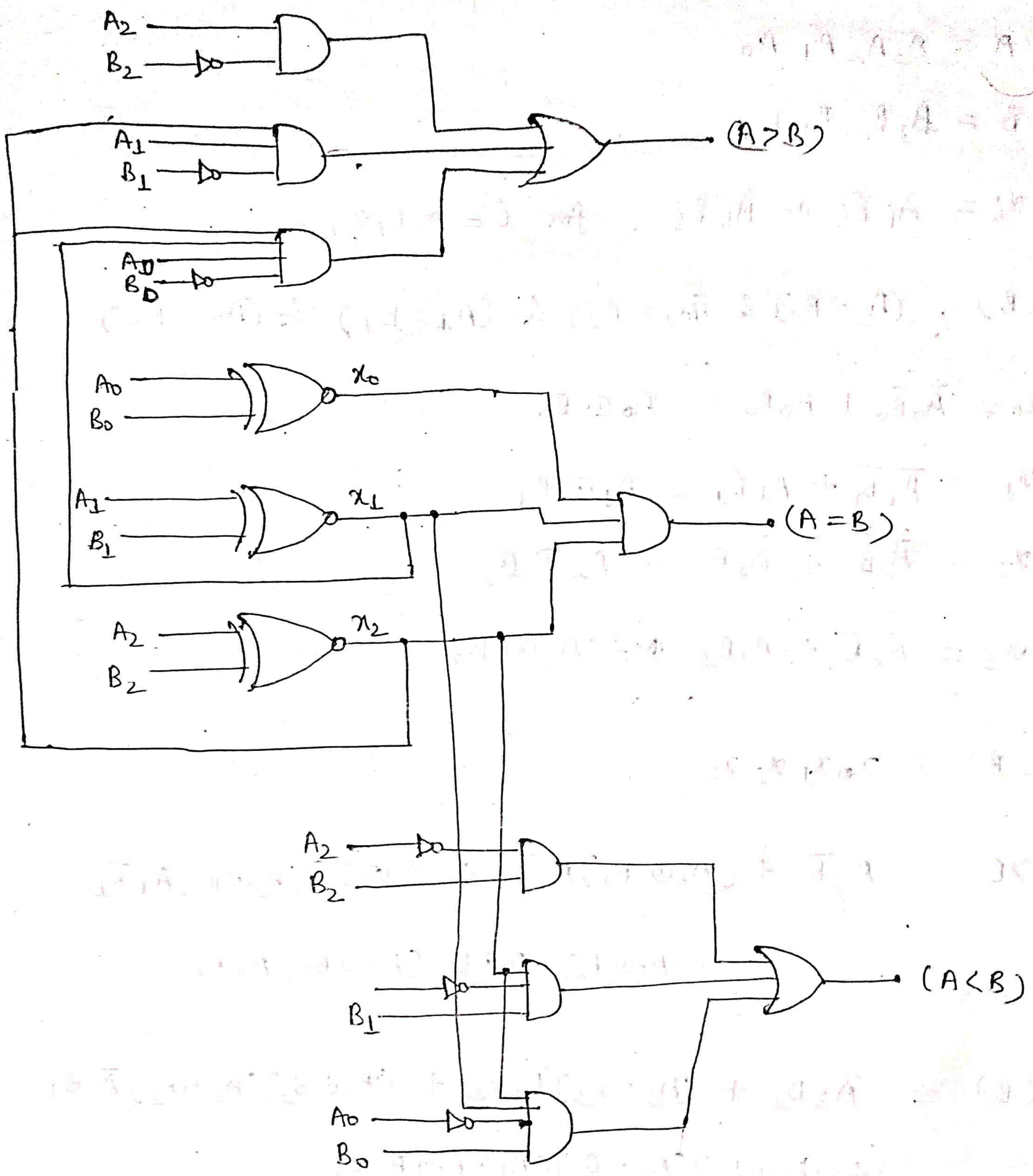
then,

$(A = B) = x_2 x_1 x_0$

$(A > B) = A_2 \bar{B}_2 + (A_2 \odot B_2) A_1 \bar{B}_1 + (A_2 \odot B_2)(A_1 \odot B_1) A_0 \bar{B}_0$

$(A < B) = \bar{A}_2 B_2 + (A_2 \odot B_2) \bar{A}_1 B_1 + (A_2 \odot B_2)(A_1 \odot B_1) \bar{A}_0 B_0$

* Circuit Diagram of 3-bit comparator :-



* 4-bit Comparator :—

$$A = A_3 A_2 A_1 A_0$$

$$B = B_3 B_2 B_1 B_0$$

$$x_i = \overline{A_i}\,\overline{B_i} + A_i B_i \quad \text{for } i = 0, 1, 2, ---$$

$(A=B)$, $(A_3=B_3)$ & $(A_2=B_2)$ & $(A_1=B_1)$ & $(A_0=B_0)$

$$x_0 = \overline{A_0}\,\overline{B_0} + A_0 B_0 = A_0 \odot B_0$$

$$x_1 = \overline{A_1}\,\overline{B_1} + A_1 B_1 = A_1 \odot B_1$$

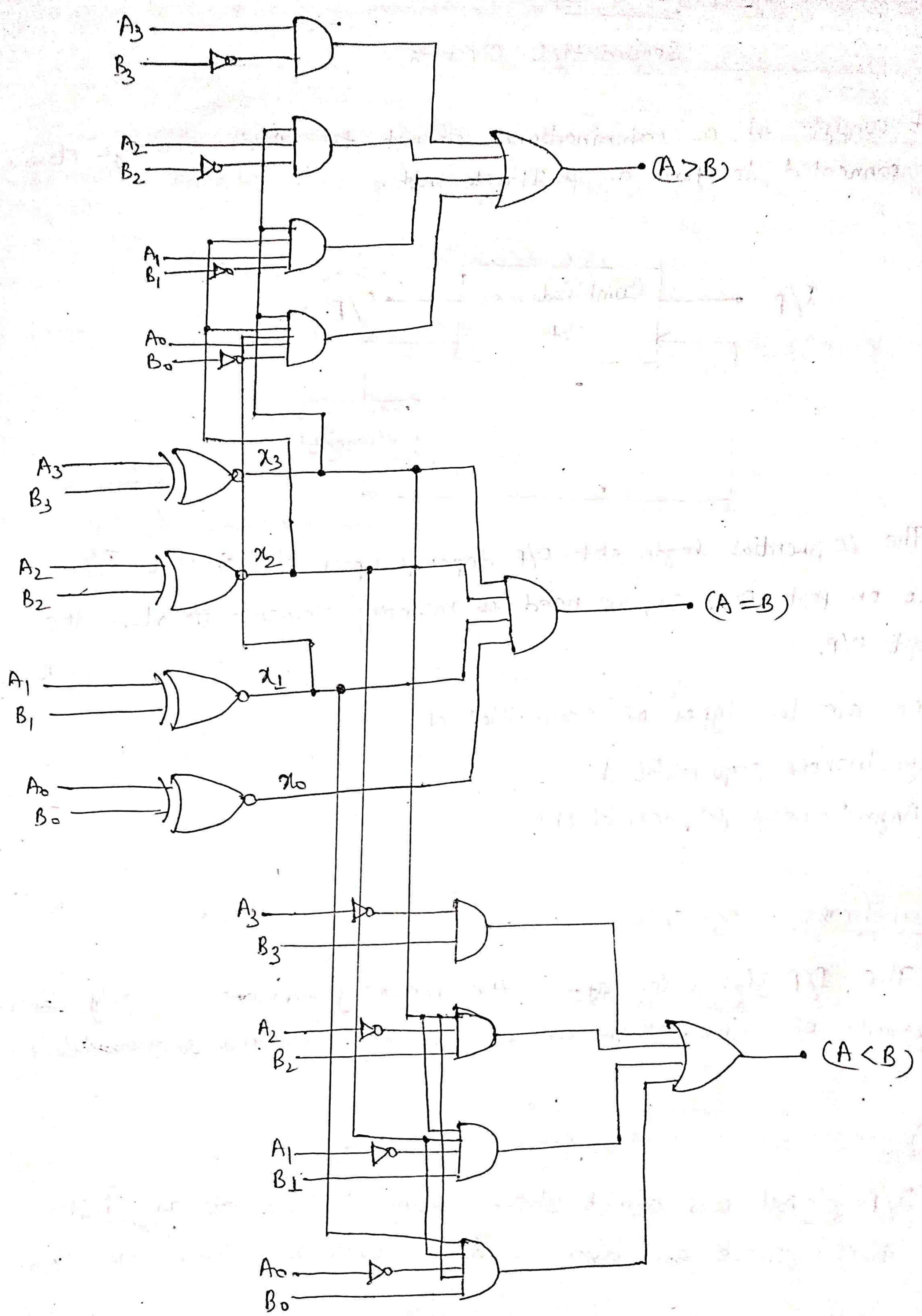
$$x_2 = \overline{A_2}\,\overline{B_2} + A_2 B_2 = A_2 \odot B_2$$

