

UNIT - 1

Introduction :- 'Computer' is a machine that can store and process information. Most computers rely on a binary system that uses two variables 0 and 1 to complete tasks such as storing data calculating algorithms and displaying information.

Organization :- Group of people who work together and to reach a goal by using proper System.

System :- A set of things working together to accomplish particular Goal.

Ex :- School system, college system and railway system.

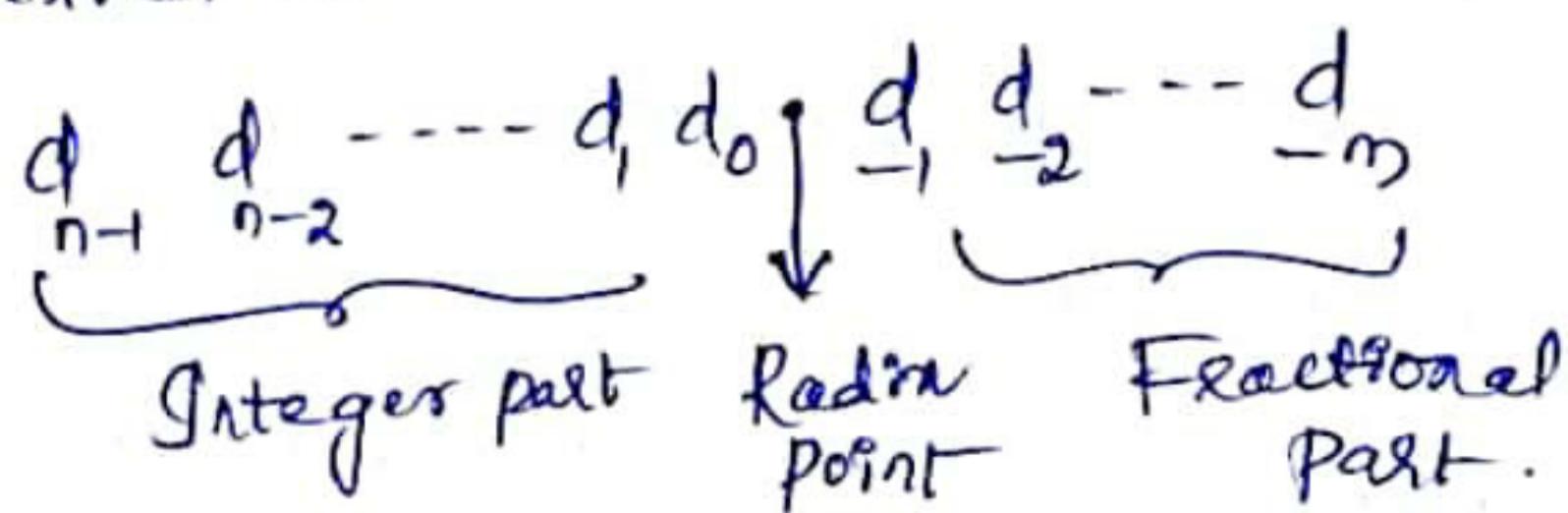
Number system :- Represent data in digital form.

Binary Number system :- A number is made up of a collection of digits and it has two parts.

(a) Integer part

(b) Fractional part

Digits are separated by a radix point (.). The number is represented as



Number systems are classified as

- (a) Binary Number System
- (b) Decimal Number System
- (c) Octal Number System
- (d) Hexadecimal Number System.

- (a) The binary number system is a radix-2. This is represented in terms of 0 and 1. The radix point is known as the binary point and 0 and 1 is a binary bits.
- (b) The decimal number system is a radix-10. These are 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9 (0-9). The radix point is known as decimal point.
- (c) Octal number system is a radix 8. They are 0, 1, 2, 3, 4, 5, 6, and 7 (0-7). The radix point is known as octal point.
- (d) The hexadecimal number system is a radix 16. They are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15. The radix point is known as hexadecimal point. The decimal equivalent of A, B, C, D, E and F are 10, 11, 12, 13, 14 and 15.

Radix :- Number of symbols presented in a respective number system is called Radix.

Ex :- 1 0 1 1 0 1 0
(Most Significant MSB) (Least Significant bit)
 ↴

Decimal number system to Binary number system conversion :-

$$\textcircled{1} \quad (25)_{10} = (?)_2$$

2	25
2	12 - 1
2	6 - 0
2	3 - 0
2	1 - 1

From Bottom to top

$$\therefore (25)_{10} = (11001)_2$$

(Successive Division method)

$$\textcircled{2} \quad (72)_{10} = (?)_2$$

2	72
2	36 - 0
2	18 - 0
2	9 - 0
2	4 - 1
2	2 - 0
	1 - 0

From
Bottom
to
Top

$$\therefore (72)_{10} = (1001000)_2$$

⇒ Fractional part :-

$$\textcircled{3} \quad (0.25)_{10} = (?)_2$$

Top to
Bottom

$$0.25 \times 2 = 0.5 \rightarrow 0$$

$$0.5 \times 2 = 1.0 \rightarrow 1$$

$$0.0 \times 2 = 0.0 \rightarrow 0$$

$$0.0 \times 2 = 0 \rightarrow \text{ignore it}$$

Whenever we get repeated
numbers just ignore it

$$\therefore (0.25)_{10} = (0.010)_2$$

$$\textcircled{4} \quad (0.8125)_{10} = (?)_2$$

$$0.8125 \times 2 = 1.6250 \rightarrow 1$$

$$0.6250 \times 2 = 1.250 \rightarrow 1$$

$$0.250 \times 2 = 0.50 \rightarrow 0$$

$$0.50 \times 2 = 1.0 \rightarrow 1$$

$$0.0 \times 2 = 0.0 \rightarrow 0$$

Top to Bottom

$$\therefore (0.8125)_{10} = (0.11010)_2$$

$$\textcircled{5} \quad (10.625)_{10} = (?)_2$$

2	10
2	5-0
2	2-1
2	1-0
	(1010)

$$0.625 \times 2 = 1.250 \rightarrow 1$$

$$0.250 \times 2 = 0.50 \rightarrow 0$$

$$0.50 \times 2 = 1.0 \rightarrow 1$$

$$0.0 \times 2 = 0.0 \rightarrow 0$$

(1010)

$$\therefore (10.625)_{10} = (1010.1010)_2$$

$$\textcircled{6} \quad (25.125)_{10} = (?)_2$$

2	25
2	12-1
2	6-0
2	3-0
	1-1

$$0.125 \times 2 = 0.250 \rightarrow 0$$

$$0.250 \times 2 = 0.50 \rightarrow 0$$

$$0.5 \times 2 = 1.0 \rightarrow 1$$

$$0.0 \times 2 = 0.0 \rightarrow 0$$

(0010)

$$\therefore (25.125)_{10} = (11001.0010)_2$$

Binary number system to Decimal number system conversion :-

$$\textcircled{1} \quad (1101)_2 = (?)_{10}$$

(Successive multiplication method)

1	1	0	1
2^3	2^2	2^1	2^0

$$1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 0 + 1 = 13 \quad \therefore (1101)_2 = (13)_{10}$$

$$\textcircled{2} \quad (101011)_2 = (?)_{10}$$

1	0	1	0	1	1
2^5	2^4	2^3	2^2	2^1	2^0

$$1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 32 + 0 + 8 + 0 + 2 + 1 \\ = 43$$

$$\therefore (101011)_2 = (43)_{10}$$

$$\textcircled{3} \quad (101.10)_2 = (?)_{10}$$

1	0	1	.	1	0
2^2	2^1	2^0	$\bar{2}^1$	$\bar{2}^2$	

$$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times \bar{2}^1 + 0 \times \bar{2}^2 \\ = 4 + 0 + 1 + 0.5 + 0 \\ = 5 + 0.5 = 5.5$$

$$\therefore (101.10)_2 = (5.5)_{10}$$

$$\textcircled{4} \quad (11010.010)_2 = (?)_{10}$$

1	1	0	1	0.	0	1	0
2^4	2^3	2^2	2^1	2^0	$\bar{2}^1$	$\bar{2}^2$	$\bar{2}^3$

$$= 1 \times 2^4 + 1 \times 2^3 + 0 \times \bar{2}^2 + 1 \times 2^1 + 0 \times \bar{2}^0 + 0 \times \bar{2}^1 + 1 \times \bar{2}^2 + 0 \times \bar{2}^3 \\ = 16 + 8 + 0 + 2 + 0. + 0 + 0.25 + 0 \\ = 26 + 0.25 = 26.25$$

$$\therefore (11010.010)_2 = (26.25)_{10}$$

$$\textcircled{5} \quad (10010)_2 = (?)_{10}$$

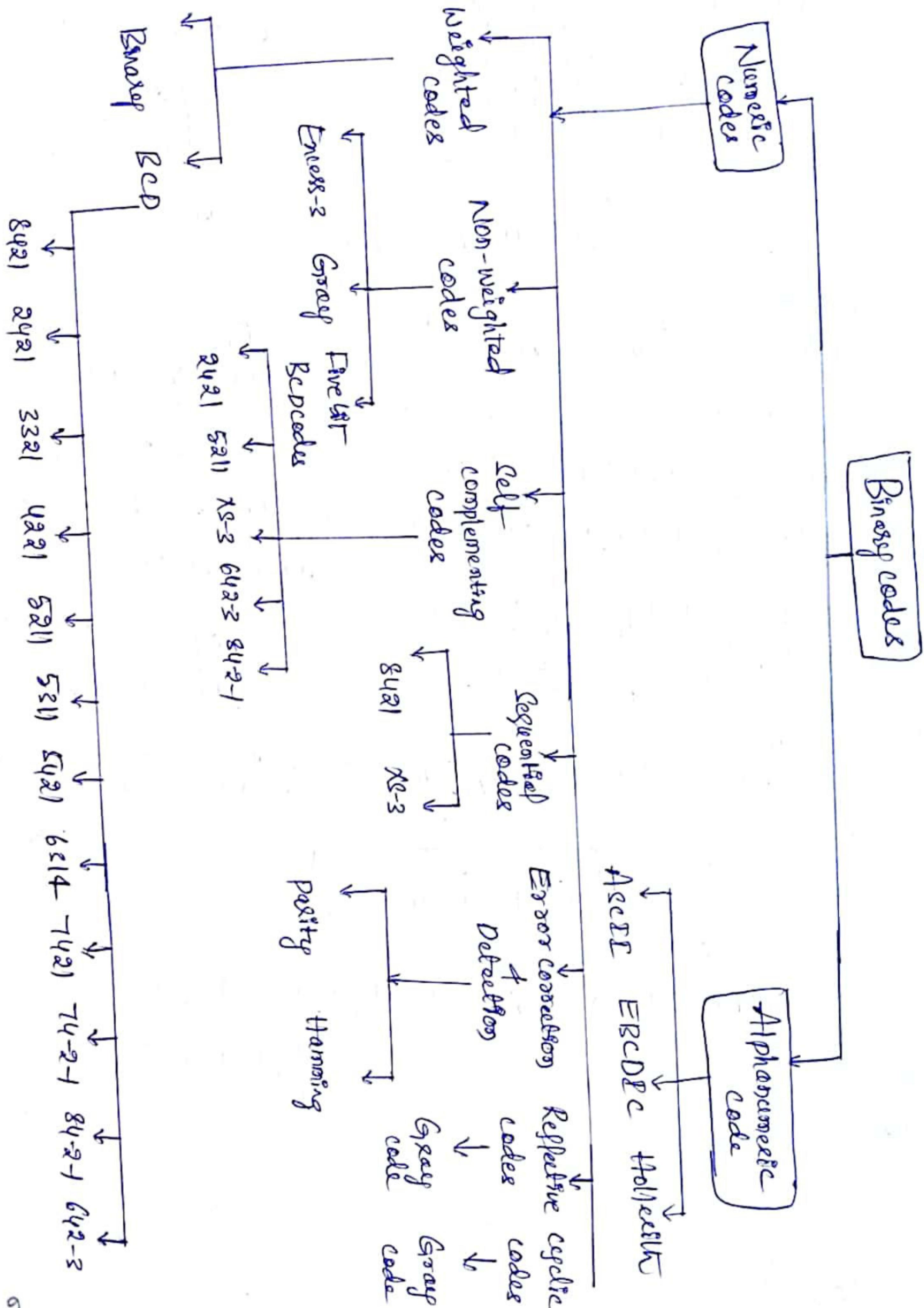
1	0	0	1	0
2^4	2^3	2^2	2^1	2^0

$$1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \\ = 16 + 0 + 0 + 2 + 0 = (18)_{10}$$

$$\textcircled{6} \quad (011.01)_2 = (?)_{10}$$

0	1	1.	0	1
2^2	2^1	2^0	$\bar{2}^1$	$\bar{2}^2$

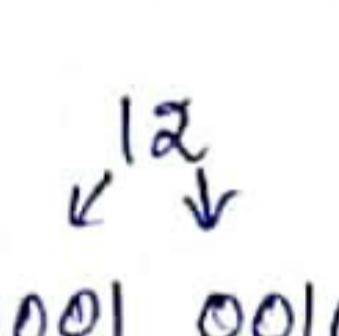
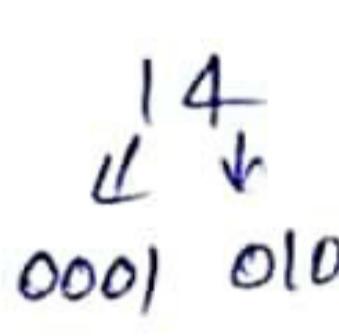
$$0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times \bar{2}^1 + 1 \times \bar{2}^2 \\ = 0 + 2 + 1. + 0 + 0.25 = (3.25)_{10}$$



→ Binary coded decimal Numbers (BCD) :-

- BCD code uses four bits to represents the decimal numbers. i.e (0-9). Each single decimal number can be represented by a four bit pattern.
- 8421 is also known as Natural BCD.
- ex:- 8421, 2421, 3321, 4221, 5211, 5421, 6314, 7421
84-2-1, 642-3.

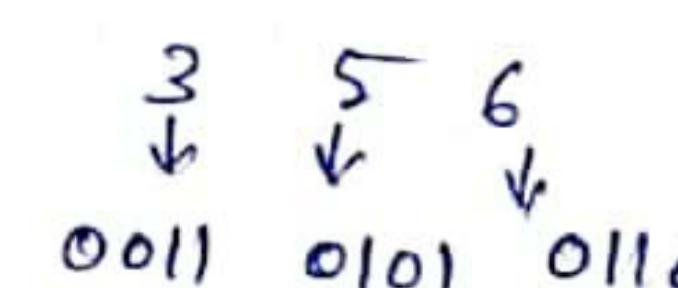
→ Representation of BCD code

- Ex:- ①  (Each digit is represented by four bits)
- Ex:- ②  (Each digit is indicated by group of 4-bits)

There are 6 unused states in BCD ($1010, 1011, 1100, 1101, 1110, 1111$)
 $\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow$
 10 11 12 13 14 15.

<u>Decimal number</u>	<u>BCD</u>
14	0001 0100
234	0010 0011 0100
239.56	0010 0011 1001 . 0101 0110
653.96	0110 0101 0011 . 1001 0110

* Represent 356 in BCD format

ans:- 

⇒ Pure binary representation :-

Ex:- 14 —

8	4	2
1	1	1
Only 4-bits		

Ex:- 12 —

8	4	2
1	1	0
Only 4-bits		

In BCD

1	4
↓	↓
0001	0100
8-bits required	

1	2
↓	↓
0001	0010
8-bits required.	

Q) To represent 12 in binary what are the minimum no. of digits we required.

Binary →

8	4	2
1	1	0
Only 4-bits		

Q) To represent 12 in BCD what are the minimum no. of digits we required.

BCD →

1	2
0001	0010
8-bits.	

Note :-

From the above concept, we can conclude one thing that BCD is simple to represent decimal numbers but some times it takes more no. of bits. so, it occupies memory. Arithmetic operations are more complex than the binary.

Ex:-

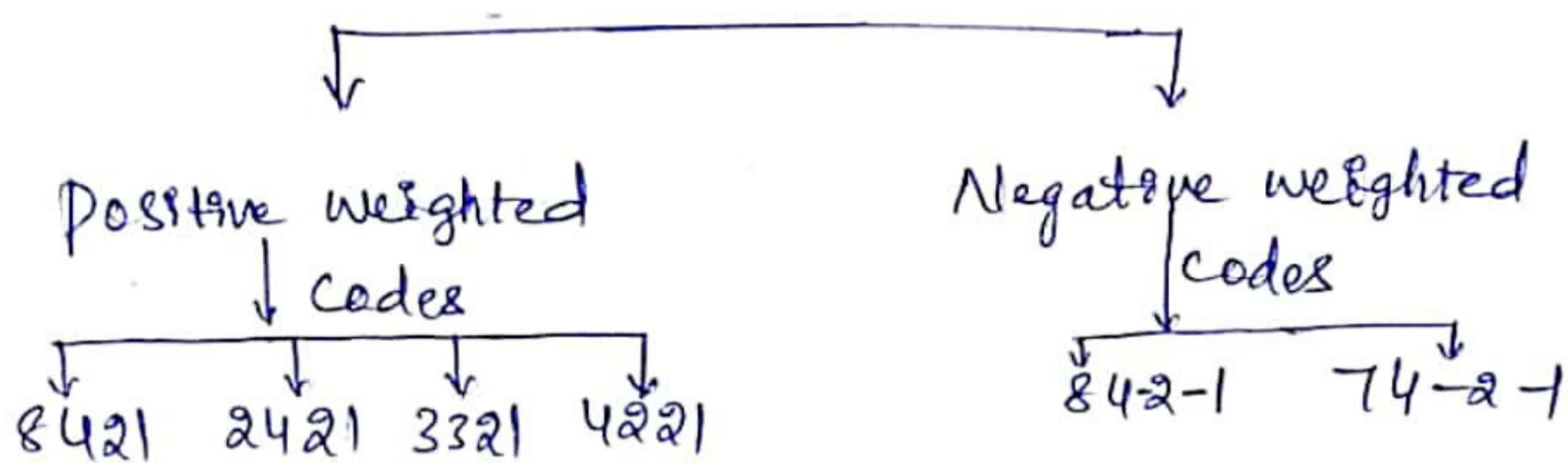
92 in Binary representation

64	32	16	8	4	2
1	0	1	1	1	0

92 in BCD

1	001	0010
↓	↓	↓

→ Weighted codes :- The weighted codes are those which obey the position weighting principle. Each position of the number represents a specific weight.