$$x_3 = \overline{A_3}\,\overline{B_3} + A_3 B_3 = A_3 \odot B_3$$

$$(A=B) = x_0 x_1 x_2 x_3$$

$$(A>B) = A_3 \overline{B_3} + (A_3 \odot B_3) A_2 \overline{B_2} + (A_3 \odot B_3)(A_2 \odot B_2) A_1 \overline{B_1} + (A_3 \odot B_3)(A_2 \odot B_2)(A_1 \odot B_1) A_0 \overline{B_0}$$

$$(A<B) = \overline{A_3} B_3 + (A_3 \odot B_3) \overline{A_2} B_2 + (A_3 \odot B_3)(A_2 \odot B_2) \overline{A_1} B_1 + (A_3 \odot B_3)(A_2 \odot B_2)(A_1 \odot B_1) \overline{A_0} B_0$$

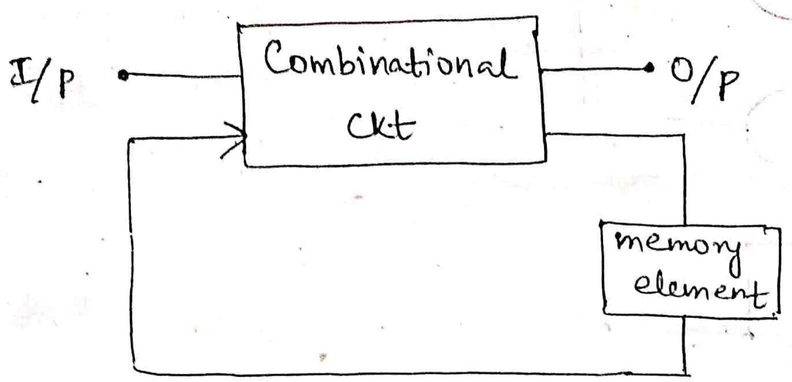


A3
B3
A2
B2
A1
B1
A0
B0
(A > B)
A3
B3
x3
A2
B2
x2
A1
B1
x1
A0
B0
x0
(A = B)
A3
B3
A2
B2
A1
B1
A0
B0
(A < B)

UNIT - 3

Sequential Circuits

It consist of a combinational circuit to which storage elements are connected to form a feedback path.



The sequential logic ckt O/P depends upon the present I/P and also on past O/P. So, we need a memory element to store the past O/P.

There are two types of sequential ckt:

① Synchronous sequential ckt.

② Asynchronous sequential ckt.

* Synchronous sequential ckt:

The I/P signal can affect the memory element at only discrete instant of time. These are also called as clocked sequential ckt.

* Asynchronous sequential ckt:

I/P signal can affect the memory element at any instant of time. These are also called as unclocked sequential ckt.

* Difference between combinational and sequential ckt:

Combinational ckt	Sequential ckt
① The output depends on the present input.	① The output depends not only on the present input but also on the past output.
② Memory unit is not required.	② Memory unit is required to store the past output.
③ These ckt are faster but the delay between the input and the output is due to propagation delay of logic gates.	③ The ckt are slower than combinational ckt.
④ It is easy to design.	④ It is difficult to design.

* Difference between synchronous and Asynchronous sequential ckt.

~~Sequential~~ Synchronous	Asynchronous
① Memory elements are clocked flip-flop.	① Memory elements are either unclocked flip-flop or time delay element.
② The change in input signal can affect memory elements upon activation of clock signal.	② The change in input signal can affect memory element at any instant.
③ The maximum operating speed of the clock depends on time delay involve.	③ The absence of the clock, asynchronous can operate faster than synchronous ckt.
④ Easy to design	④ Difficult to design.

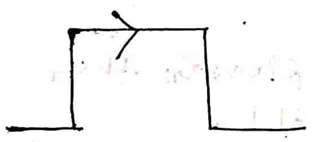
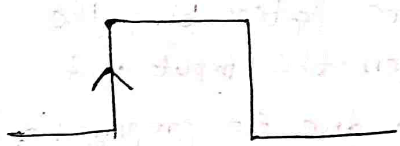
elements

nd

rete

ct.

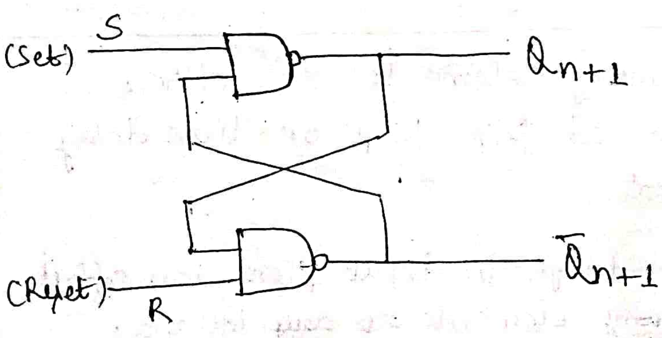
* Difference between latch and flip-flop :

Latch	Flip-Flop
① level triggered.	① Edge triggered.
② Used in asynchronous ckt.	② In synchronous ckt.
③ No clock signal	③ clock signal.
	

* <u>Edge Trigger</u> : In edge trigger ckt output may change only once in single clock.

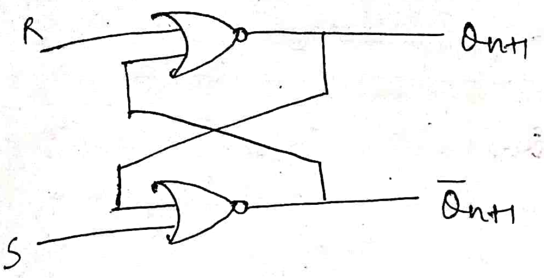
* <u>level Trigger</u> : In level trigger, ckt output may changes many time in single clock.

* SR latch using NAND gate :



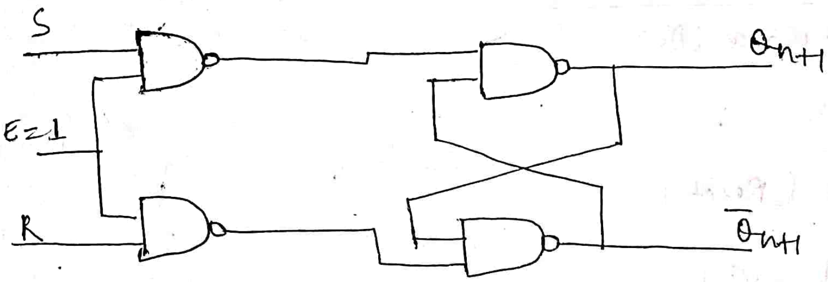
S	R	Q_{n+1}	$\overline{Q}_{n+1}$	
0	0	1	1	invalid
0	1	1	0	Set
1	0	0	1	reset
1	1	no change		

* SR latch using NOR gate :



S	R	Q_{n+1}	$\overline{Q}_{n+1}$	
0	0	no change		
0	1	0	1	Reset
1	0	1	0	Set
1	1	1	1	Invalid

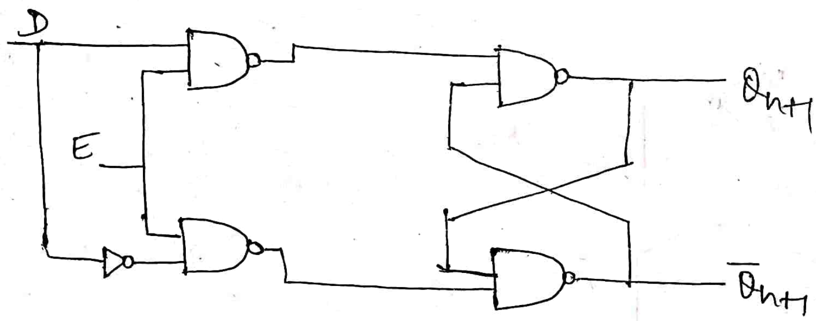
* SR latch with control input :



Truth Table :

E	S	R	Q_{n+1}
0	X	X	No change
1	0	0	Q_n
1	0	1	0 Reset
1	1	0	1 Set
1	1	1	Invalid State

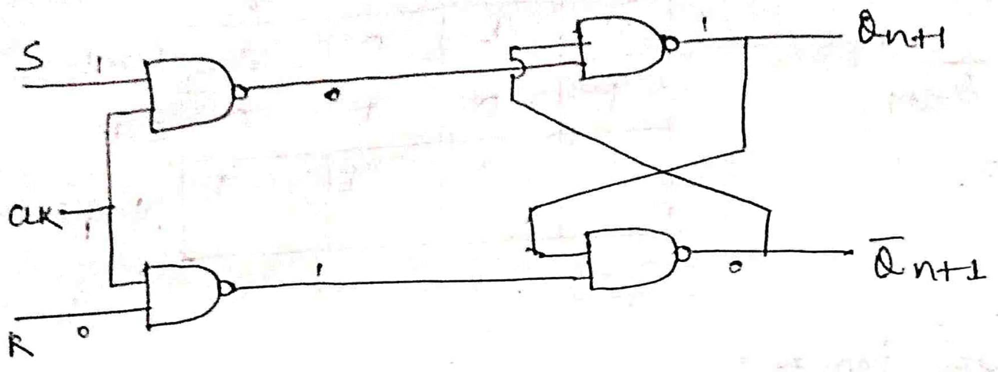
* D-latch with control input :



E	D	Q_{n+1}
0	X	No change
1	0	0 (Reset)
1	1	1 (Set)

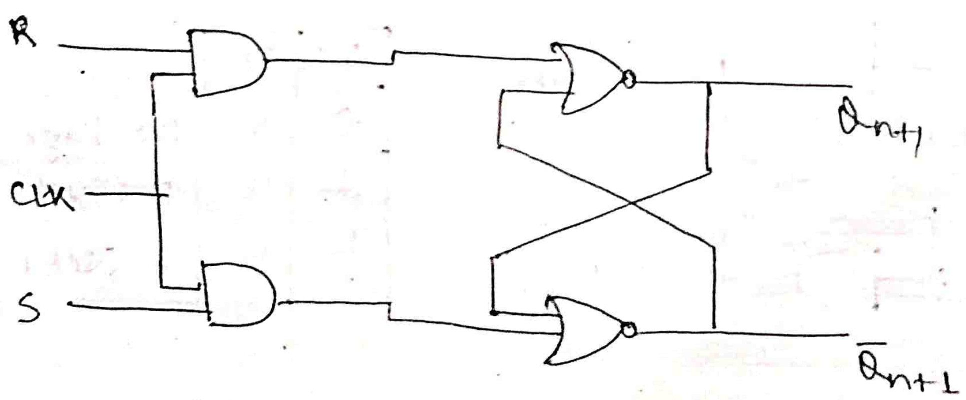
* Storage Element Flip-Flop:

⊙ SR Flip-Flop with NAND gate:



CLK	S	R	Q_{n+1}
0	X	X	No change (Q_n)
1	0	0	Q_n
1	0	1	0 (Reset)
1	1	0	1 (Set)
1	1	1	Invalid state

⊙ SR-FF using NOR gate:

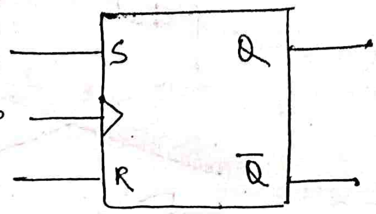


CLK	S	R	Q_{n+1}
0	X	X	No change (Q_n)
1	0	0	Q_n
1	0	1	Reset (0)
1	1	0	Set (1)
1	1	1	Invalid

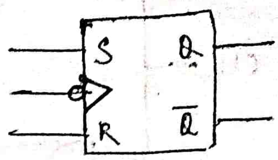
* Characteristic Table for SR-FF :

S	R	Q_n	Q_{n+1}	
0	0	0	0	
0	0	1	1	
0	1	0	0	(Reset)
0	1	1	0	
1	0	0	1	(Set)
1	0	1	1	
1	1	0	X	
1	1	1	X	

IC for SR-FF :



+ve edge triggered



−ve edge triggered

⊙ Characteristic Equation :

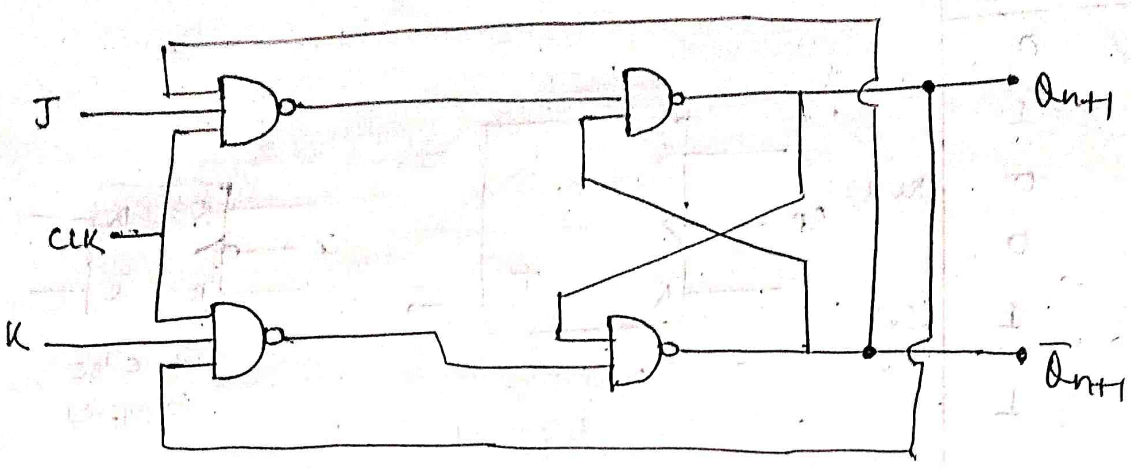
S \ RQ_n	00	01	11	10
$\overline{S}$ 0	(0)	1 (1)	(3)	(2)
S 1	1 (4)	1 (5)	X (7)	X (6)

$$\boxed{Q_{n+1} = S + \overline{R}Q_n}$$

⊙ Excitation Table :

P.S	N.S	FF - I/p	
Q_n	Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

* Jack Kilby (JK) Flip Flop :



Truth Table :

CLK	J	K	Q_{n+1}
0	X	X	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	$\overline{Q}_n$ (toggle)

Characteristic Table :

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Characteristic equation :