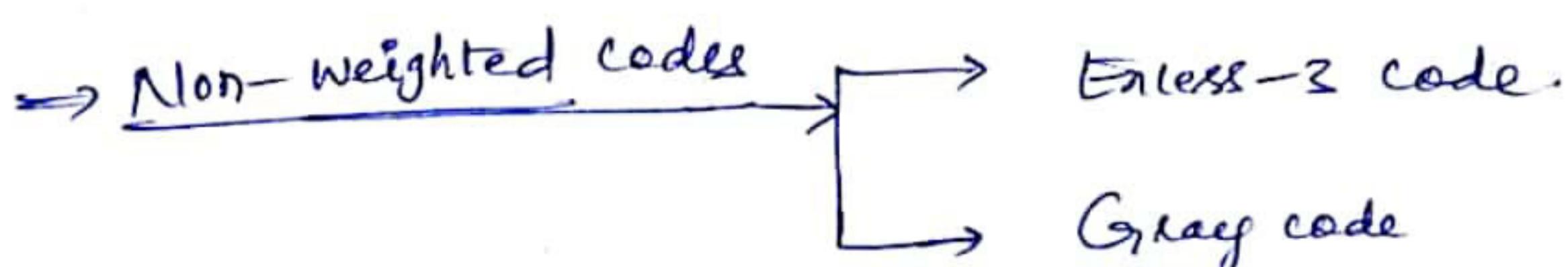
Ex:-

Decimal	8421	2421	3321	4221	
0	0000	0000	0000	0000	
1	0001	0001	0001	0001	
5	0101	{ 1010 0101 }	{ 1010 0110 }	{ 1001 0111 }	
7	0111	{ 1101 1011 }	1101	{ 1101 1011 }	

All these are positive weighted codes

Ex:-

Decimal	84-2-1	74-2-1
0	0 0 0 0	0 0 0 0
5	1 0 1 1	1 0 1 0
7	1 0 0 1	1 0 0 0
9	1 1 1 1	1 1 1 0



→ Self-complementing code :- It is said to be self-complementing if the code word of the 9's complement of N i.e. $9-N$ can be obtained from the code word of N by interchanging all the 0's and 1's.

Ex:- 2421, 5211, 642-3, 84-2-1 & Excess-3.

$$\begin{array}{cccc} 2+4+2+1 & 5+2+1+1 & 6+4+2-3 & 8+4-2-1 \\ =9 & =9 & =9 & =9 \end{array}$$



Ex:①

5 in Excess-3 is $5+3=8 = \begin{smallmatrix} 8 & 4 & 2 & 1 \\ & & & \downarrow \end{smallmatrix}$

It can be obtained by adding 3 to that binary number

$1000 \xrightarrow{\substack{\text{1's} \\ \text{complement}}} 0111$ [Ex-3 code of decimal number 4]

4 is the 9's complement of 5 ($9-5=4$)

∴ It is a self-complementing code.

Ex:②

4 in $\begin{smallmatrix} 2 & 4 & 2 & 1 \\ & & & \downarrow \end{smallmatrix}$ = 0100

$0100 \xrightarrow{\text{1's complement}} 1011$ (This is 2421 code for decimal number 5)

5 is the 9's complement of 4.

∴ It is a self-complementing code.

Ex:③

BCD code for 6 is $\begin{smallmatrix} 8 & 4 & 2 & 1 \\ & & & \downarrow \end{smallmatrix}$ 0110

$0110 \xrightarrow{\text{1's complement}} 1001$ (This is BCD code for decimal number 9)

9 is not the 9's complement of 6.

∴ BCD is not a self-complementing code.

Ex:④ Binary (8421) code for 7 is 0111

$0111 \xrightarrow{\text{1's complement}} 1000 = 8$ (Binary code 8)
q-decimal

8 is not the 9's complement of 7

∴ Binary code (8421) is not a self-complementing code.

⇒ Cyclic codes :- cyclic codes are those in which each successive code word differs from the preceding one in only 1-bit position. cyclic codes are also called as unit distance codes ex:- Gray code.

* Gray code is also called as Reflective code.

Reflective code means in 8421 code 0-7 is the mirror

image of 8-15. Gray code is not a sequence code.

That's why we can't do arithmetic operation by using

Ex:- this code.

<u>Decimal No.</u>	<u>Binary</u>	<u>Gray</u>
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

① convert $(1010)_2$ to gray code

$$\begin{array}{l} \text{Sof} \\ \hline \text{ans } \underline{\underline{1111}} \end{array}$$

② convert $(0110)_2$ to gray code

$$\begin{array}{l} 0110 \\ \downarrow \\ \text{ans } \underline{\underline{0101}} \end{array}$$

X-OR Truth Table

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

→ Alphanumeric codes :- These are the codes which represent alphanumeric information i.e. letters of the alphabet and decimal numbers as a sequence of 0's and 1's.

Eg:- ASCII, EBCDIC codes

- ASCII → American standard code for information interchange
 - EBCDIC → Extended binary coded decimal interchange code.
 - Alphanumeric codes consists of numbers as well as alphabetic characters.
 - It contains 26 Alphabets with Capital & Small letters, numbers (0-9), punctuation marks and other symbols.
 - ASCII code is a 7-bit code and more commonly used worldwide.
 $\therefore 2^7 = 128$ symbols are used to represent info's.
 - EBCDIC code is a 8 bit code and used in large ~~PRM~~ computers.
 $\therefore 2^8 = 256$ symbols are used
- International
Business
Machine.

32 → Space	42 → *	61 → =
33 → !	43 → +	62 → >
34 → "	44 → ,	63 → ?
35 → #	45 → -	64 → @
36 → \$	46 → .	65-90 (A-Z) Capital letters
37 → %	47 → /	91 → [
38 → &	48 → 57 → (0-9)	92 → \
39 → C	58 → :	93 →]
40 → >	59 → ;	94 → ^
	60 → <	95 → _ (Underscore)

96 → `

97-122 (a-z) small letters

0-31 → character control

123 → {

124 → | (vertical bar)

125 → }

126 → ~

Note :- Binary - Decimal - ASCII (Basic phenomena to do
ASCII problems).

Ex :-

10010011000001	1001101
↓	↓
73	65
↓	↓
Z	A

Ex :-

1001010	1001100
↓	↓
J	L
↓	↓
J	L

→ ASCII codes are used in micro computer (or) personal computer

→ EBCDIC codes are used in large computers.

→ Hollerith code :- This code is used in system to

represent alphanumeric information.

→ It consists of 80 columns and 12 rows

→ It is a 12-bit code.

Ex :-

1010111	Space	1000010	Space	1000111	Space	1000011
↓	↓	↓	↓	↓	↓	↓
87	32	66	32	74	32	67
↓	↓	↓	↓	↓	↓	↓
w	B	G				C

→ Error correcting codes :- Codes which allow error detection and correction are called Error correcting codes.

Eg :- Hamming code.

→ Hamming code is a specific type of error correcting code that allows the detection and correction of single bit transmission errors. Hamming code algorithm can solve only single bit issues. These are used in satellite communication.

Ex:- Encode the data (or) message bits 0011 into the 7-bit even parity Hamming code.

Sol Given message = 0011

Number of message bits M = 4

Number of parity bits required is calculated using the formula

$$2^P \geq M+P+1$$

$$2^P \geq 4+P+1$$

$$2^3 \geq 4+3+1$$

$$8 \geq 8$$

Number of parity bits P = 3

Total no of bits M+P = 4+3 = 7

Decimal no	2^2	2^1	2^0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

7 6 5 4 3 2 1
 2^2 2^1 2^0

m_4 m_2 m_1 P_3 m_1 P_2 P_1

$$P_1 = 1 \ 3 \ 5 \ 7 = P_1 \text{110}$$

= 0110 ($\because P_1 = 0$; to maintain the even parity)

$$\therefore P_1 = 0.$$

$$P_2 = 2, 3, 6, 7 = P_2 100 \\ = 1100 \quad \text{To become the even parity } (\because P_2=1) \\ \therefore P_2 = 0$$

$$P_3 = 4, 5, 6, 7 = P_3 100 \quad \text{to become the even parity } (\because P_3=1) \\ = 1100 \\ \therefore P_3 = 1$$

Error position = By combining the parity bits

$$P_3 P_2 P_1 = \underline{P_3} \underline{P_2} \underline{P_1} = 0110 = (6)_{10}$$

Error is located at 3rd position

Total message bits = 0011110

After correcting = 0111110.

Sol:

Generate Hamming code for the message 110

Sol:

$$2^P \geq p+m+1$$

p = parity bits

$$2^P \geq p+4+1$$

m = message bits

$$2^P \geq p+5$$

p should be atleast 3 to satisfy the condition

$$2^3 \geq 3+5 \therefore 8 \geq 8 \text{ (true)}$$

1	2	3	4	5	6	7
2^0	2^1	2^2				
P_1	P_2	m_1	P_3	m_2	m_3	m_4
P_1	P_2	1	P_3	1	1	0

(The code may be any length the process will be same)

(For even parity)

$$P_1 \Rightarrow 1, 3, 5, 7 \rightarrow P_1 110 \rightarrow 0110 \quad (P_1=0)$$

$$P_2 \Rightarrow 2, 3, 6, 7 \rightarrow P_2 110 \rightarrow 0110 \quad (P_2=0)$$

$$P_3 \Rightarrow 4, 5, 6, 7 \rightarrow P_3 110 \rightarrow 0110 \quad (P_3=0)$$

Total message bits

$$= 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0$$

Odd parity :-

$$P_1 \rightarrow 1, 3, 5, 7 \rightarrow P_1 110 \rightarrow 1110 \ (P_1 = 1)$$

$$P_2 \rightarrow 2, 3, 6, 7 \rightarrow P_2 110 \rightarrow 1110 \ (P_2 = 1)$$

$$P_3 \rightarrow 4, 5, 6, 7 \rightarrow P_3 110 \rightarrow 1110 \ (P_3 = 1)$$

Total message bits

$$= P_1 \ P_2 \ m_1 \ P_3 \ m_2 \ m_3 \ m_4$$

$$= 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0$$

\Rightarrow Error correction in Hamming code

Q3 A (7,4) hamming code is received as 1110000 determine the corrected code when even and odd parity.

Sol

$$\begin{matrix} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ & | & | & | & 0 & 0 & 0 & 0 \end{matrix}$$

To ensure that error is there are not

$E_1 \rightarrow 1, 3, 5, 7 \rightarrow 1100 \rightarrow$ to make it even parity
 $E_1 = 0$

(even parity) $E_2 \rightarrow 2, 3, 6, 7 \rightarrow 1100 \Rightarrow E_2 = 0$

$E_3 \rightarrow 4, 5, 6, 7 \rightarrow 0000 \Rightarrow E_3 = 0$

Error = $E_1 E_2 E_3 = 000$ (0th position)

odd parity

$E_1 \rightarrow 1, 3, 5, 7 \rightarrow 1100 \rightarrow E_1 = 1$. (to make it odd parity)

$E_2 \rightarrow 2, 3, 6, 7 \rightarrow 1100 \rightarrow E_2 = 1$ (to make it odd parity)

$E_3 \rightarrow 4, 5, 6, 7 \rightarrow 0000 \rightarrow E_3 = 1$

Error = $E_1 E_2 E_3 = 111$ ^ 1st position error is there
 connected code may be 1110001

Q Determine the single error-

correcting code for the information

Code 10111 for odd parity

Sol Given message bit
 $m = 10111$

By using trial and error method
 we should find parity bits

$$2^P \geq m + p + 1$$

$$\therefore P = m$$

$$2^1 \geq 5 + 1 + 1$$

$$\text{Let } P = 1$$

$$2 \geq 7 \times$$

$$2^2 \geq 5 + 2 + 1 \quad \text{Let } P = 2$$

$$2^2 \geq 8 \times$$

$$2^3 \geq 5 + 3 + 1 \quad \text{Let } P = 3$$

$$8 \geq 9 \times$$

$$2^4 \geq 5 + 4 + 1 \quad \text{Let } P = 4$$

$$16 \geq 10 \checkmark$$

So, we need 4 parity bits.

We should take parity bits

always powers of 2.

$$2^0 = 1; 2^1 = 2, 2^2 = 4, 2^3 = 8$$

$$2^4 = 16; 2^5 = 32 \text{ and so on.}$$

Find the value of the parity

1 2 3 4 5 6 7 8 9

$P_1 P_2 m_1 P_2 m_2 m_3 m_4 P_4 m_5$

$P_1 P_2 1 P_3 1 1 0 P_4 1$

$$\therefore m = 10111 \\ m_5 m_4 m_3 m_2 m_1$$

Step 1
 ✓

Bit destination	m_5	P_8	m_4	m_3	m_2	P_4	m_1	P_2	P_1
Bit Location	9	8	7	6	5	4	3	2	1
Information bits	(001)	(000)	(011)	(010)	(0101)	(0100)	(0011)	(0010)	(000)
Parity bits	1	?	0	1	1	?	1	?	?
Received code	1	0	0	1	1	1	1	1	0

$$P_1 \rightarrow P_1 m_1 m_2 m_3 m_4 m_5 = P_1 1101$$

To make it become odd
 we kept $P_1 = 0$; so $P_1 = 0$

$$P_2 \rightarrow P_2 m_1 m_3 m_4 = P_2 110 \text{ To make it become odd}$$

we kept $P_2 = 1$; so $P_2 = 1$

$$P_4 \rightarrow P_4 m_2 m_3 m_4 = P_4 110 \text{ To make it odd}$$

we kept $P_4 = 1$; $P_4 = 1$

$$P_8 \rightarrow P_8 m_5 = P_8 1 \text{ To make it odd}$$

we kept $P_8 = 0$; $P_8 = 0$

Error position $P_8 P_4 P_2 P_1$

$$= \underline{0} \underline{1} \underline{0} = 6^{\text{th}} \text{ position}$$

1111 is 9 bit hamming code

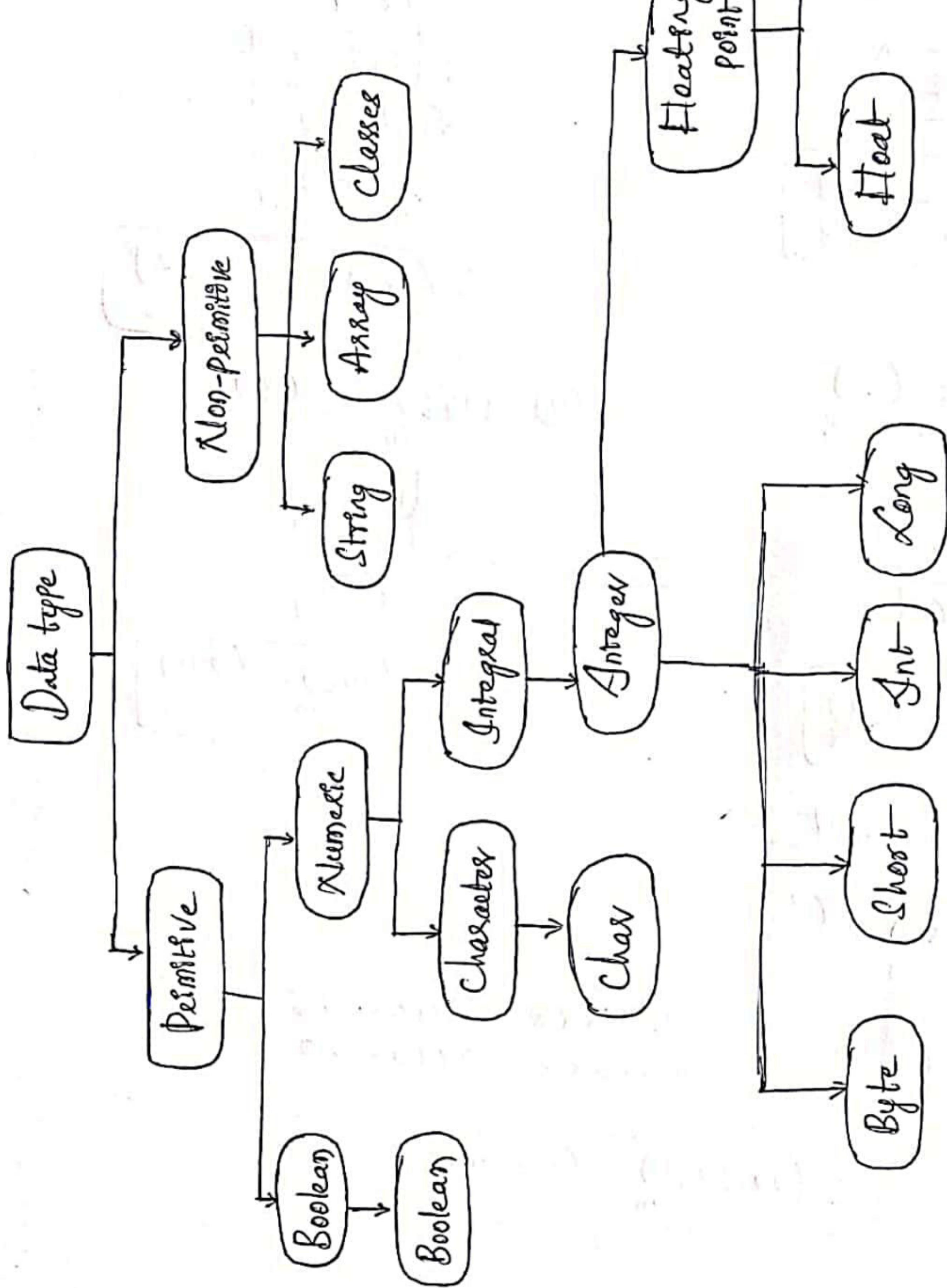
$$\checkmark \quad \begin{array}{ccccccccc} 9 & 8 & 7 & 6 & 5 & 4 & 1 & 2 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \end{array}$$

so 9 is 6th position

$$\checkmark \quad \boxed{100011110}$$

⇒ Data Representation :-

⇒ Data types :-



→ Decimal to octal :-

$$\textcircled{1} \quad (20)_{10} = (?)_8$$

$$8 \left| \begin{array}{r} 20 \\ -2-4 \\ \hline \end{array} \right.$$

$$\therefore (20)_{10} = (24)_8$$

$$\textcircled{2} \quad (1234)_{10} = (?)_8$$

$$\begin{array}{r} 8 \left| \begin{array}{r} 1234 \\ -154-2 \\ \hline 19-2 \\ \hline 2-3 \end{array} \right. \end{array}$$

$$\therefore (1234)_{10} = (2322)_8$$

$$\textcircled{3} \quad (183)_{10} = (?)_8$$

$$\begin{array}{r} 8 \left| \begin{array}{r} 183 \\ -22-7 \\ \hline 2-6 \end{array} \right. \end{array}$$

$$\therefore (183)_{10} = (267)_8$$

$$\textcircled{4} \quad (145)_{10} = (?)_8$$

$$\begin{array}{r} 8 \left| \begin{array}{r} 145 \\ -18-1 \\ \hline 2-2 \end{array} \right. \end{array}$$

$$\therefore (145)_{10} = (221)_8$$

→ Fractional part :-

$$\textcircled{1} \quad (27.625)_{10} = (?)_8$$

$$8 \left| \begin{array}{r} 27 \\ -3-3 \\ \hline \end{array} \right.$$

$$\begin{aligned} 0.625 \times 8 &= 5.000 \rightarrow 5 \\ 0.000 \times 8 &= 0.000 \rightarrow 0 \end{aligned}$$

$$\therefore (27.625)_{10} = (33.50)_8$$

$$\textcircled{2} \quad (3287.5100098)_{10} = (?)_8$$

Sol

Integer part

$$8 \overline{)3287} \\ 8 \overline{)410} -7 \\ 8 \overline{)51} -2 \\ 6-3$$

$$0.5100098 \times 8 = 4.0800 \rightarrow 4 \\ 0.0800 \times 8 = 0.640 \rightarrow 0 \\ 0.640 \times 8 = 5.125 \rightarrow 5 \\ 0.125 \times 8 = 1.000 \rightarrow 1$$

$$\therefore (3287.5100098)_{10} = (6327.4051)_8$$

$$\textcircled{3} \quad (20.5)_{10} = (?)_8$$

$$8 \overline{)20} \\ 2-4$$

Fractional part

$$0.5 \times 8 = 4.0 \rightarrow 4 \\ 0.0 \times 8 = 0.0 \rightarrow 0$$

$$\therefore (20.5)_{10} = (24.40)_8$$

$$\textcircled{4} \quad (60.7)_{10} = (?)_8$$

$$8 \overline{)60} \\ 7-4$$

$$0.7 \times 8 = 5.6 \rightarrow 5 \\ 0.6 \times 8 = 4.8 \rightarrow 4 \\ 0.8 \times 8 = 6.4 \rightarrow 6 \\ 0.4 \times 8 = 3.2 \rightarrow 3 \\ 0.2 \times 8 = 1.6 \rightarrow 1$$

$$\therefore (60.7)_{10} = (74.54631)_8$$

Decimal to Hexadecimal :- ($H=16$)

$$\textcircled{1} \quad (20)_{10} = (?)_{16}$$

$$16 \overline{)20} \\ \underline{-14} \\ 1-4$$

$$\therefore (20)_{10} = (14)_{16}$$

$$\textcircled{2} \quad (1234)_{10} = (?)_{16}$$

$$16 \overline{)1234} \\ \underline{-16} \\ 77-2 \\ \underline{-16} \\ 4-13 \\ \text{(or)} \\ \underline{D}$$

$$\therefore (1234)_{10} = (4D2)_{16}$$

$$\textcircled{3} \quad (20.5)_{10} = (?)_{16}$$

$$16 \overline{)20} \\ \underline{-14} \\ 1-4$$

$$0.5 \times 16 = 8.0 \rightarrow 8 \\ 0.0 \times 16 = 0.0 \rightarrow 0$$

$$\therefore (20.5)_{10} = (14.8)_{16}$$

$$\textcircled{4} \quad (675.625)_{10} = (?)_{16}$$

$$16 \overline{)675} \\ \underline{-16} \\ 42-3 \\ \underline{-48} \\ 2-10 \\ \text{(or)} \\ \underline{A}$$

$$0.625 \times 16 = 10.000 \rightarrow 10 \text{ (or)} A \\ 0.000 \times 16 = 0.000 \rightarrow 0$$

$$\therefore (675.625)_{10} = (A73.A)_{16}$$

⇒ Binary to octal :-

To convert binary to octal, starting from binary point make group of 3 bits and write its equivalent.

$$\textcircled{1} \quad (101)_2 = (?)_8$$

$$101 \rightarrow 5$$

$$\therefore (101)_2 = (5)_8$$

$$\textcircled{2} \quad (1101)_2 = (?)_8$$

$$\begin{array}{r} 001101 \\ \underbrace{\quad}_{1} \quad \underbrace{10}_{5} \\ = 15 \end{array}$$

$$\therefore (1101)_2 = (15)_8$$

$$\textcircled{3} \quad (10.11001)_2 = (?)_8$$

$$\xleftarrow{10} \xrightarrow{11001}$$

$$\begin{array}{r} 010 \\ \downarrow 2 \\ 110 \\ \downarrow 6 \\ 010 \\ \downarrow 2 \end{array}$$

$$\therefore (10.11001)_2 = (2.6)_8$$

$$\textcircled{4} \quad (011010110.11)_2 = (?)_8$$

$$\xleftarrow{011010110} \xrightarrow{.11}$$

$$\begin{array}{r} 011 \\ \downarrow 3 \\ 010 \\ \downarrow 2 \\ 110 \\ \downarrow 6 \\ .110 \\ \downarrow 6 \end{array}$$

$$\therefore (011010110.11)_2 = (326.6)_8$$

$$\textcircled{5} \quad (1101101.01101)_2 = (?)_8$$

$$\begin{array}{r} 001101 \\ \downarrow \text{append zero} \\ 001101 \\ \downarrow 5 \\ 101 \\ \downarrow 5 \\ 011010 \\ \downarrow 3 \\ 011010 \\ \downarrow 2 \end{array}$$

$$\therefore (1101101.01101)_2 = (155.32)_8$$

Binary to Hexadecimal :-

To convert binary to Hexadecimal, Group 4-bits of binary and write its equivalent hexadecimal digit.

$$\textcircled{1} \quad (1101011011)_2 = (?)_{16}$$

0011 0101 1011
 ↘ ↘ ↘
 3 5 11 (0x)B

$$\therefore (1101011011)_2 = (35B)_{16}$$

$$\textcircled{2} \quad (1101011011.110101)_2 = (?)_{16}$$

0011 0101 1011 . 1101 0100
 ↘ ↘ ↘ ↘ ↘
 3 5 11 (0x)B 13 (0x)D 4

$$\therefore (1101011011.110101)_2 = (35B.D4)_{16}$$

$$\textcircled{3} \quad (10100110101111)_2 = (?)_{16}$$

0010 1001 1010 1111
 ↘ ↘ ↘ ↘
 2 9 10 (0x)A 15 (0x)F

$$\therefore (10100110101111)_2 = (29AF)_{16}$$

$$\textcircled{4} \quad (100101011.101110)_2 = (?)_{16}$$

0001 0010 1011 . 1011 1000
 ↘ ↘ ↘ ↘ ↘
 1 2 11 (0xB) 11 (0xB) 8

$$\therefore (100101011.101110)_2 = (12B.B8)_{16}$$

→ Octal to other Number Systems :-

⇒ Octal to Decimal :-

$$\textcircled{1} \quad (24)_8 = (?)_{10}$$

2	4
8^1	8^0

$$2 \times 8^1 + 4 \times 8^0 \\ = 16 + 4 = 20$$

$$\therefore (24)_8 = (20)_{10}$$

$$\textcircled{2} \quad (24.4)_8 = (?)_{10}$$

2	4.	4
8^1	8^0	8^{-1}

$$= 2 \times 8^1 + 4 \times 8^0 + 4 \times 8^{-1} \\ = 16 + 4 + 4 \times \frac{1}{8^2} \\ = 16 + 4 + 0.5 = 20.5$$

$$\therefore (24.4)_8 = (20.5)_{10}$$

$$\textcircled{3} \quad (6327.4051)_8 = (?)_{10}$$

6	3	2	7	.	4	0	5	1
8^3	8^2	8^1	8^0	.	8^{-1}	8^{-2}	8^{-3}	8^{-4}

$$= 6 \times 8^3 + 3 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 + 4 \times \frac{1}{8} \\ + 0 \times \frac{1}{8^2} + 5 \times \frac{1}{8^3} + 1 \times \frac{1}{8^4}$$

$$= 3072 + 192 + 96 + 7 + 0.5100098$$

$$= (3367.5100098)_{10}$$

$$\textcircled{4} \quad (1234.342)_8 = (?)_{10}$$

1	2	3	4	.	3	4	2
8^3	8^2	8^1	8^0	.	8^{-1}	8^{-2}	8^{-3}

$$1 \times 8^3 + 2 \times 8^2 + 3 \times 8^1 + 4 \times 8^0 + 3 \times \frac{1}{8} + 4 \times \frac{1}{8^2} \\ + 2 \times \frac{1}{8^3}$$

$$= 512 + 128 + 24 + 4 + 0.375 + 0.062 + 0.003$$

$$= (668.440)_{10}$$

⇒ Octal to Binary :- To convert octal to binary just replace each octal digit by its 3-bit binary equivalent.

$$\textcircled{1} \quad (15)_8 = (?)_2$$

$$\begin{array}{r} 1 \rightarrow 101 \\ 5 \rightarrow 101 \end{array}$$

$$\therefore (15)_8 = (001101)_2$$

$$\textcircled{2} \quad (736)_8 = (?)_2$$

$$\begin{array}{r} 7 \rightarrow 111 \\ 3 \rightarrow 011 \\ 6 \rightarrow 110 \end{array}$$

$$\therefore (736)_8 = (111011110)_2$$

$$\textcircled{3} \quad (563)_8 = (?)_2$$

$$\begin{array}{r} 5 \rightarrow 101 \\ 6 \rightarrow 110 \\ 3 \rightarrow 011 \end{array}$$

$$\therefore (563)_8 = (101110011)_2$$

$$\textcircled{4} \quad (725)_8 = (?)_2$$

$$\begin{array}{r} 7 \rightarrow 111 \\ 2 \rightarrow 010 \\ 5 \rightarrow 101 \end{array}$$

$$\therefore (725)_8 = (111010101)_2$$

$$\textcircled{5} \quad (326)_8 = (?)_2$$

$$\begin{array}{r} 3 \rightarrow 011 \\ 2 \rightarrow 010 \\ 6 \rightarrow 110 \end{array}$$

$$(326)_8 = (011010110)_2$$

$$\textcircled{6} \quad (452)_8 = (?)_2$$

$$\begin{array}{r} 4 \rightarrow 100 \\ 5 \rightarrow 101 \\ 2 \rightarrow 010 \end{array}$$

$$\therefore (452)_8 = 100101010$$

⇒ Octal to Hexadecimal :- There is no direct conversion available for octal to hexadecimal. To convert octal number into a hexadecimal number by converting octal to binary then binary to hexadecimal (②) Octal to decimal then decimal to hexadecimal.

Note :- $()_8 \xrightarrow{\text{Step 1}} ()_2 \xrightarrow{\text{Step 2}} ()_{16}$
 $()_8 \xrightarrow{\text{Step 1}} ()_{10} \xrightarrow{\text{Step 2}} ()_{16}$

① $(356.63)_8 = (?)_{16}$

Step ① Octal to Binary

Step ② Binary to Hexadecimal

$$\begin{array}{ccccccc} 3 & 5 & 6. & 6 & 3 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 011 & 101 & 110 & . & 110 & 011 \end{array}$$

$$\begin{array}{ccccccccc} 0000 & \underbrace{110} & \underbrace{110.} & \underbrace{1100} & \underbrace{1100} \\ 0 & E & E & C & C \\ =14 & =14 & =12 & =12 \end{array}$$

∴ $(356.63)_8 = (0EE.EC)_{16}$

② $(247.52)_8 = (?)_{16}$

Step ① Octal to Binary

Step ② Binary to Hexadecimal

$$\begin{array}{ccccccc} 2 & 4 & 7. & 5 & 2 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 010 & 100 & 111 & . & 101 & 010 \end{array}$$

$$\begin{array}{ccccccccc} 0000 & \underbrace{1010} & \underbrace{0111.} & \underbrace{1010} & \underbrace{1000} \\ 0 & \overset{(10)}{\underset{(0)}{\text{A}}} & 7 & \overset{(10)}{\underset{(0)}{\text{A}}} & 8 \\ & \text{Step 1} & & \text{Step 2} & \end{array}$$

∴ $(247.52)_8 = (0A7.A8)_{16}$

⇒ Hexadecimal to Other Number System :-

⇒ Hexadecimal to Binary :-

① $(2F9A)_{16} = (?)_2$

$$\begin{array}{l} 2 \rightarrow 0010 \\ F \rightarrow 1111 \\ 9 \rightarrow 1001 \\ A \rightarrow 1010 \end{array}$$