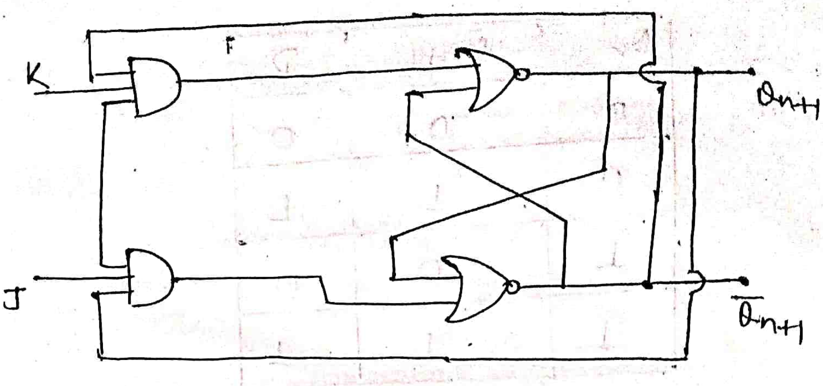
J \ KQ_n	00	01	10	11
0	(0)	1 (1)	(3)	(2)
1	1 (4)	1 (5)	(7)	1 (6)

$$\boxed{Q_{n+1} = J\overline{Q}_n + \overline{K}Q_n}$$

JK-FF using NOR gate:

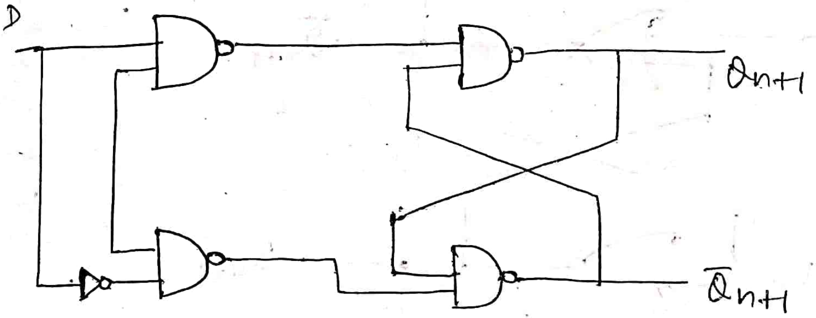
Excitation Table:

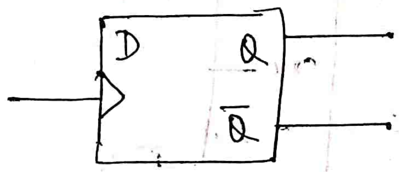
P.S	N.S	FF-I/P	
Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0



* D-Flip-Flop (Transparent Flip Flop):

[Drawback is many time toggle, race around condition of JK.]





CLK	D	Q_{n+1}
0	X	Q_n (no change)
1	0	0
1	1	1

⊙ Characteristic Table:

D	Q_n	Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

Characteristic eq^n:

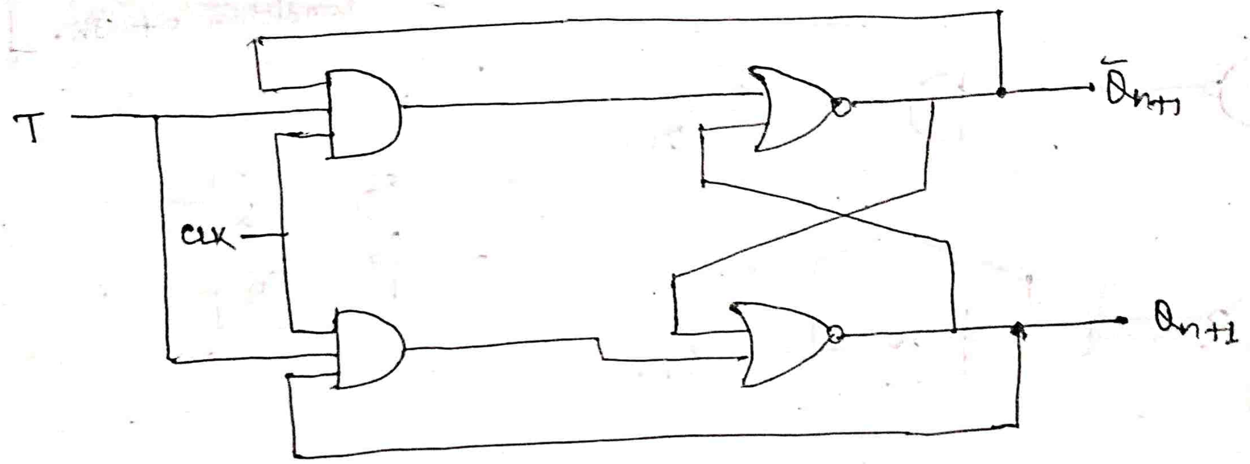
D \ Q_n	$\overline{Q_n}$	Q_n
$\overline{D}$	(0)	(1)
D	1 (2)	1 (3)

$$\boxed{Q_{n+1} = D}$$

Excitation Table:

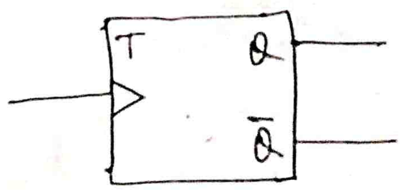
Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

* T- Flip Flop or Toggle Flip Flop:



* Truth Table:

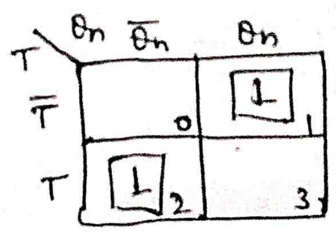
CLK	T	Q_{n+1}
0	X	Q_n
1	0	Q_n
1	1	$\bar{Q}_n$



* Characteristic Table:

T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

* Characteristic Equation:



$$Q_{n+1} = \bar{T}Q_n + T\bar{Q}_n$$
$$= T \oplus Q_n$$

* **Excitation Table:**

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

* **Conversion of one Flip-Flop to another Flip-Flop:**

	J	K	Q_{n+1}	
SR-FF	0	0	Q_n	T-FF
	0	1	0	D-FF
	1	0	1	
	1	1	$\bar{Q}_n$	

Since, all tables are inside in JK-FF, Therefore it is also called as universal Flip-Flop.

* **Excitation Table:**

Q_n	Q_{n+1}	S	R	J	K	D	T
0	0	0	X	0	X	0	0
0	1	1	0	1	X	1	1
1	0	0	1	X	1	0	1
1	1	X	0	X	0	1	0

(Q-1) Convert: (i) SR-FF to JK-FF
(ii) SR-FF to D-FF
(iii) SR-FF to T-FF

Sol^n: (i) SR-FF to JK-FF

* Procedure:

(i) Required FF characteristic table.
(ii) Given FF excitation table.
(iii) Using conversion table
(iv) Write logical expression for excitation.