∴ $(2F9A)_{16} = (001011110011010)_2$

$$\textcircled{2} \quad (6A3)_{16} = (?)_2$$

6 → 0110

A → 1010

3 → 0011

$$\therefore (6A3)_{16} = (0110\ 1010\ 0011)_2$$

$$\textcircled{3} \quad (58C)_{16} = (?)_2$$

5 → 0101

8 → 1000

C → 1100

$$\therefore (58C)_{16} = (0101\ 1000\ 1100)_2$$

$$\textcircled{4} \quad (7DE3)_{16} = (?)_2$$

7 → 0111

D → 1101

E → 1110

3 → 0011

$$\therefore (7DE3)_{16} = (0111\ 1101\ 1110\ 0011)_2$$

⇒ Hexadecimal to Decimal :-

$$\textcircled{1} \quad (3A.2F)_{16} = (?)_{10}$$

3	A	.	2	F
16 ¹	16 ⁰	.	16 ⁻¹	16 ⁻²

$$3 \times 16^1 + 10 \times 16^0 + 2 \times 16^{-1} + 15 \times 16^{-2}$$

$$= 48 + 10 + \frac{2}{16} + \frac{15}{16^2}$$

$$\therefore (3A.2F)_{16} = (58.1836)_{10}$$

$$\textcircled{2} \quad (5E.7A)_{16} = (?)_{10}$$

5	E	.	7	A
16 ¹	16 ⁰	.	16 ⁻¹	16 ⁻²

$$5 \times 16^1 + 14 \times 16^0 + 7 \times \frac{1}{16^1} + 10 \times \frac{1}{16^2}$$

$$= 90 + 14 + 0.43 + 0.03$$

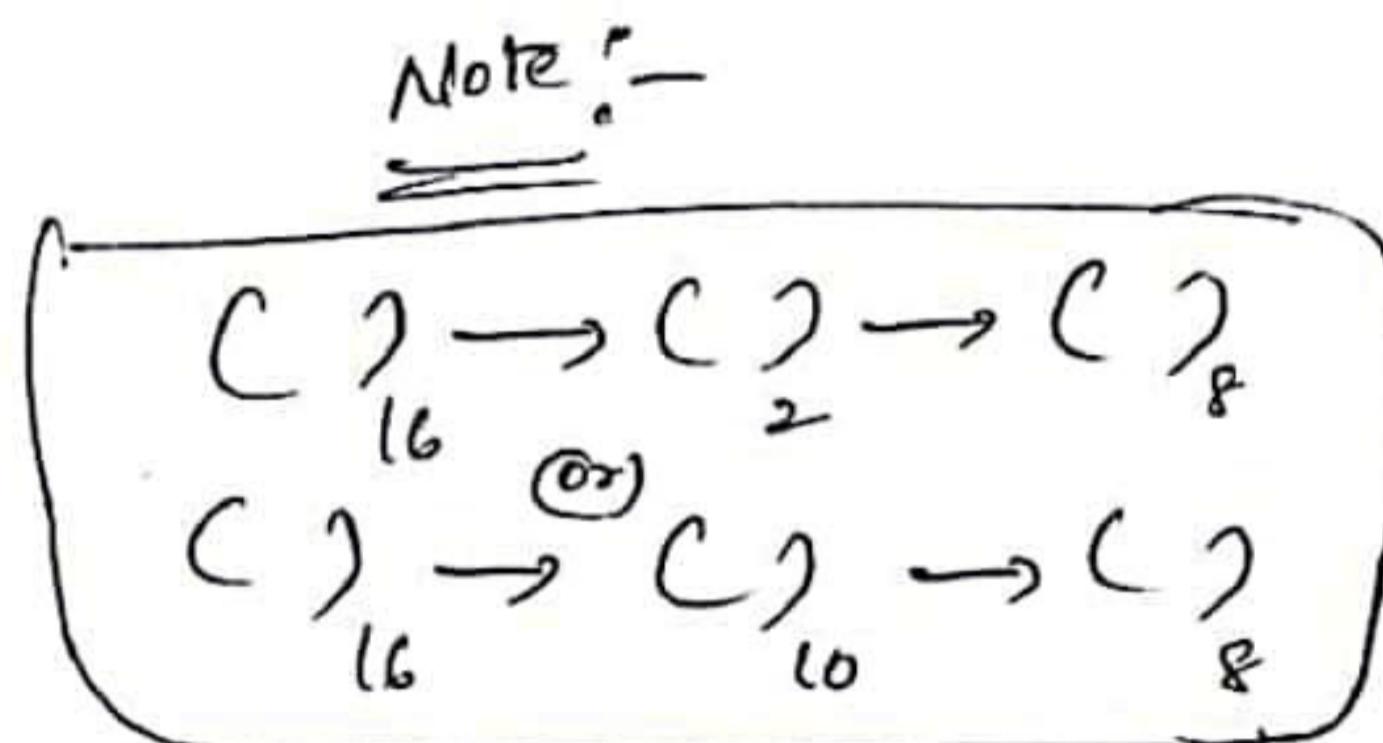
$$= (104.46)_{10}$$

$$\therefore (5E.7A)_{16} = (104.46)_{10}$$

→ Hexadecimal to Octal conversion :-

No direct conversion available, to convert hexadecimal to octal
first convert given hexadecimal number into decimal/binary
then into Octal system.

$$\textcircled{1} \quad (B9F.AE)_{16} = (?)_8$$



- ✓ Hexadecimal to binary
- ✓ Binary to Octal

$$\begin{array}{r}
 \begin{array}{c|c|c|c|c|c}
 B & 9 & F & . & A & E \\
 \hline
 1011 & 1001 & 1111 & . & 1010 & 1110
 \end{array} = \underbrace{101}_{5} \underbrace{110}_{6} \underbrace{011}_{3} \underbrace{111}_{7} . \underbrace{101}_{5} \underbrace{011}_{3} \underbrace{100}_{4}
 \end{array}$$

$$\therefore (B9F.AE)_{16} = (5637.534)_{8}$$

$$\textcircled{2} \quad (A8C.BC7)_{16} = (?)_8$$

- ✓ Hexadecimal to binary
- ✓ Binary to octal

$$\begin{array}{r}
 \begin{array}{c|c|c|c|c|c}
 A & 8 & C & . & B & C \\
 \hline
 1010 & 1000 & 1100 & 1011 & 1100 & 0111
 \end{array} = \dots
 \end{array}$$

$$\therefore (A8C.BC7)_{16} = (12434.5707)_{8}$$

Complement of Numbers :

(or) $(r-1)$'s complement and r 's complement

complements are used in systems to simplify the subtraction operation base (r adic) system there are two useful types of complements, r 's complement (Radix complement) and $(r-1)$'s complement (Diminished Radix complement).

⇒ $(r-1)$'s complement :-

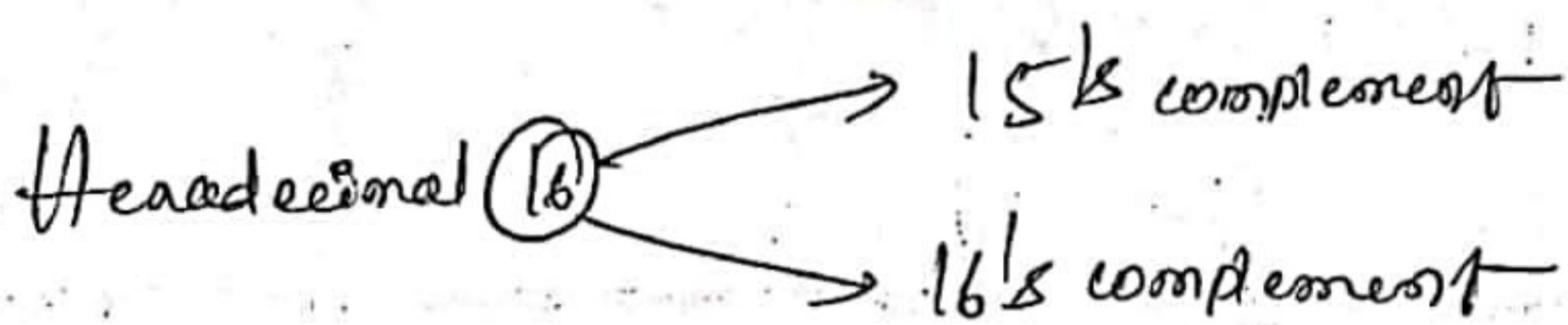
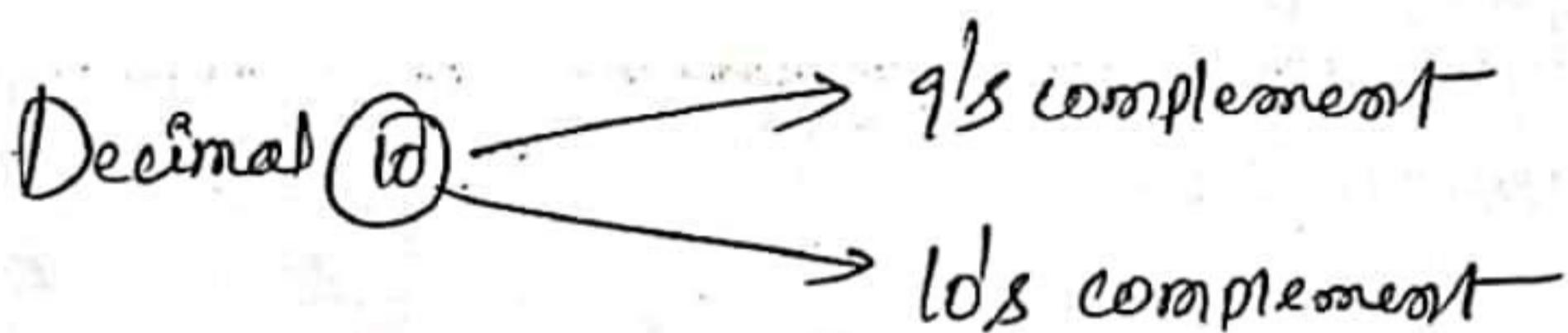
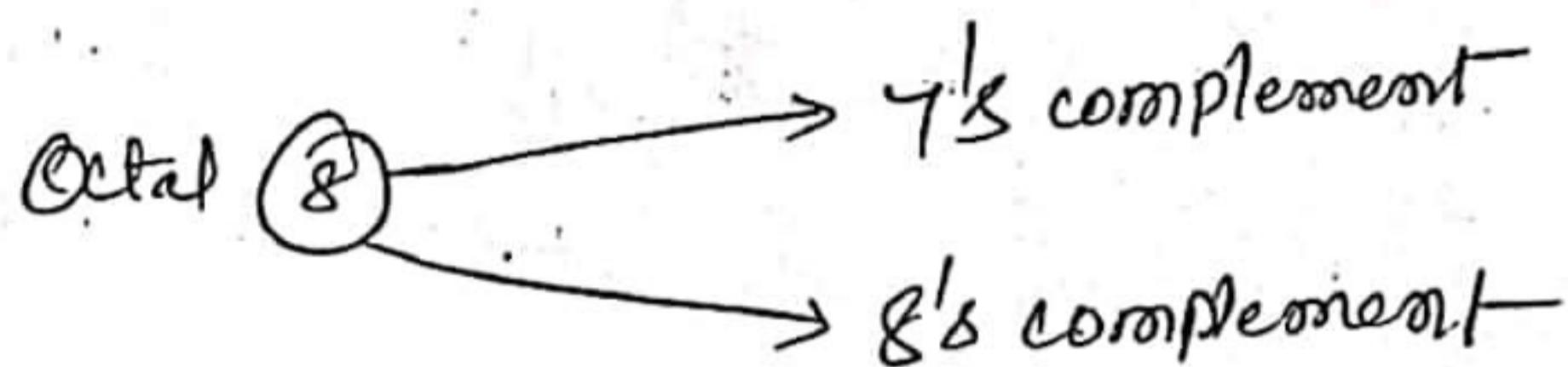
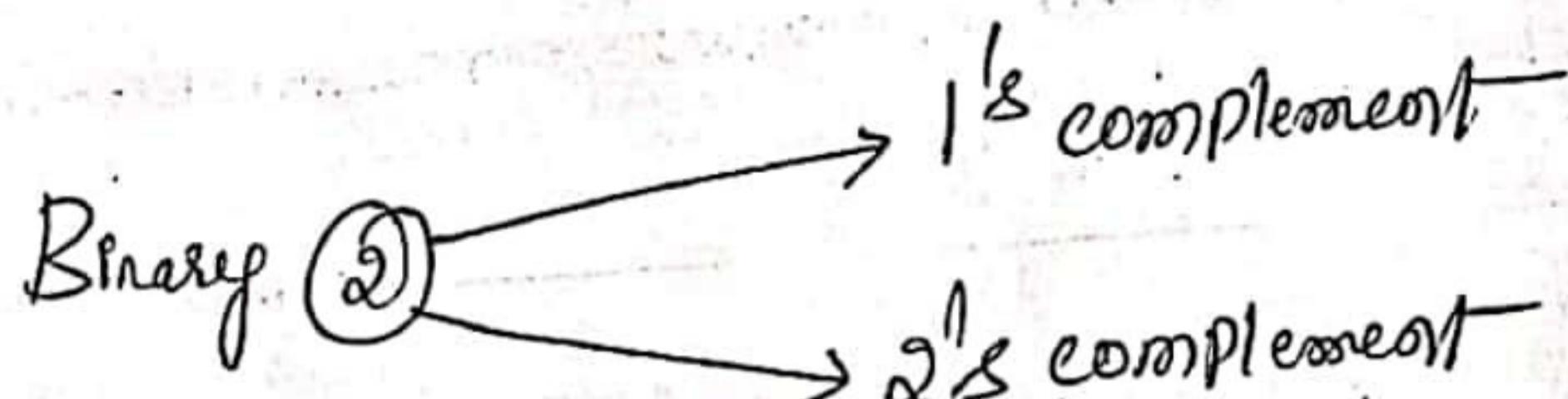
For a given Number ' N ' have the no. of digits ' n ' belonging to ' r ' number system, then $(r-1)$ complement is given by

$$(r^n - N) - 1$$

⇒ r 's complement :-

for a given number ' N ' have the no. of digits ' n ' belonging to ' r ' number system, then r 's complement is given by

$$r^n - N$$



⇒ 1's and 2's complements :-

The 1's complement of a binary number is obtained complementing all its bits, that is by replacing all 0's by 1's and all 1's by 0's.

Ex:- ① 0101101000 → (Given Binary Number)
1010010111 → (1's complement form)

The 2's complement of a binary number is obtained by adding '1' to its 1's complement.

①

$$\begin{array}{r}
 \text{Given} \\
 \hline
 0101101000 \rightarrow (\text{Given binary number}) \\
 1010010111 \rightarrow (1\text{'s complement form}) \\
 \hline
 \begin{array}{r}
 & 1 \text{ (adding 1)} \\
 \hline
 1010011000 & \rightarrow 2\text{'s complement}
 \end{array}
 \end{array}$$

\Rightarrow Binary addition

$0+0 = 0$ $0+1 = 1$ $1+0 = 1$ $1+1 = 1$ $0 \rightarrow \text{sum}$ \downarrow Carry
$1+1+1 = 1$ 1 \downarrow \downarrow Carry Sum

② $01101010 \rightarrow$ Given binary number

$100101010 \rightarrow$ 1's complement form

$+1 \rightarrow$ adding '1'

\hline
 $100101011 \rightarrow$ 2's complement form,

1's complement Arithmetic

- In 1's complement subtraction, add the 1's complement of the subtrahend to the minuend.
- If there is a carryout, bring the carry around and add it to the LSB. This is called "End around carry".
- Look at the sign bit (MSB). If this is a '0', the result is positive and is \pm pure binary. If MSB is '1', the result is negative and is \pm its 1's complement form. Take its 1's complement and put -ve sign to get magnitude in binary.

Ex ①

- ① Subtract 14 from 25 using 8-bit 1's complement arithmetic Method.

Sol Normally

$$\begin{array}{r}
 25 \longrightarrow 00011001 \\
 -14 \longrightarrow 11110001 \\
 \hline
 +11 \longrightarrow \boxed{1}00001010
 \end{array}$$

End around
carry

$$\begin{array}{l}
 14 \rightarrow 00001110 \\
 1^{\text{st}} \text{ complement} \rightarrow 11110001
 \end{array}$$

1 → (adding of round around
carry)

The MSB is '0', so the result is positive and is \pm pure binary. Therefore, the result is $00001010 = 1010_{10}$

② Add -25 to $+14$ using 8-bit 1's complement method

$$\begin{array}{r}
 +14 \rightarrow 00001110 \\
 -25 \rightarrow 11100110 \\
 \hline
 -11 \quad \underline{\quad} \quad \underline{11110100} \rightarrow \text{No carry}
 \end{array}
 \quad \begin{array}{l}
 25 \rightarrow 00011001 \\
 1s \rightarrow 1100110 \\
 \text{complement}
 \end{array}$$

\Rightarrow There is no carry. The MSB is a '1'. So, the result is negative and is in its 1's complement form. Take 1's complement to get the result.

\Rightarrow The complement of 11110100 is -00001011 . The result is -110 .

③ Add -25 to -14 using 8-bit 1's complement method.

$$\begin{array}{r}
 -25 \rightarrow 11100110 \quad (1's \text{ complement}) \\
 -14 \rightarrow 11110001 \quad (1's \text{ complement}) \\
 \hline
 -39 \quad \underline{\quad} \quad \underline{111010011}
 \end{array}
 \quad \begin{array}{l}
 25 \rightarrow 00011001 \quad (\text{normal form}) \\
 14 \rightarrow 00001011 \quad (\text{normal form})
 \end{array}$$

End around
carry
 $\frac{1101011}{1111}$ (Adding end around
carry)
 $\underline{\quad}\quad\underline{1101000}$
 MSB $\frac{0010011}{\quad}$ \rightarrow 1's complement

\rightarrow The MSB is '1' so the result is negative and we should find 1's complement above answer. The 1's complement of 11011000 is $\underline{\quad} 0010011$ therefore the result is -39 .

Q1 Add +25 to +14 using 8-bit 1's complement arithmetic.

$$\begin{array}{r}
 +25 \rightarrow 0001100 \\
 +14 \rightarrow \underline{0\ 001\ 0110} \\
 +39 \quad \underline{0\ 0100111}
 \end{array}$$

There is no carry. The MSB is '0'. So, the result is positive and is in pure binary. Therefore, the result is +39₁₀.

Ex: Q2

1) Subtract 20 from 36 using 8-bit 1's complement form

$$\begin{array}{r}
 36 \rightarrow 00100100 \\
 -20 \rightarrow \underline{1110101} \quad \text{(1st complement)} \\
 +16 \quad \underline{\quad \quad \quad \quad \quad \quad \quad \quad \quad} \\
 \text{End-around} \leftarrow \underline{100001111} \quad \text{(adding of end-around carry)} \\
 \text{carrying} \\
 \underline{00010000} \\
 \text{MSB is } 0
 \end{array}$$

⇒ The MSB is zero. The result is positive and is in True binary form.

Q2 Add +36 to +20 using 8-bit 1's complement form

$$\begin{array}{r}
 +36 \rightarrow 000100100 \\
 +20 \rightarrow \underline{00010100} \\
 \text{MSB is } 0 \quad \underline{00101000}
 \end{array}$$

$$\begin{array}{r}
 +36 \rightarrow 000100100 \\
 +20 \rightarrow \underline{00010100}
 \end{array}$$

MSB is zero. The result is positive and it is in true binary form.

③ Add -36 to -20 using 8-bit 1's complement form.

$$\begin{array}{r}
 -36 \rightarrow 11011011 \rightarrow 1\text{'s complement} \\
 -20 \rightarrow 11101011 \\
 \hline
 -56 \rightarrow 11111111 \\
 \text{end around carry} \quad \text{11000111} \\
 \hline
 \text{MSB = 1}
 \end{array}$$

$36 \rightarrow 00100100$
 $20 \rightarrow 00010100$

→ MSB is 1 the result is negative and it is in 1's complement form. To get the correct result take 1's complement to the result and put -ve sign before the result.

$$11000111 \xrightarrow{\text{1's complement}} -00111000 = -(56)_{10}$$

④ Add -36 to $+20$ using 8-bit 1's complement form.

$$\begin{array}{r}
 -36 \xrightarrow{\text{1's complement}} 11011011 \\
 +20 \xrightarrow{\text{Normal form}} 00010100 \\
 \hline
 -16 \rightarrow 11101111 \\
 \text{MSB = 1}
 \end{array}$$

$36 \rightarrow 00100100$
 $20 \rightarrow 00010100$

→ The MSB is 1. The result is negative and it is in 1's complement form.

→ Take 1's complement to the result and put -ve sign before the result

$$11101111 \rightarrow -00010000 = (-16)_{10}$$

⇒ 2's complement :- In 2's complement subtraction, add 2's complement of the subtrahend to the minuend. If there is a carryout, ignore it. If the MSB is '0', the result is positive and is in true binary form. If MSB is '1' the result is negative and is in 2's complement form.

Sol-
① Subtract 14 from 25 using 8-bit 2's complement arithmetic

$$\begin{array}{r}
 +14 = 0000\ 1110 \text{ (normal format)} \\
 1111\ 0001 \text{ (1's complement)} \\
 \hline
 \quad\quad\quad 1 \text{ (2's complement)} \\
 \hline
 \underbrace{1111\ 0010} \rightarrow 2\text{'s complement form}
 \end{array}$$

$$\begin{array}{r}
 25 \rightarrow 00011001 \\
 -14 \rightarrow \underline{11110010} \\
 \hline
 \boxed{1} 0001011
 \end{array}$$

Ignore
carry msb

→ There is a carry ignore it. The MSB is '0', so, the result is positive and is in normal binary form. Therefore,

$$\text{The result is } +0001011 = +17_{10}$$

② Add -25 to +14 using 8-bit 2's complement arithmetic

$$+25 \rightarrow 00011001$$

$$\begin{array}{r} -25 \rightarrow 11100110 \\ \hline 11100111 \end{array} \rightarrow \text{2's complement form}$$

$$+14 \rightarrow 00001110$$

$$\begin{array}{r} -25 \rightarrow 11100111 \\ \hline 11110101 \end{array} \text{(No carry)}.$$

\downarrow
MSB = 1

There is no carry, the MSB is '1'. So, the result is negative and is in 2's complement form. The magnitude is 2's complement of 11110101, that is 00001010 .

$$\underline{-00001011} = (-11)_{10}$$

Q3 Given the two binary numbers $X = 1010100$ and $Y = 1000011$, perform subtraction (a) $X-Y$ (b) $Y-X$ using 2's complements

$$\begin{array}{r} X = 1010100 \\ Y = 1000011 \rightarrow \text{subtrahend} \\ 0111100 \rightarrow 1's \text{ complement} \\ \hline 0111101 \rightarrow 2's \text{ complement} \end{array}$$

① Find 2's complement of subtrahend

② Add subtrahend to the minuend.

$$X = 1010100$$

$$Y = \underline{\begin{array}{r} 0111101 \\ 1111 \end{array}} \rightarrow 2's \text{ complement of } Y$$

$\boxed{1} \quad \boxed{0} \quad 00001$

Discard carry
MSB = 0

If MSB = 0 the result is positive and it is in true binary form.

(b)

$$y - x$$

$$y = 1000011 \quad X = 1010100$$

$$\begin{array}{r} 0101011 \rightarrow 1's \text{ complement} \\ \hline 111 \\ \hline 0101100 \rightarrow 2's \text{ complement} \end{array}$$

$$Y = 1000011$$

$$\begin{array}{r} 0101100 \rightarrow 2's \text{ complement of } \underline{Y} \\ \hline 1101111 \\ \hline \text{MSB} = 1 \end{array}$$

There is no carry. And the MSB is '1' so the answer is 2's complement form. So find 2's complement of the result to get the correct answer

$$\begin{array}{r} 1101111 \\ 0010000 \rightarrow 1's \text{ complement} \\ \hline 1 \\ \hline 0010001 \rightarrow \text{correct answer} \\ \text{2's complement form} \end{array}$$

Simp Given the two binary numbers $X = 1010100$ and $Y = 1000011$, perform the subtraction ① $X - Y$ and ② $Y - X$ using 1's complement.

①

$$X = 1010100$$

$$\begin{array}{r} = 0111100 \rightarrow 1\text{'s complement of } Y \\ \hline 1111 \\ \boxed{1}0010000 \end{array}$$

End around
carry ↓

$$\begin{array}{r} \\ \\ \hline 0010001 \end{array}$$

$mSB = 0$ so, it is in true binary form

②

$$Y - X$$

$$Y = 1000011$$

$$\begin{array}{r} X = 0101011 \rightarrow 1\text{'s complement of } Y \\ \hline 11 \\ \hline 1101110 \\ \hline mSB = 1 \end{array}$$

There is no carry. And the mSB is '1' so the result is in 1's complement form. So, find 1's complement of answer

$$1\text{'s complement of } 1101110 \underline{\quad} -0010001$$

\Rightarrow q1s and 101s complement :-

\rightarrow In q1s complement subtraction just follow the below rules

- ① Find the q1s complement of Subtrahend and Add. q1s complement of Subtrahend to minuend.
- ② If there is a carry it indicates that the answer is +ve then add carry to the LSD of the result to get true answer.
- ③ If there is no carry, it indicates that the answer is negative and the result obtained is its q1s complement
- ④ Find the q1s complement of the following decimal numbers

① 3465

Sof

$$\begin{array}{r} 9999 \\ - 3465 \\ \hline 6534 \end{array}$$

(q1s complement)

+ 3465

By using formula

$$\left. \begin{array}{l} (2^4 - n) - 1 \\ (10^4 - 3465) - 1 = 6534 \end{array} \right\}$$

② 782.54

Sof

$$\begin{array}{r} 999.98 \\ - 782.54 \\ \hline 217.45 \end{array}$$

③ 4526.075

Sof

$$\begin{array}{r} 9999.998 \\ - 4526.075 \\ \hline 5473.924 \end{array}$$

9's complement Method of Subtraction

Q Subtract the following numbers using 9's complement method.

① $745.81 - 436.62$

Step ①

$$\begin{array}{r} 999.99 \\ 436.62 \\ \hline 563.37 \end{array}$$

→ 9's complement of
436.62

Step ②

$$745.81$$

$$\begin{array}{r} 563.37 \\ \hline 309.18 \end{array}$$

→ (adding 1 and crossed carry)
309.19 → final answer.

Crossed
indicates
the answer is true

② $436.62 - 745.81$

Step ①

$$\begin{array}{r} 999.99 \\ 745.81 \\ \hline 254.18 \end{array}$$

→ 9's complement of 745.81

Step ②

$$\begin{array}{r} 436.62 \\ 254.18 \\ \hline 690.80 \end{array}$$

There is no carry, so it indicating that the answer is negative, so, take 9's complement of the intermediate result and put a minus sign before it.

$$\begin{array}{r}
 999.99 \\
 690.80 \\
 \hline
 -309.19 \rightarrow \text{Therefore the answer is } -309.19
 \end{array}$$

10's complement method of subtraction

The 10's complement of a decimal number is obtained by adding a '1' to its 9's complement.

Q Find the 10's complement of the following decimal numbers.

① 3465

$$\begin{array}{r}
 \text{Sof} \quad 9999 \\
 3465 \\
 \hline
 6534 \rightarrow 9's \text{ complement} \\
 \hline
 6535 \rightarrow 10's \text{ complement}
 \end{array}$$

② 782.54

$$\begin{array}{r}
 \text{Sof} \quad 999.99 \\
 782.54 \\
 \hline
 217.45 \rightarrow 9's \text{ complement} \\
 \hline
 217.46 \rightarrow 10's \text{ complement}
 \end{array}$$

③ 4526.075

$$\begin{array}{r}
 9999.999 \\
 4526.075 \\
 \hline
 5473.924 \\
 \hline
 5473.925
 \end{array}$$

10's complement method of Subtraction

- ① To perform decimal subtraction using 10's complement method, obtained the 10's complement of the subtrahend and add it to the minuend.
- ② If there is a carry, ignore it. the presence of the carry indicates that the answer is positive.
- ③ If there is no carry, it predicates the answer is negative and the result obtained in 10's complement form and put negative sign in front of the answer -

① $745.81 - 436.62$

Step ①

999.99	-	436.62
563.37		-----

563.38 → 10's.
complement form

② $436.62 - 745.81$

Step ①

999.99	-	745.81
254.18		-----

254.19

Step ②

745.8	-	562.38
209.19		-----

Ignore the carry
carry indicated result is positive

Step ②

436.62	-	254.19
690.81		-----

no carry
answer is -1

Step ③

999.99	-	690.81
309.18		-----

309.18

15's complement Method :-

Q Find the 15's complement of the following numbers

(a) $6A36$

$$\begin{array}{r} \text{Step} \\ \text{---} \\ \text{(i)} \quad \begin{array}{r} \text{FFFF} \\ \text{6A36} \end{array} \quad (\text{or}) \quad \begin{array}{r} 15 \ 15 \ 15 \ 15 \\ 6 \ A \ -3 \ 6 \end{array} \\ \text{---} \\ \text{(ii)} \quad \begin{array}{r} 95C9 \\ \hline 95C9 \end{array} \quad \begin{array}{r} 95C9 \\ \hline 95C9 \end{array} \end{array}$$

(b) $9AD\cdot3A$

$$\begin{array}{r} 15 \ 15 \ 15 \ 15 \ 15 \\ \text{(i)} \quad \begin{array}{r} 9 \ A \ D \cdot 3 \ A \\ \hline 6 \ 5 \ 2 \cdot C \ 5 \end{array} \rightarrow (15\text{'s complement}) \end{array}$$

\Rightarrow 15's complement method of subtraction

(a) $69B - C14$

Step① 15's complement of $(-C14)$

$$\begin{array}{r} 15 \ 15 \ 15 \\ \text{(i)} \quad \begin{array}{r} C \ 1 \ 4 \\ \hline 3 \ E \ B \end{array} \rightarrow (15\text{'s complement of } (-C14)) \end{array}$$

Step②

$$69B - C14 = 69B + (15\text{'s complement of } (-C14))$$

$$\begin{array}{r} 6 \ 9 \ B \\ \text{(i)} \quad \begin{array}{r} 3 \ E \ B \\ \hline A \ 8 \ 6 \end{array} \end{array} \quad \begin{array}{l} (22)_{10} = (16)_4 \\ (24)_{10} = (18)_4 \end{array}$$

There is no carry, result is \leftarrow ve

Step ③ 15's complement of intermediate result is given by

$$\begin{array}{r}
 15 \ 15 \ 15 \\
 A \ 8 \ 6 \\
 \hline
 5 \ 7 \ 9
 \end{array}$$

\therefore Final result is $-(579)_{16}$

(b) $-69B + C14 \text{ or } C14 - 69B$

Step ① 15's complement of $(-69B)$

$$\begin{array}{r}
 15 \ 15 \ 15 \\
 - 6 \ 9 \ B \\
 \hline
 9 \ 6 \ 4 \rightarrow 15's \text{ complement}
 \end{array}$$

Step ②

$$C14 - 69B = C14 + (15's \text{ complement of } (-69B))$$

$$\begin{array}{r}
 C14 \\
 + 964 \\
 \hline
 1578
 \end{array}
 \quad (21)_{10} = (15)_{16}$$

carry

There is a carry, so the result is true

$$\begin{array}{r}
 578 \\
 1 \text{ (end around carry)} \\
 \hline
 579
 \end{array}
 \quad \therefore \text{final result} = +(579)_{16}$$

16's complement method :-

First find the 15's complement and then add 1

Q Find the 16's complement of the following numbers

(a) A8C

15's complement is given by

$$\begin{array}{r}
 15 \ 15 \ 15 \\
 \text{---} \\
 \text{A} \ 8 \ \text{C} \\
 \text{---} \\
 5 \ 7 \ 3
 \end{array}
 \rightarrow \text{15's complement}$$

$$\begin{array}{r}
 16\text{'s complement} \rightarrow 573 \\
 \text{---} \\
 \text{+} \ 1 \\
 \text{---} \\
 574
 \end{array}$$

16's complement method of subtraction :-

Find the 16's complement subtraction of the following numbers.

(a) C9B - C14

Step 1 15's complement of (-C14)

$$\begin{array}{r}
 15 \ 15 \ 15 \\
 \text{---} \\
 \text{C} \ 1 \ 4 \\
 \text{---} \\
 3 \ E \ B
 \end{array}
 \rightarrow \text{15's complement}$$

$$\begin{array}{r}
 16\text{'s complement} \rightarrow \frac{3 \ E \ B}{1} \\
 \text{---} \\
 3 \ E \ C
 \end{array}$$

Step ② :-

$$C9B - C14 = C9B + (16^{\text{ls complement}} \text{ of } (-C14))$$

$$\begin{array}{r} C9B \\ + 13 \text{ E C} \\ \hline 1087 \end{array}$$

carry

$$(23)_{10} = (17)_{16}$$

$$(24)_{10} = (18)_{16}$$

$$(16)_{10} = (10)_{16}$$

There is a carry ignore it. Since the carry is 1. The result is +ve.

$$\therefore \text{Final result is } +(087)_{16}.$$

$$⑥ 2A4.2D - 3B2.3C$$

step ① :- 15's complement of $(-3B2.3C)$

$$\begin{array}{r} 15 15 15. 15 15 \\ - 3 B 2 \quad 3 . C \\ \hline C 4 D . C 3 \end{array} \rightarrow 15\text{'s complement}$$

16's complement is given by,

$$\begin{array}{r} C 4 D . C 3 \\ \hline C 4 D . C 4 \end{array} \rightarrow 16\text{'s complement}$$

step ②

$$2A4.2D - 3B2.3C = 2A4.2D + (16^{\text{ls complement}} \text{ of } (-3B2.3C))$$

$$\begin{array}{r} 2A4.2D \\ - C 4 D . C 4 \\ \hline E F 1 . F 1 \end{array} \rightarrow \text{intermediate result}$$

There is no carry. So the result is +ve

Step ③

15's complement of intermediate result is given by

$$\begin{array}{r}
 15 \ 15 \ 15 \cdot 15 \ 15 \\
 E \ F \ . \ I \ . \ F \ I \\
 \hline
 1 \ 0 \ E \cdot 0 \ E \rightarrow 15\text{'s complement} \\
 \oplus \quad | \\
 \hline
 1 \ 0 \ E \cdot 0 \ F \rightarrow 16\text{'s complement}
 \end{array}$$

∴ Final result is $-(OE \cdot OF)_{16}$

⇒ 7 and 8's complements:

Subtract the following numbers using 7's complement method.

(a) $234.65 - 135.74$

sof 7's complement of (-135.74) is given by

$$\begin{array}{r}
 777.77 \\
 135.74 \\
 \hline
 642.03 \rightarrow 7\text{'s complement}
 \end{array}$$

$\therefore 234.65 + (7\text{'s complement of } (-135.74))$

$$\begin{array}{r}
 234.65 \\
 642.03 \\
 \hline
 076.70 \quad (\text{Carry} \Rightarrow \text{result is positive})
 \end{array}$$

$(8) = (10)_8$

076.70

(+1)

76.71 (End around carry)

∴ Result is +76.71

③ $135.74 - 236.65$

7's complement of (-236.65) is given by,

$$\begin{array}{r} 777.77 \\ 236.65 \\ \hline 541.12 \end{array} \rightarrow 7's \text{ complement}$$

$$\therefore 135.74 - 236.65 = 135.74 + (7's \text{ complement of } (-236.65))$$

$$\Rightarrow 135.74$$

$$\underline{541.12} \rightarrow (7's \text{ complement})$$

$$\underline{677.06} \rightarrow \text{Intermediate result}$$

There is no carry. Hence the final result is (+ve).

Final result is the 7's complement of the above intermediate result.

$$\begin{array}{r} 777.77 \\ \ominus 677.06 \\ \hline 100.71 \end{array}$$

∴ Final result is -100.71

Solve the following using 8's complement method:

(a) $246.31 - 162.45$

7's complement of (-162.45) is given by,

$$777.77$$

$$162.45$$

$$\underline{615.32} \rightarrow (\text{7's complement})$$

$$(+1)$$

$$\underline{615.33} \rightarrow (\text{8's complement})$$

$$\therefore 246.31 - 162.45 = 246.31 + (\text{8's complement of } (-162.45))$$

$$\Rightarrow 246.31$$

$$615.33$$

$$\underline{063.64}$$

carry There is a carry. Hence the result is (+) ve
and ignore the carry.

$$\therefore \text{Final result is } +63.64$$

(b) $162.45 - 246.31$

8's complement of (-246.31) is given by,

$$777.77$$

$$(-246.31)$$

$$\underline{531.46} \rightarrow (\text{7's complement})$$

$$(+1)$$

$$\underline{531.47} \rightarrow (\text{8's complement})$$

$$\therefore 162.45 - 246.31 = 162.45 + (8\text{'s complement of } (-246.31))$$

$$\begin{array}{r} 162.45 \\ - 246.31 \\ \hline 714.14 \end{array} \rightarrow (\text{Intermediate result})$$

There is no carry. Hence the final result is \leftarrow ve. Final result is the 8's complement of the above intermediate result.

$$\begin{array}{r} 777.77 \\ - 714.14 \\ \hline 063.63 \end{array} \rightarrow 7\text{'s complement}$$
$$\begin{array}{r} (+) 1 \\ \hline 063.64 \end{array} \rightarrow 8\text{'s complement}$$

\therefore Final result is $- \underline{63.64}$

→ Floating point Representation :-

The goal of floating point representation is represent a large range of numbers.