Sol^n: (i)

* Characteristic Table of JK-FF:

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

* Excitation Table of SR-FF:

Q_n	Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

* Conversion Table :

J	K	Q_n	Q_{n+1}	S	R
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	X	0
1	1	0	1	1	0
1	1	1	0	0	1

For S :

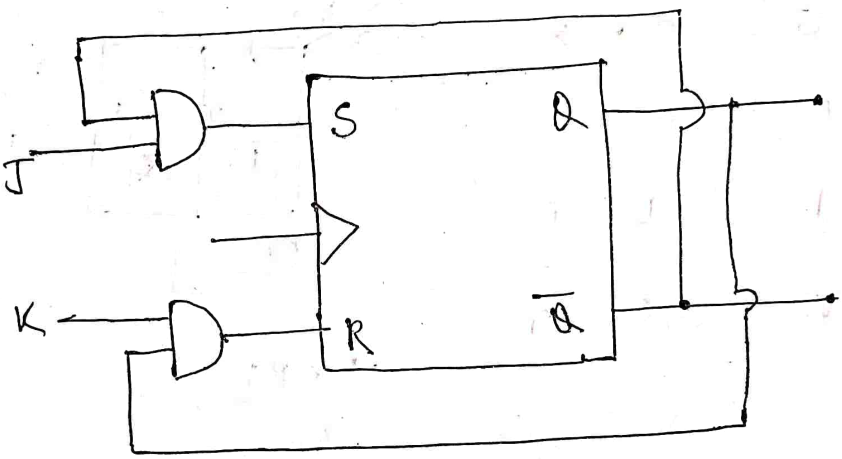
J \ KQ_n	00	01	11	10
0	(0)	X (1)	(3)	(2)
1	1 (4)	X (5)	(7)	1 (6)

$$\boxed{S = J\overline{Q}_n}$$

For R :

J \ KQ_n	00	01	11	10
0	X (0)	(1)	1 (3)	X (2)
1	(4)	(5)	1 (7)	(6)

$$\boxed{R = KQ_n}$$



Soln : (ii) SR-FF to D-FF :

* Characteristic Table of D-FF :

D	Q_n	Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

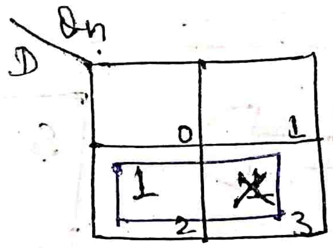
* Excitation Table of SR-FF:

Q_n	Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

* Conversion Table:

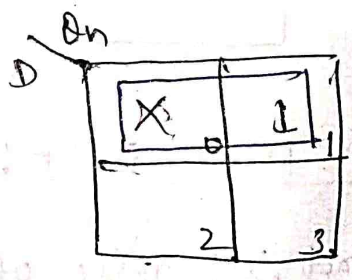
D	Q_n	Q_{n+1}	S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

For S:

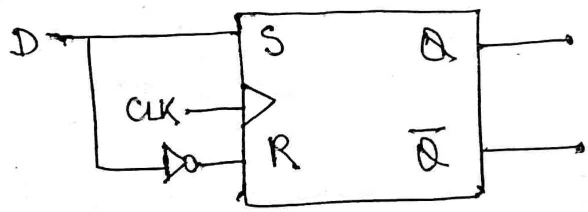


$\boxed{S = D}$

For R:



$\boxed{R = \overline{D}}$



Sol^n: (iii) SR-FF to T-FF!

* Characteristic Table of T-FF!

T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

* Excitation Table of SR-FF :

Q_n	Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

For S :

$T \backslash Q_n$	$\overline{Q_n}$	Q_n
$\overline{T}$	0 (0)	X (1)
T	$\boxed{1}$ (2)	(3)

$$\boxed{S = T\overline{Q_n}}$$

* Conversion Table :

T	Q_n	Q_{n+1}	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1

For R :

$T \backslash Q_n$	$\overline{Q_n}$	Q_n
$\overline{T}$	X (0)	(1)
T	(2)	$\boxed{1}$ (3)

$$\boxed{R = TQ_n}$$

(Q-2) Convert D-FF to SR, JK, T FF.

Sol^n: (I) D-FF to SR-FF :

* characteristic Table of SR-FF :

S	R	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X
1	1	1	X

* Excitation Table of D-FF:

D	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

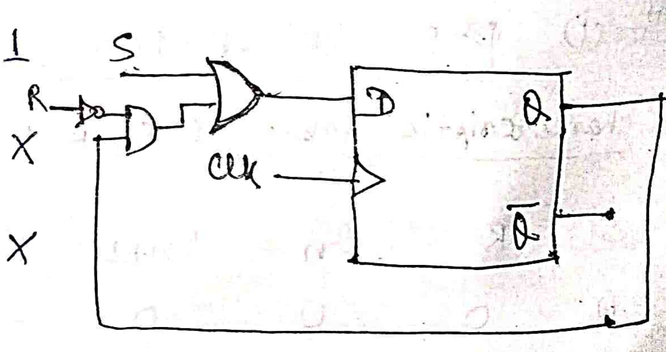
* Conversion Table:

S	R	Q_n	Q_{n+1}	D
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	X	X
1	1	1	X	X

For D:

S \ RQ_n	00	01	11	10
0	(0)	1 (1)	(3)	(2)
1	1 (4)	1 (5)	X (7)	X (6)

$$D = S + \overline{R}Q_n$$



(ii) ~~DFF~~ D-FF to JK-FF:

* characteristic Table of JK-FF:

J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

* Excitation Table of D-FF:

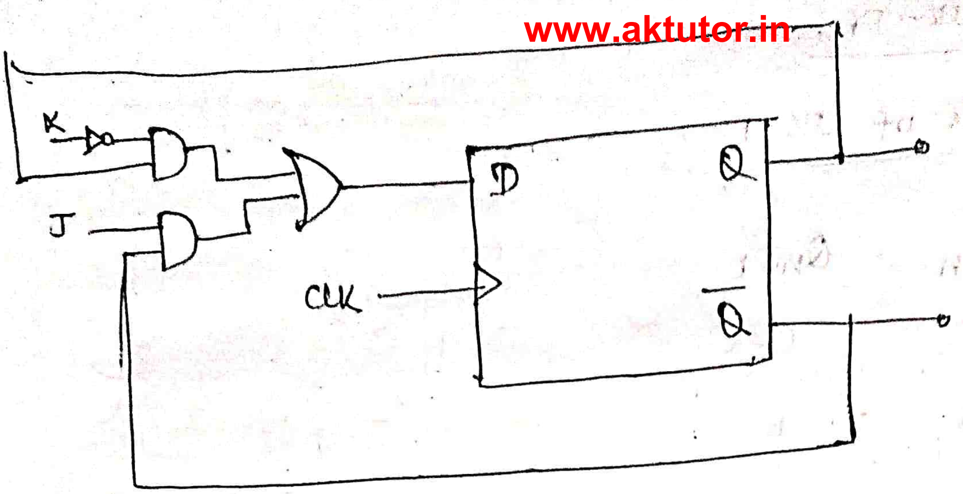
Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

* Conversion Table:

J	K	Q_n	Q_{n+1}	D
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

J \ KQ_n	00	01	11	10
0	0	1 1	3	2
1	1 4	1 5	7	1 6

$$\boxed{D = \overline{K}Q_n + J\overline{Q_n}}$$



(iii) D-FF to T-FF:

* Characteristic Table of T-FF:

T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

* Excitation Table of D-FF:

D	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

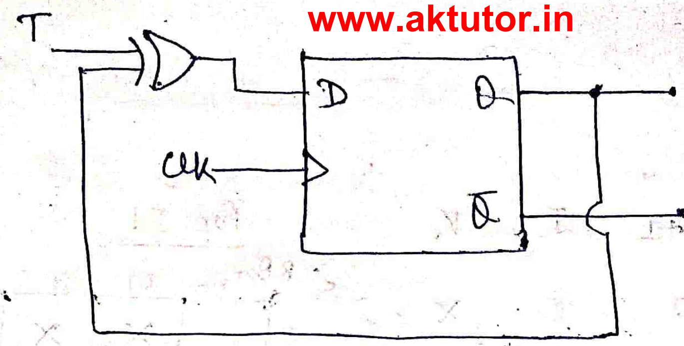
* Conversion Table:

T	Q_n	Q_{n+1}	D
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

For D:

T \ Q_n	$\bar{Q}_n$	Q_n
$\bar{T}$	0	1
T	1	0

$$\boxed{D = T\bar{Q}_n + \bar{T}Q_n}$$



Q-3 (i) JK-FF to SR-FF :

* characteristic Table of SR-FF :

S	R	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X
1	1	1	X

* Excitation Table of JK-FF :

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

* **Conversion Table:**

S	R	Q_n	Q_{n+1}	J	K
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	0	1	1	X	0
1	1	0	X	X	X
1	1	1	X	X	X

For J:

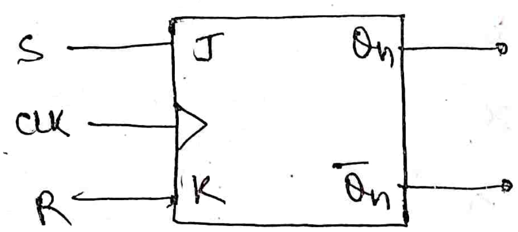
S \ RQ_n	00	01	11	10
0		X	X	
1	1	X	X	X

$\boxed{J = S}$

For K:

S \ RQ_n	00	01	11	10
0	X		1	X
1	X		X	X

$\boxed{K = R}$



Soln (ii): **JK-FF to D-FF:**

* **Characteristic Table of D-FF:**

D	Q_n	Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

* Excitation Table of JK-FF:

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

* Conversion Table:

D	Q_n	Q_{n+1}	J	K
0	0	0	0	X
0	1	0	X	1
1	0	1	1	X
1	1	1	X	0

For J:

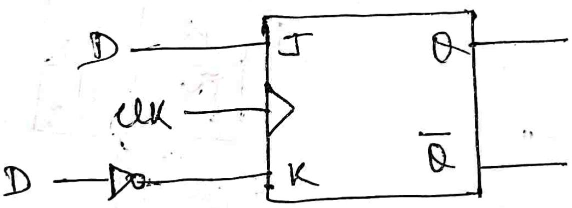
D \ Q_n	$\bar{Q}_n$	Q_n
$\bar{D}$	0 (0)	X (1)
D	1 (2)	X (3)

$\boxed{J = D}$

For K:

D \ Q_n	$\bar{Q}_n$	Q_n
$\bar{D}$	X (0)	1 (1)
D	X (2)	(3)

$\boxed{K = \bar{D}}$



Soln (iii): JK-FF to T-FF:

* Characteristic Table of T-FF:

T	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	1
1	1	0

* Excitation Table of JK FF:

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

* Conversion Table:

T	Q_n	Q_{n+1}	J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	X
1	1	0	X	1

For J:

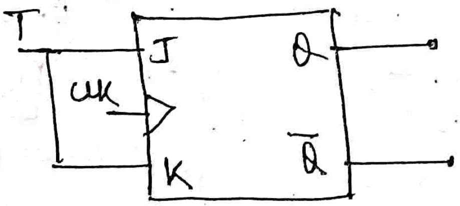
T \ Q_n	$\bar{Q}_n$	Q_n
$\bar{T}$	(0)	X (1)
T	1 (2)	X (3)

$\boxed{J = T}$

For K:

T \ Q_n	$\bar{Q}_n$	Q_n
$\bar{T}$	X (0)	(1)
T	X (2)	1 (3)

$\boxed{K = T}$



(Q-4) (i) T-FF to SR-FF:

Soln: * Characteristic Table of SR-FF:

S	R	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1

1	1	0	X
1	1	1	X

* Excitation Table of T-FF :

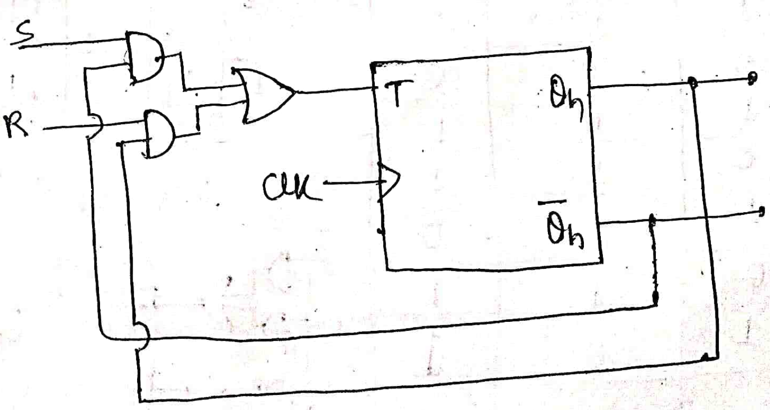
Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

* Conversion Table :

S	R	Q_n	Q_{n+1}	T
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	1
1	0	1	1	0
1	1	0	X	X
1	1	1	X	X

S \ RQ_n	00	01	11	10
0			1	
1	1		X	X

$T = S\overline{Q_n} + RQ_n$



(ii) T-FF to JK-FF :

Soln:

* Characteristic Table of JK-FF :

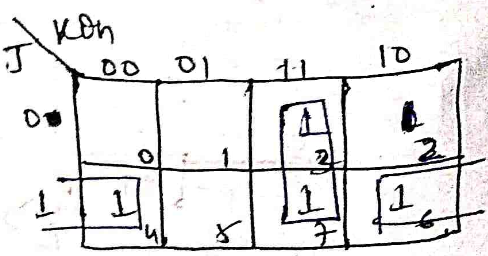
J	K	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

* Excitation Table of T-FF :

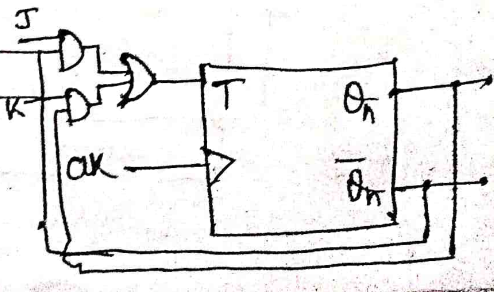
Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

* Conversion Table :

J	K	Q_n	Q_{n+1}	T
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	1
1	0	1	1	0
1	1	0	1	1
1	1	1	0	1



$T = J\overline{Q_n} + KQ_n$



(iii) T-FF to D-FF :

Solⁿ: * Characteristic Table of D-FF :

D	Q_n	Q_{n+1}
0	0	0
0	1	0
1	0	1
1	1	1

* Excitation Table of T-FF :

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

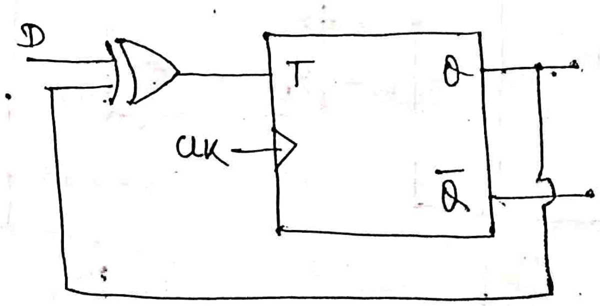
* Conversion Table :

D	Q_n	Q_{n+1}	T
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

D \ Q_n	$\bar{Q}_n$	Q_n
$\bar{D}$	0	$\boxed{1}$ 1
D	$\boxed{1}$ 2	3

$T = D\bar{Q}_n + \bar{D}Q_n$

or $T = D \oplus Q_n$



Imp

* Analysis of clock sequential circuit :