Ex :- Given the number -123.154×10^5

Sign = - (Negative)

①

Mantissa = 123.154

Exponent = 5

Base = 10 (Decimal)

Eg :-

Distance b/w two
planet = 5.9×10^{12}

② mass of electron
 $= 9.1 \times 10^{-31}$ gm.

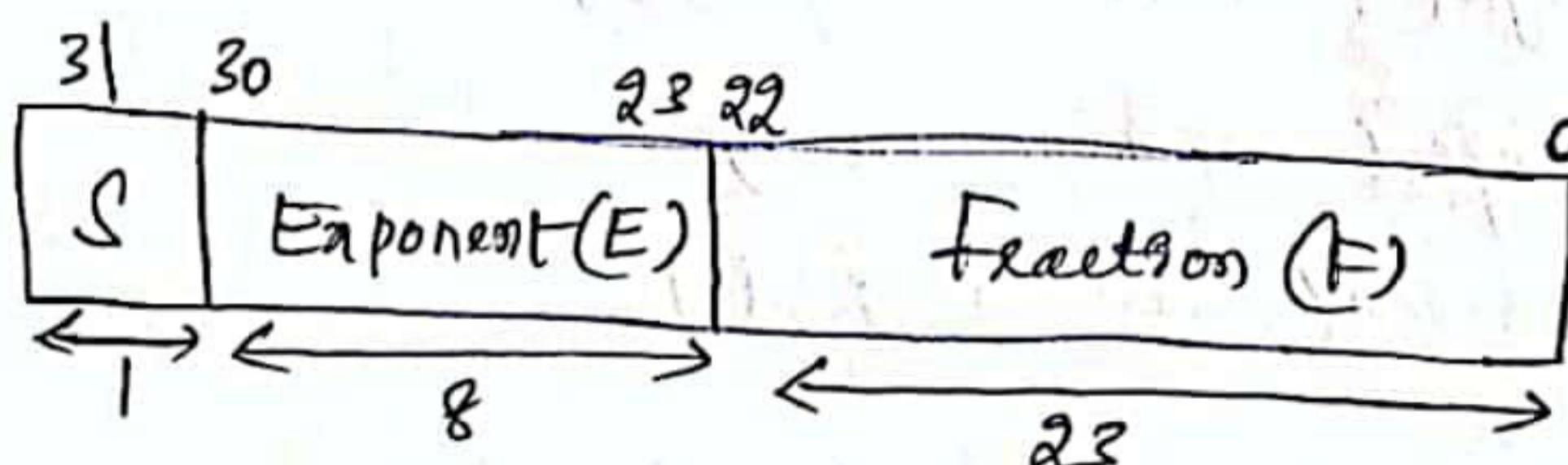
Ex :- Given the number 732.136×10^7

Sign = + (Positive)

Mantissa = 732.136

Exponent = 7

Base = 10 (Decimal)



32-bit single-precision Floating point Number

Ex :-

4.2×10^8 → Exponent
↓ ↓
Mantissa Base

∴ Only the mantissa and exponent are stored. The base is implied (Already known). It will save the memory.

$$\sum_{n=0}^{\infty} \frac{(11.8)_n}{10^n} = (1011.11001\dots)_2$$

Decimal representation

$$= (1.0111001) \times 2^3$$

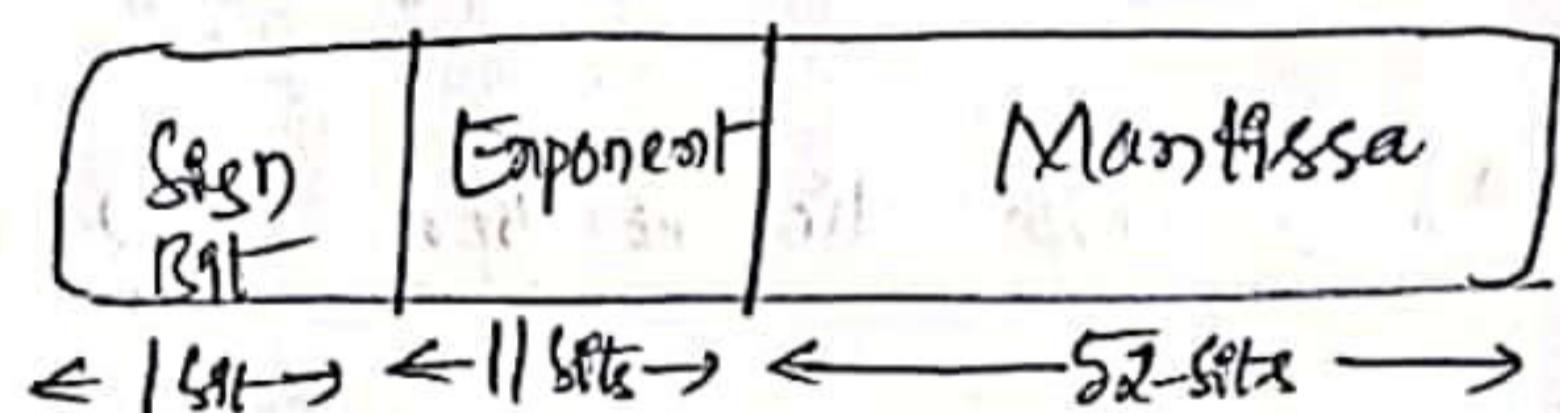
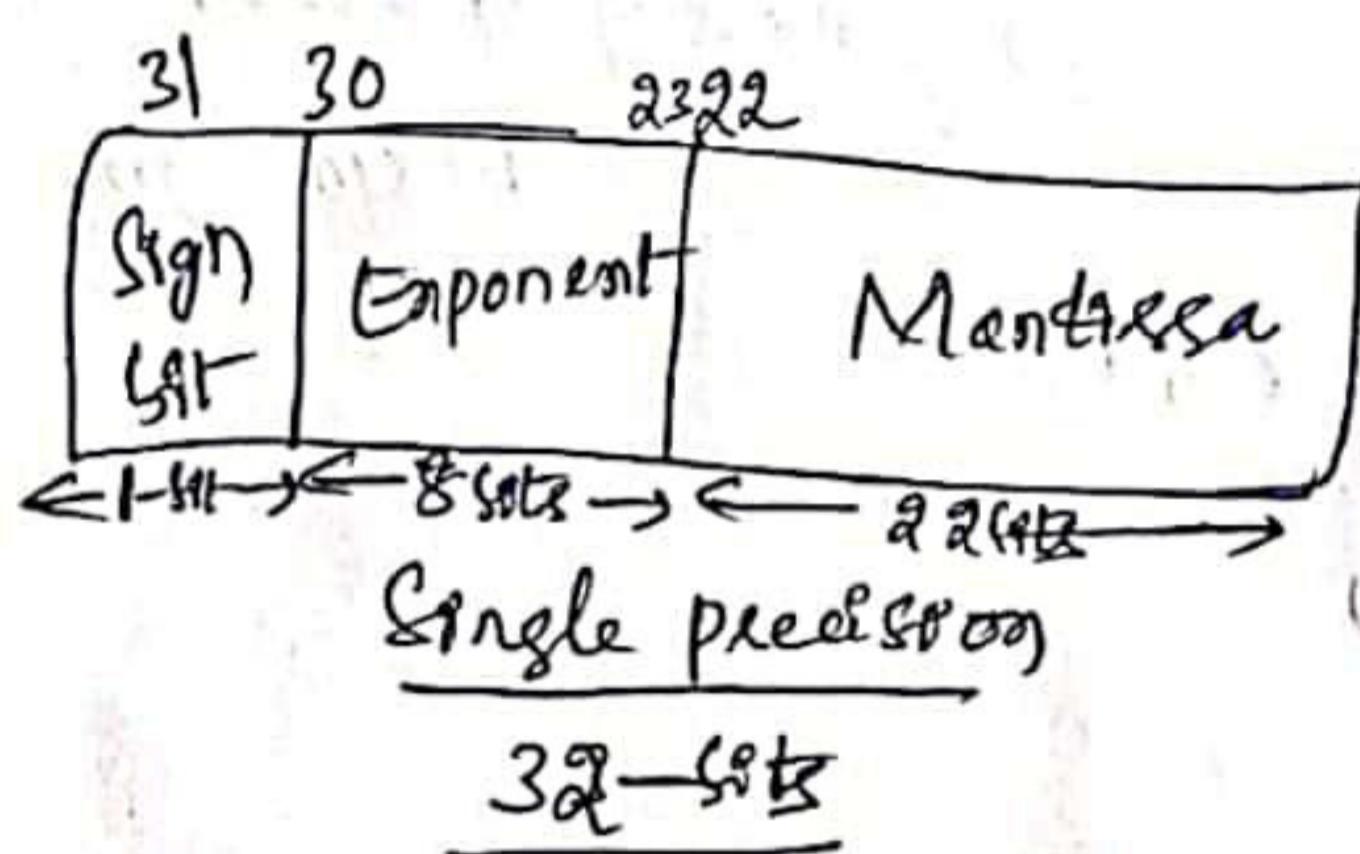
$$= \left(1.01111001\right)_2 \cdot 2^{(11)_2}$$

$$12345 = \frac{1.2345}{\downarrow} \times \frac{10^4}{\downarrow}$$

mantissa Base
(10⁴)

↑ Exponent

⇒ We will represent floating point numbers in single precision
and double precision formats. They are shown below



Double precision

CREEE

* 1st for the sign (positive or negative)

8 bit for the range (exponent field)

23 bit for the precision (fraction field)

$$\left\{ \begin{array}{l} N = (-1)^s \times 1.\text{fraction} \times 2^{\text{exponent}-127} \quad 1 \leq \text{exponent} \leq 254 \\ N = (+1)^s \times 0.\text{fraction} \times 2^{\text{exponent}-126}, \quad \text{exponent} = 0. \end{array} \right.$$

$$\text{Value} = (-1)^S \times (1+F) \times 2^{E-127} @ r$$

↑ single precision

$\boxed{\text{Value} = (-1)^S \times 2^{E-1024} \times 1.M}$

$\boxed{N = (-1)^S \times 2^{E-127} \times 1.M}$

↑ double precision

Fixed point

Representation

- ① A representation of real data type for a number that has a fixed number of digits after the radix point.
- ② Used to represent a limited range of values.
- ③ High performance
- ④ Less flexible

Floating point

Representation

- ① A formulaic representation of real numbers as an approximation so as to support a trade off between range and precision.
- ② used to represent a wide range of values.
- ③ High performance
- ④ More flexible

*

Real Number



questions:-

Q what is the 111.11 in decimal

- Ⓐ 7.75
- Ⓑ 31
- Ⓒ 7.375
- Ⓓ 15.25

* Limitation of floating points

✓ Size of mantissa is fixed.
Sol Use a floating point format with a larger mantissa.
 double (80 bits) long double (64 bits)

Q what is 8.5 in binary

- Ⓐ 1111111.1111
- Ⓑ 1000.01
- Ⓒ 0.100011
- Ⓓ 1000.10

Ex: -114.625 represent in binary

Sol:

128	64	32	16	8	4	2	1	0.5	0.25	0.125	
0	1	1	1	0	0	1	0	.	1	0	1

$$64 + 32 + 16 + 8 = \underline{114} \uparrow \quad 0.5 + \cancel{0.25} + 0.125$$

$$= 01110010.101$$

133 in binary

$$= 1.\underline{110010101} \times 2^6$$

$$10000101$$

$$\therefore \boxed{1} \boxed{10000101} \boxed{110010101}$$

Sign bit

Sign Exponent Mantissa

0	1001010	11101000
---	---------	----------

$$\begin{array}{ccccccc}
 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
 & \downarrow \\
 & 4 & 10 & 14 & 8 & & & & & & & & & & & \\
 & A & E & & & & & & & & & & & & & \\
 \end{array} = (4 \times 10^8) \text{ H}$$

Ex:

0.000110011001100110011001100 representation floating point in 32-bits.

Sol:

$$1.00011001100110011001100 \times 2^{-4}$$

$$\begin{array}{ccccccc}
 & 128 & 64 & 32 & 16 & 8 & 4 & 2 & 1 \\
 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
 \end{array} = \underline{132}$$

$$\text{Exponent} = -4 + 127 = 123$$

Sign bit = 0

Mantissa = 100011001100110011001100

S (1bit)	Exponent 8bits	Mantissa (23 bits)
0	01111011	100011001100110011001100.

Fixed point representation :-

⇒ Representation of signed binary numbers :-

Positive numbers can be represented by unsigned numbers however to represent negative numbers, we need notation for negative numbers.

There are two types of numbers

- ① Unsigned numbers
- ② Signed numbers

① Unsigned numbers :- There is no specific bit for sign representation. The numbers without positive or negative signs are known as unsigned numbers. The unsigned numbers are always positive numbers.

② Signed Numbers :- There is a specific bit for sign representation. In signed numbers, the number may be positive or negative. Different formats are used for representation of signed binary numbers. They are

① Signed magnitude representation

② 1's complement representation

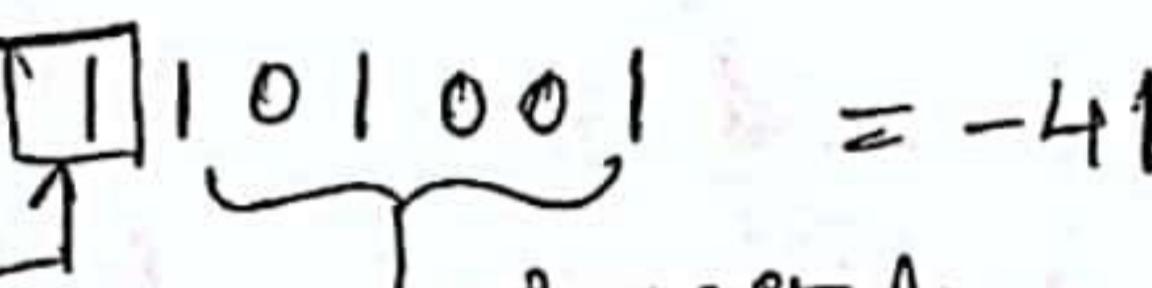
③ 2's complement representation

① Sign magnitude representation :-

In signed magnitude form, an additional bit called the 'sign bit' is placed in front of the number. If the sign bit is a '0', the number is positive. If it is a '1', the number is negative.

→ For example :-


0 1 0 1 0 0 1 = +41
Sign bit magnitude


1 1 0 1 0 0 1 = -41
Sign bit magnitude

In sign magnitude representation the '1's' represents the 'sign' and remaining 'bits' represents the 'magnitude'.

② 1's complement representation :-

In 1's complement representation the positive numbers remain unchanged. 1's complement representation of negative numbers can be obtained by the 1's complement of the binary number.

0 110011 = +51; MSB = 0 for the sign bit

1 001100 = -51; MSB = 1 for -ve sign bit

② 2's complement representation

In 2's complement representation, the positive numbers remain unchanged, 2's complement representation of negative numbers can be obtained by

1. Find the 1's complement of the number
2. To find 2's complement of the number adding '1' to the 1's complement number.

0 110011 = +51
Sign bit magnitude

1 001101 = -51 [In sign 2's complement form]
Sign bit magnitude

Number systems	+9	-9
unsigned	+1001	-1001
Sign magnitude	<u>0</u> 1001	<u>1</u> 1001
Sign 1's complement form	0 1001	1 0110
Sign 2's complement form	0 1001	1 0111

Decimal	Sign magnitude form	Sign 1's complement form	Sign 2's complement form
+7	0 111	0 111	0 111
+6	0 110	0 110	0 110
+5	0 101	0 101	0 101
+4	0 100	0 100	0 100
+3	0 011	0 011	0 011
+2	0 010	0 010	0 010
+1	0 001	0 001	0 001
+0	0 000	0 000	—
-0	1 000	1 111	1 111
-1	1 001	1 110	1 111
-2	1 010	1 101	1 110
-3	1 011	1 100	1 101
-4	1 100	1 011	1 100
-5	1 101	1 010	1 011
-6	1 110	1 001	1 010
-7	1 111	1 000	1 001

Q Represent +51 and -51 in sign magnitude, sign 1's complement and sign 2's complement representation.

Sol

Sign magnitude

$$\begin{array}{r} +51 \\ \underline{0\ 110011} \\ \text{Sign bit} \end{array} \quad \begin{array}{r} -51 \\ \underline{1\ 110011} \\ \text{Sign bit} \end{array}$$

Sign 1's complement

$$\begin{array}{r} 0\ 110011 \\ \underline{\text{Sign bit}} \end{array} \quad \begin{array}{r} 1\ 001100 \\ \underline{\text{Sign bit}} \end{array}$$

Sign 2's complement

$$\begin{array}{r} 0\ 110011 \\ \underline{\text{Sign bit}} \end{array} \quad \begin{array}{r} 1\ 001101 \\ \underline{\text{Sign bit}} \end{array}$$

Q

Represent +43 and -43 in sign magnitude, sign 1's complement and 2's complement representation

Sol

Sign magnitude

$$\begin{array}{r} +43 \\ \underline{0\ 101011} \\ \text{Sign bit} \end{array} \quad \begin{array}{r} -43 \\ \underline{1\ 101011} \\ \text{Sign bit} \end{array}$$

Sign 1's complement

$$\begin{array}{r} 0\ 101011 \\ \underline{\text{Sign bit}} \end{array} \quad \begin{array}{r} 1\ 010100 \\ \underline{\text{Sign bit}} \end{array}$$

Sign 2's complement

$$\begin{array}{r} 0\ 101011 \\ \underline{\text{Sign bit}} \end{array} \quad \begin{array}{r} 1\ 010101 \\ \underline{\text{Sign bit}} \end{array}$$

→ Combinational Circuits :-

→ Boolean Expressions :- Boolean Algebra is a division of mathematics which deals with operations on logic values and incorporates binary variable. Boolean algebra was invented by great mathematician George Boole in 1854.

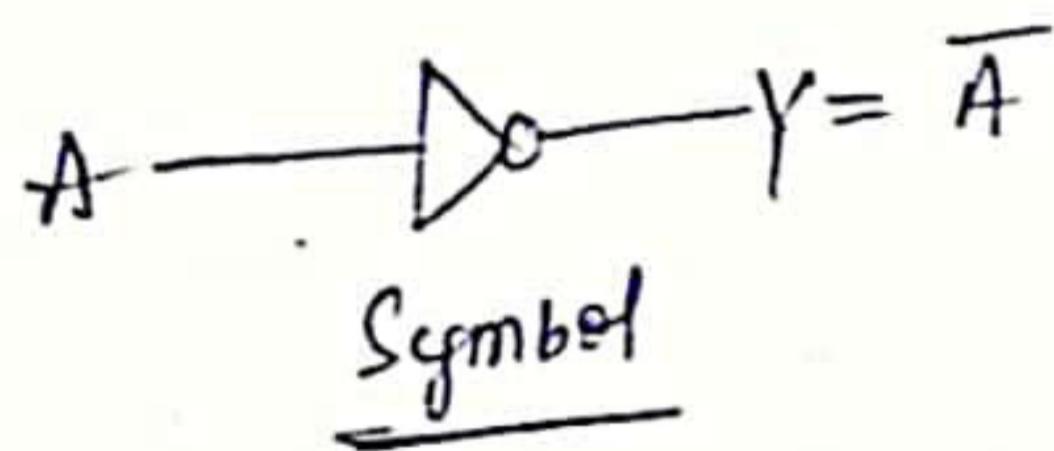
→ Minimization of logic expressions can be done by using boolean theorems and laws.

→ In Boolean algebra, Karnaugh map (K-map) are used for boolean minimization.

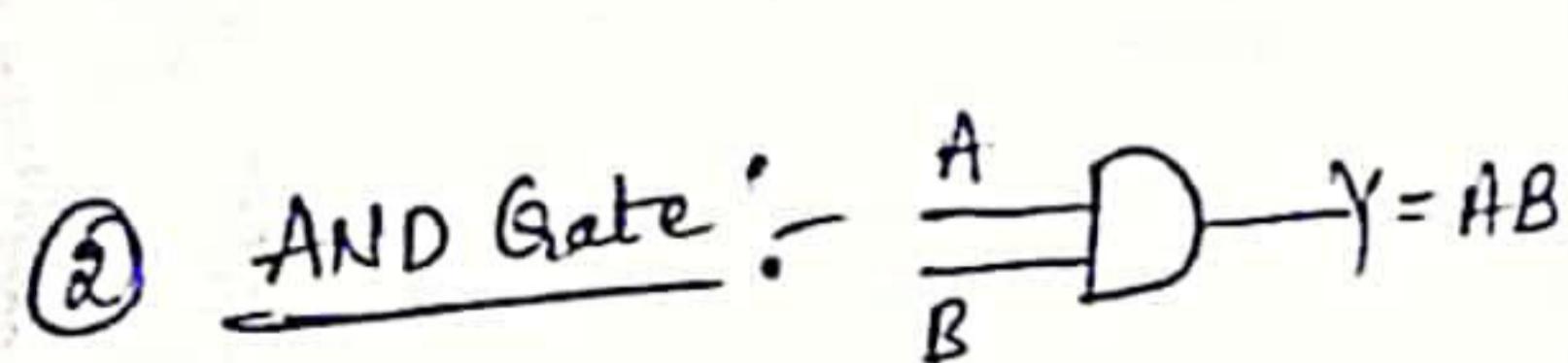
→ The main motto of this concept is to make information simpler, cheaper and low cost.

→ Logic Gates :- ① Not gate (or) Inverter :-

Truth Table

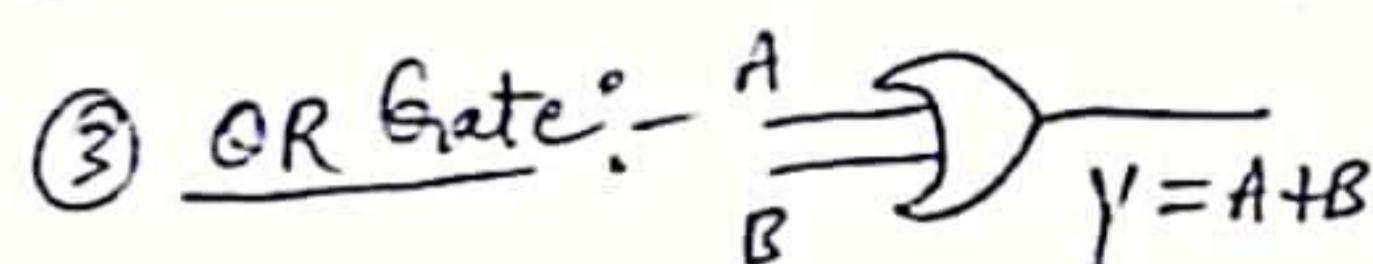


A	$Y = \bar{A}$
0	1
1	0

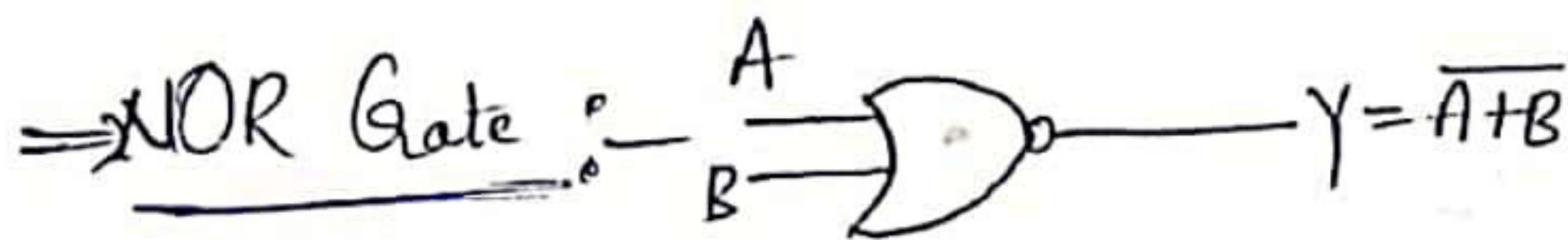


A	B	$Y = AB$
0	0	0
0	1	0
1	0	0
1	1	1

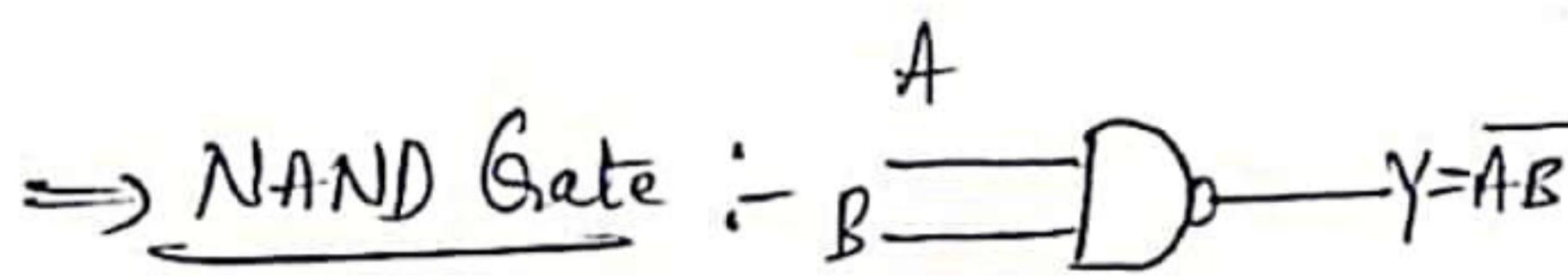
∴ These 3 gates
are primary
gates



A	B	$Y = A+B$
0	0	0
0	1	1
1	0	1
1	1	1



A	B	Y = $\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

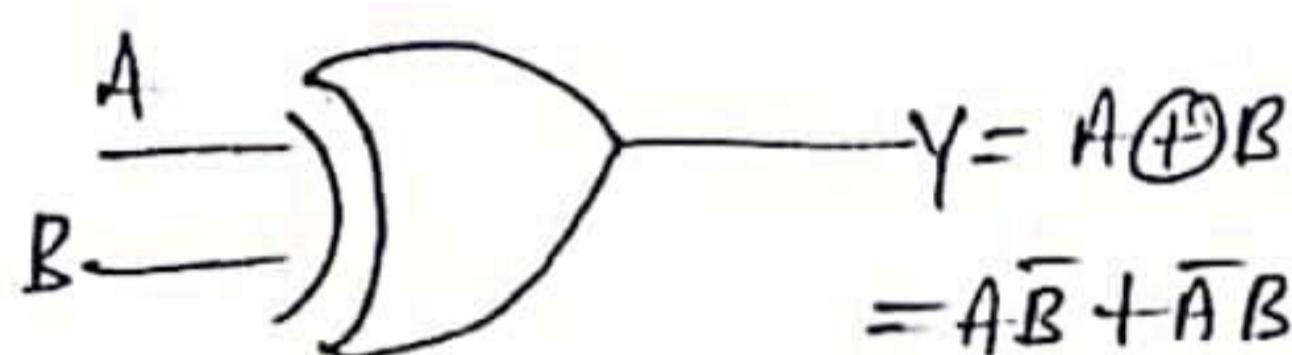


∴ The above two gates are universal gates

A	B	Y = \overline{AB}
0	0	1
0	1	1
1	0	1
1	1	0

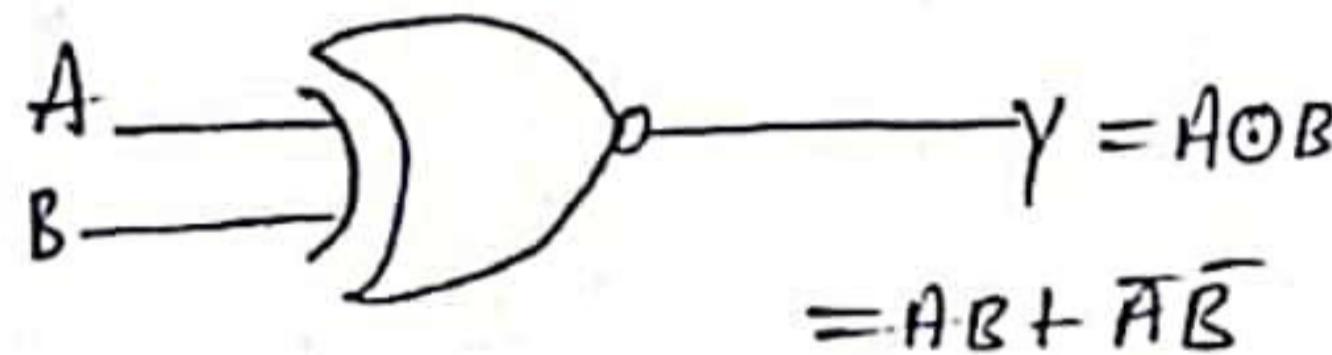
Special Gates :-

Ex-OR gate :-



A	B	Y = $A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Ex-NOR gate :-



A	B	Y = $A \ominus B$
0	0	1
0	1	0
1	0	0
1	1	1

Laws of Boolean Algebra :-

AND Operation

- 1) $0 \cdot 0 = 0$
- 2) $0 \cdot 1 = 0$
- 3) $1 \cdot 0 = 0$
- 4) $1 \cdot 1 = 1$

OR Operation

- 5) $0 + 0 = 0$
- 6) $0 + 1 = 1$
- 7) $1 + 0 = 1$
- 8) $1 + 1 = 1$

NOT Operation

- 9) $\overline{0} = 1$
- 10) $\overline{1} = 0$

⇒ Complement Law

$$\overline{0} = 1$$

$$1 = 0$$

If $A = 0$ then $\bar{A} = 1$

If $A = 1$ then $\bar{A} = 0$

$$\overline{\overline{A}} = A$$

AND Law

$$A \cdot 0 = 0$$

$$A \cdot 1 = A$$

$$A \cdot A = A$$

$$A \cdot \bar{A} = 0$$

Proof

$$= A \cdot A$$

$$= A \cdot A + 0$$

$$= A \cdot A + A \cdot \bar{A}$$

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

⇒ OR Law :-

$$A + 0 = A$$

$$A + 1 = 1$$

$$A + \bar{A} = 1$$

$$A + A = A$$

Proof

$$A + A = A$$

$$(A+A) 1$$

$$A+A (A+\bar{A})$$

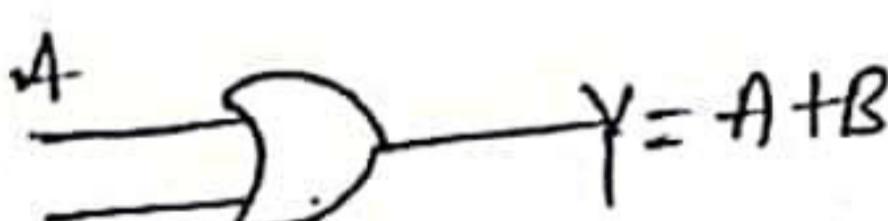
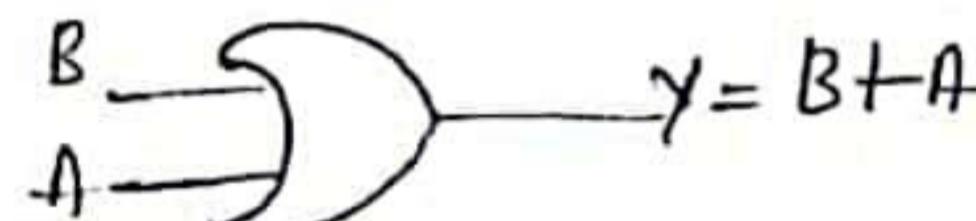
$$A+A \cdot A + A \cdot \bar{A}$$

$$= A + A \cdot A + 0$$

$$= A + (A+A) = A \quad (HA=1)$$

⇒ Commutative Law :-

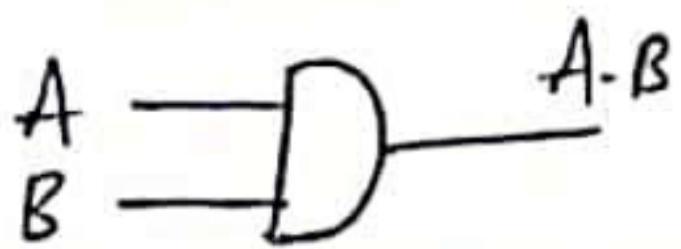
$$\textcircled{1} \quad A+B = B+A$$



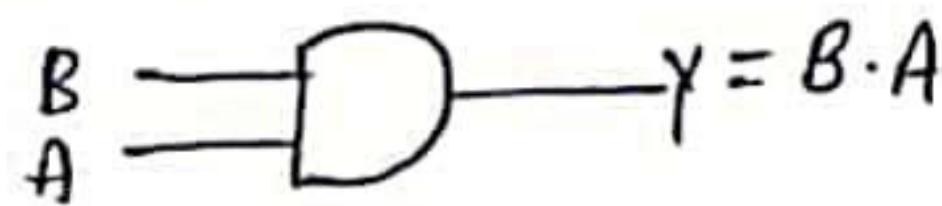
A	B	$A+B$
0	0	0
0	1	1
1	0	1
1	1	1

B	A	$B+A$
0	0	0
0	1	1
1	0	1
1	1	1

Law @ :- $A \cdot B = B \cdot A$



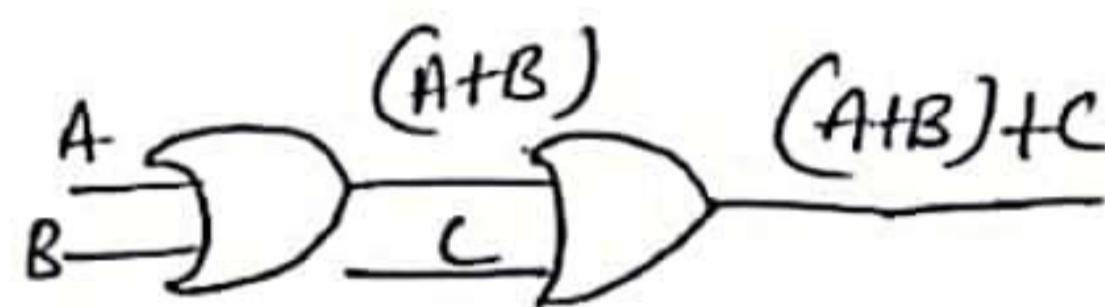
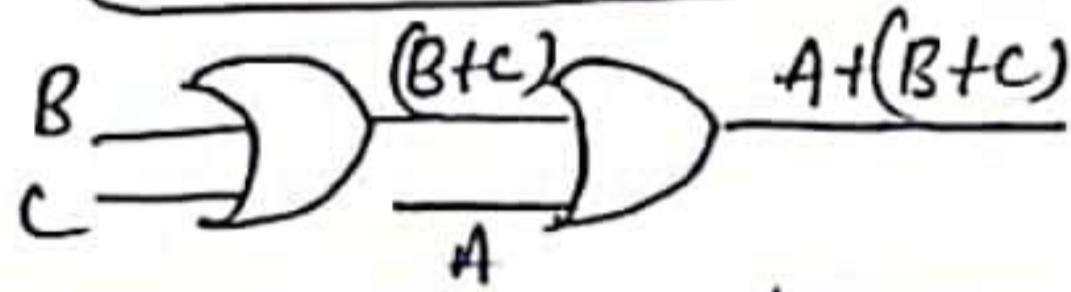
A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1