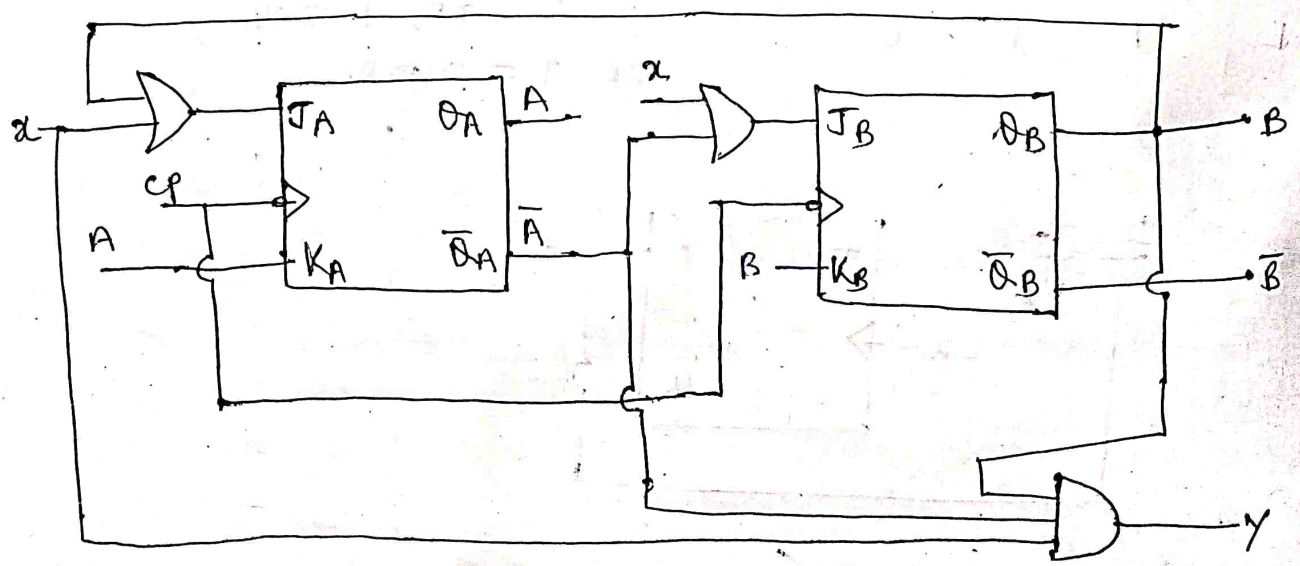
- The analysis of a sequential circuit consists of obtaining a table or a diagram for the time sequence of I/P's, O/P's and internal states (Q_n, Q_{n+1}).

- A state table and state diagram are then presented to describe the behaviour of the sequential circuit.

* Steps for solving :

- Find and write FF I/P's (FF I/P eq^n and O/P for ckt.)
- Write characteristic eq^n for given FF.
- Find state equations (transition eq^n).
- Find state table (transition table).
- Draw the state diagram.

(Ques 1) Derive the state table and state diagram for the sequential ckt shown in figure.



- FF I/P eqn:

$$J_A = x + B, \quad K_A = A, \quad J_B = x + \bar{A}, \quad K_B = B$$

- characteristic eqn of JK-FF:

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n$$

- State eqn putting the value of FF i/p eqn in characteristic eqn:

$$A_{n+1} = J_A \bar{Q}_A + \bar{K}_A Q_A$$
$$= (x+B)\bar{A} + \bar{A} \cdot A \qquad \{\because Q_A = A\}$$
$$= (x+B)\bar{A}$$

$$B_{n+1} = J_B \bar{Q}_B + \bar{K}_B Q_B$$
$$= (x+\bar{A})\bar{B} + \bar{B} \cdot B \qquad \{\because Q_B = B\}$$
$$= (x+\bar{A})\bar{B}$$

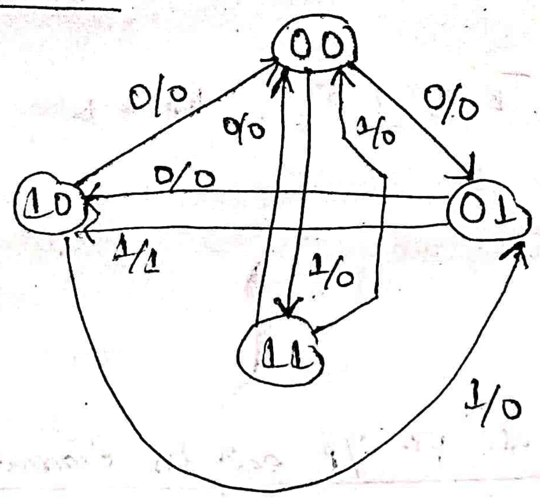
- O/P eqn of Y:

$$Y = x\bar{A}B$$

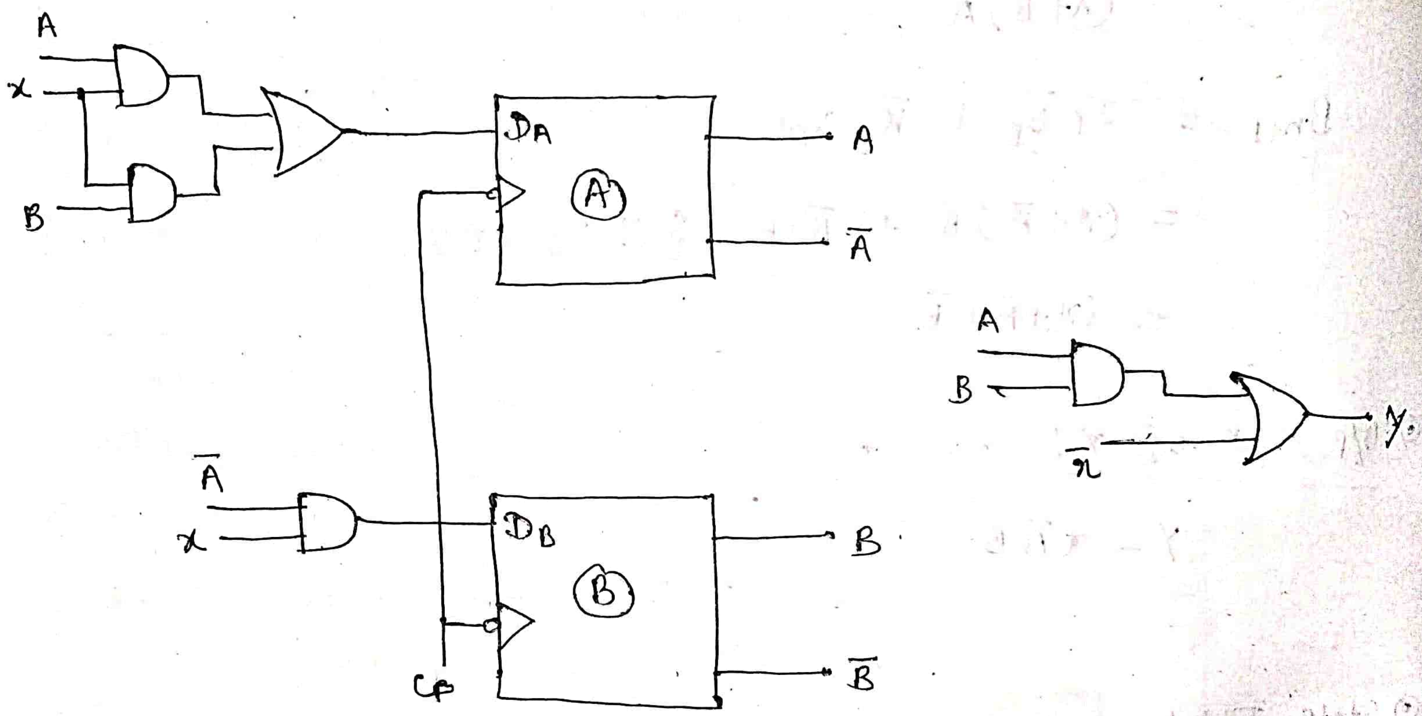
- State Table:

P.S A	P.S B	i/p x	N.S A+	N.S B+	O/P Y
0	0	0	0	1	0
0	0	1	1	1	0
0	1	0	1	0	0
0	1	1	1	0	1
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	0	0
1	1	1	0	0	0

⊙ State Diagram!



(Q-2) Derive the state table and state diagram for the sequential ckt shown in fig.



⊙ FF I/P eqn!

$$D_A = Ax + Bx \quad , \quad D_B = \bar{A}x$$

⊙ characteristic eqn of D-Flip-Flop!

$$Q_{n+1} = D$$

⊙ State eqn putting the value of FF I/p eqn in characteristic eqn:

$$A_{n+1} = D_A$$

$$A_{n+1} = Ax + Bx$$

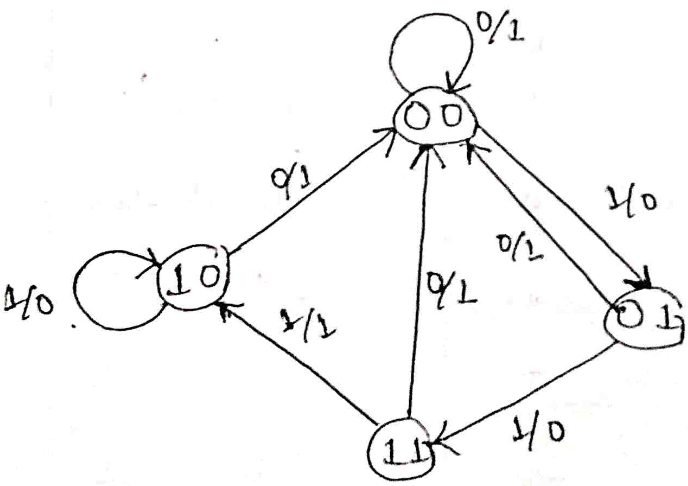
$$B_{n+1} = \overline{A}x$$

⊙ O/p eqn of Y:

$$Y = AB + \overline{x}$$

⊙ State Table:

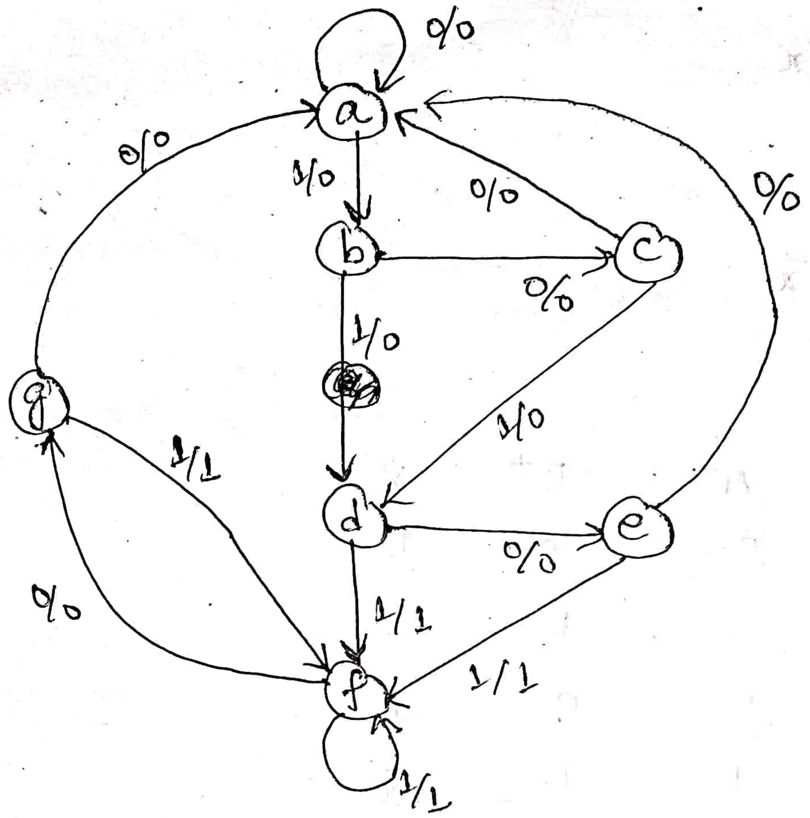
A	B	x	A+	B+	Y
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	1



Imp

* State Reduction :

Ques: Draw the reduced state table and state diagram for the given state diagram.



* State Table :

Present State	Next State $x=0$	Next State $x=1$	O/P $x=0$	O/P $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1

Since, present state e and g go to the next state a and f and output are 0 & 1 for $x=0$ & $x=1$. So, states e & g are equivalent then row present state g is removed and g is replaced by e because g and e are equivalent row.

* Reduced State Table:

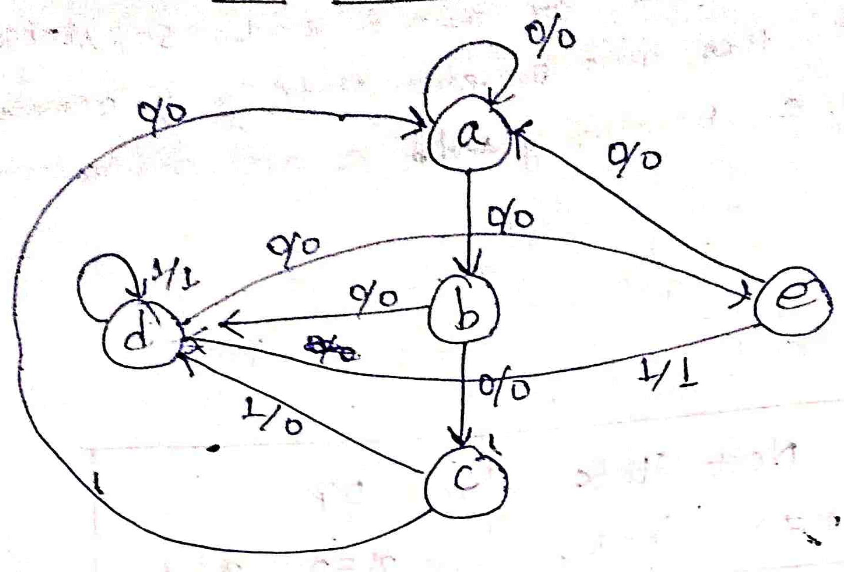
Present State	Next State $x=0$	Next State $x=1$	O/P $x=0$	O/P $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	e	f	0	1

Since, the present state d & f go to the next state e & f and output are 0 & 1 for $x=0$ & $x=1$. So state d & f are equivalent then row present state f is removed and f replaced by d because d & f are equivalent.

* Reduced State Table:

Present State	Next State $x=0$	Next State $x=1$	O/P $x=0$	O/P $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

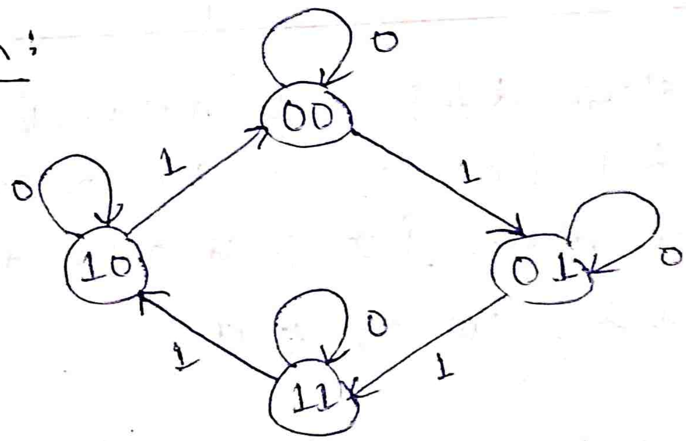
* **Reduced State Diagram:**



* **Design Procedure:**

(Q) Design a sequential ckt with two D-FF, A & B and one I/P x. When x=0 the state of the ckt remains the same. When x=1, the ckt passes through the state transition from (00) to (01) to (11) to (10) back to (00) and repeats.

* **State Diagram:**



* **State Table:**

Present State A	B	I/P x	Next State A^+	B^+	FF I/Ps D_A	D_B
0	0	0	0	0	0	0
0	0	1	0	1	0	1
0	1	0	0	1	0	1
0	1	1	1	1	1	1
1	0	0	1	0	1	0
1	0	1	0	0	0	0
1	1	0	1	1	1	1
1	1	1	1	0	1	0

* Excitation Table of D-FF :

Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

For $D_A = A\bar{x} + Bx$

A \ Bx	$\bar{B}\bar{x}$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$	0		1	
A	1		1	1

For $D_B = \bar{A}x + B\bar{x}$

A \ Bx	$\bar{B}\bar{x}$	$\bar{B}x$	Bx	$B\bar{x}$
$\bar{A}$		1	1	1
A				1

e I/p x.
When
(00)