B	A	$B \cdot A$
0	0	0
0	1	0
1	0	0
1	1	1

⇒ Associative Law :-

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

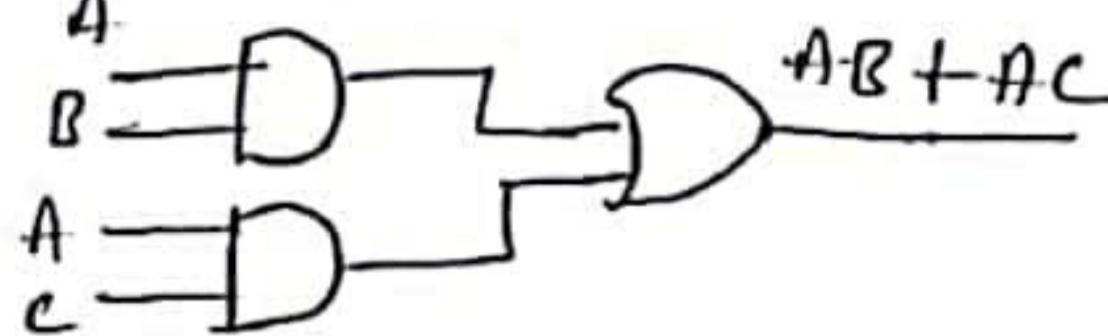
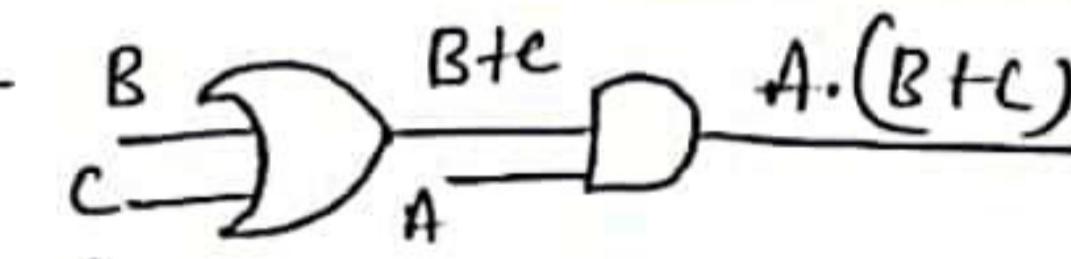
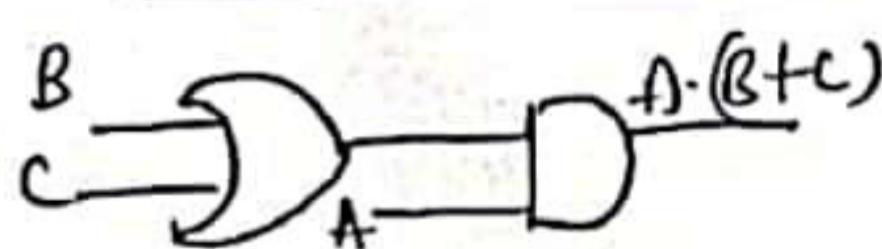


A	B	C	$(B+C)$	$A+(B+C)$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

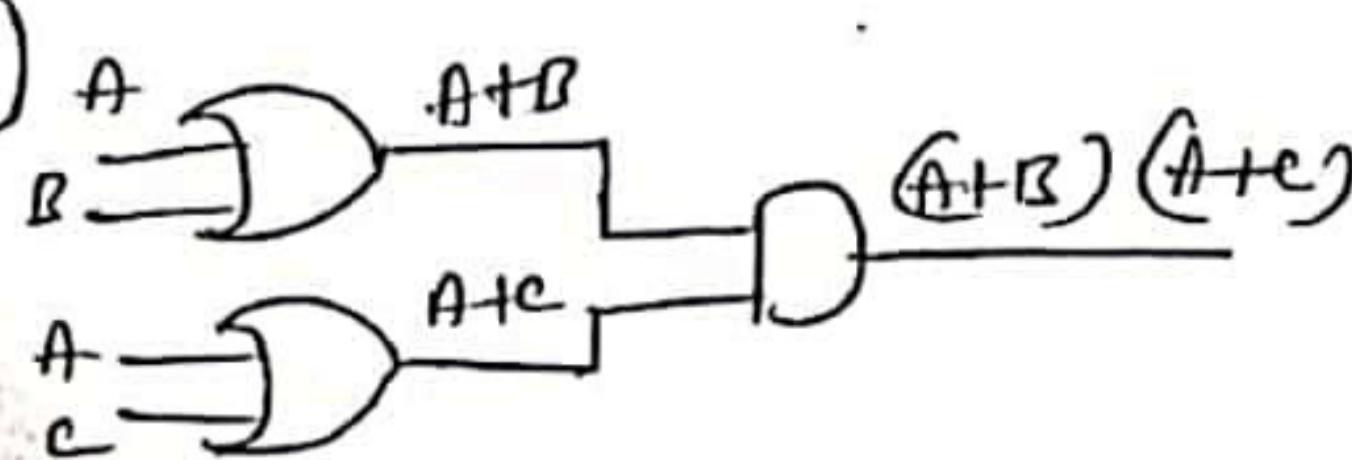
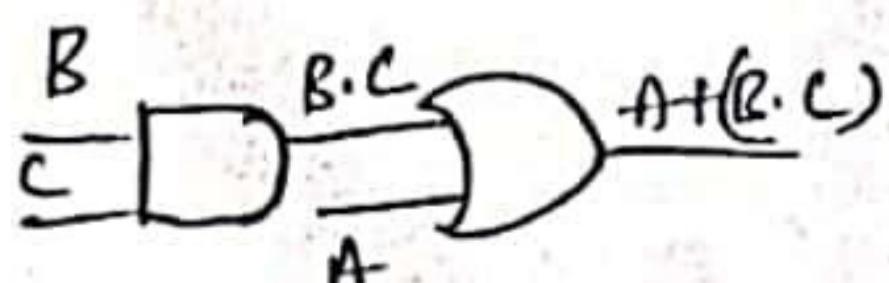
A	B	C	$(A+B)$	$(A+B)+C$
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

⇒ Distributive Law :-

$$\textcircled{1} \quad A \cdot (B+C) = AB + AC$$



$$\textcircled{2} \quad A + (B \cdot C) = (A+B)(A+C)$$



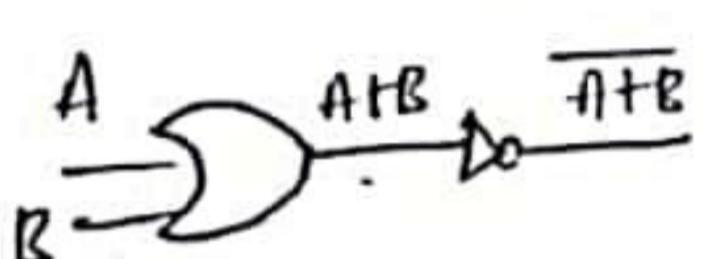
⇒ Consensus Theorem :-

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

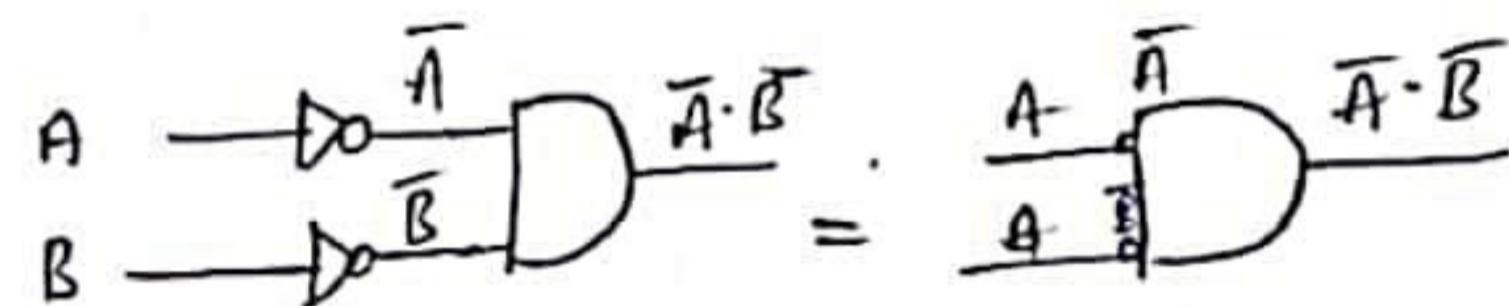
$$\begin{aligned}
 L.H.S &= AB + \bar{A}C + BC \\
 &= AB + \bar{A}C + BC(A + \bar{A}) \\
 &= AB + \bar{A}C + BCA + BC\bar{A} \\
 &= ABC(1+C) + \bar{A}C(1+B) \\
 &= AB(1) + \bar{A}C(1) \\
 &= AB + \bar{A}C \\
 &= R.H.S \quad \therefore \boxed{L.H.S = R.H.S}
 \end{aligned}$$

⇒ Demorgan's Theorem :-

$$\textcircled{1} \quad \overline{A+B} = \bar{A} \cdot \bar{B}$$

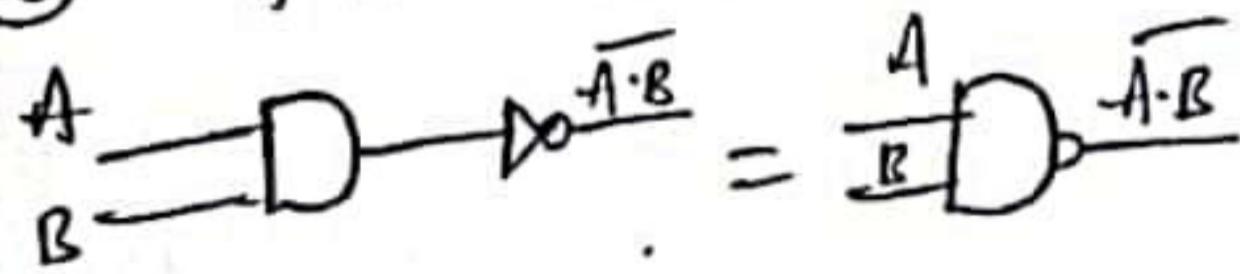


A	B	$\overline{A+B}$	$\bar{A} \cdot \bar{B}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

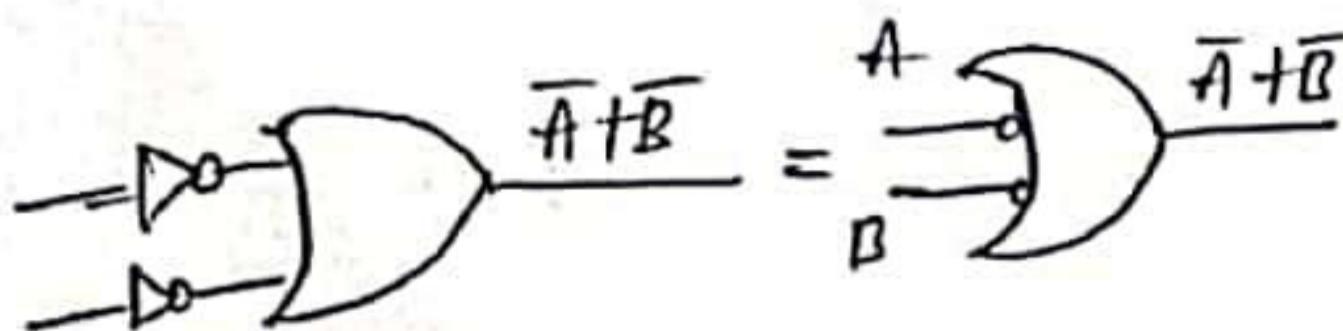


A	B	\bar{A}	\bar{B}	$\bar{A} \cdot \bar{B}$
0	0	1	1	1
0	1	1	0	0
1	0	0	1	0
1	1	0	0	0

$$\textcircled{2} \quad \overline{A \cdot B} = \bar{A} + \bar{B}$$



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



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

⇒ Duality :- when changing from one logic system to another system; 0 becomes 1 and 1 becomes 0. An AND gate becomes OR gate and OR gate becomes an AND gate.

⇒ Complement :- If a boolean identity is given, we should change '+' sign '•' sign and '•' sign to '+' sign. Variables (A, B) also complemented.

Note :- ① $\overbrace{AB + AB + AB + AB}^{\text{parenthesis}} = AB$ (Same no of variables are repeated we may consider single term)

② $A \cdot B \cdot \bar{B} = A \cdot 0 = 0$; ③ $AB\bar{C}\bar{C} = AB \cdot 0 = 0$

④ $ABC\bar{C} + ABC\bar{D} + ABC\bar{D}$
 $ABC(1+D) = ABC(1) = ABC$.

④ $ABC\bar{D} + ABC\bar{D}$
 $= ABC(\bar{D} + \bar{D})$
 $= ABC(1) (\because \bar{D} + \bar{D} = 1)$

① $f = A + B [AC + (B + \bar{C})D]$

$$= A + B [AC + BD + \bar{C}D]$$

$$= A + [ABC + \underbrace{BD}_{\substack{\text{repeated terms; we can consider as single.} \\ \text{BD}}} + \bar{C}D]$$

$$= A(1 + BC) + BD(1 + \bar{C})$$

$$\boxed{f = A + BD}$$

$$\therefore \frac{BD}{\substack{\text{repeated terms; we can consider as single.}}} = BD$$

$$\therefore 1 + BC = 1; 1 + \bar{C} = 1$$

② $f = (\overline{A + \bar{B}C}) \cdot (\bar{A}\bar{B} + A\bar{B}C)$

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

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

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

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

$$= \cancel{0} + \cancel{0} \quad \boxed{f = 0}$$

$$\therefore \bar{A}\bar{B} = 0; \bar{B}\bar{B} = 0$$

Hence the boolean expression has been reduced by boolean theorem.

Q Write the duality for the following functions

① $\bar{A}B + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}CD + \bar{A}\bar{B}\bar{C}\bar{D}E$

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

② $\bar{a}yz + a\bar{y}\bar{z} + a\bar{y}z + a\bar{y}\bar{z}$

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

Q Find the complements of the following expressions

① $AB + A(B+C) + \bar{B}(B+D)$

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

② $\bar{B}\bar{C}D + (\overline{B+C+D}) + \bar{B}\bar{C}\bar{D}\bar{E}$

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

→ Karnaugh Map (K-map) Representation :-

① Sum of product (SOP) $\Sigma m = \bar{A}B + A\bar{B}$

② Product of sum (POS) $\Sigma m = (A+B)(\bar{A}+C)$

① Sum of product (SOP) :- This is also called as Disjunctive Normal form (DNF). Variables present in this variables are called 'minterms' (m_0, m_1, m_2, \dots)

$$\text{Ex. } f(A, B, C) = m_1 + m_2 + m_3 + m_5 = \Sigma m(1, 2, 3, 5)$$

⇒ Standard SOP form :- (SOP) It is also called as Disjunctive Canonical form (DCF)

② Product of sum (POS) :- Is also called as conjunctive normal form (CNF). Variables present in this form is called 'maxterms' (M_1, M_2, M_3, \dots)

$$\text{Ex: } f(A, B, C) = \prod(M_1 + M_2 + M_6, M_7)$$

$$= \prod(M_1, M_2, M_6, M_7)$$

\Rightarrow Standard POS form \rightarrow This form is also called as conjugate Canonical form (CCF)

$$\text{Ex: } f(A, B, C) = (\bar{A} + \bar{B})(\bar{A} + B)$$

$$= (\bar{A} + \bar{B} + C \cdot \bar{C})(\bar{A} + B + C \cdot \bar{C}) \quad | \because C \cdot \bar{C} = 0.$$

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

Ex: Convert SOP to standard SOP form

$$f(A, B, C) = \bar{A}C + AB + BC$$

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

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

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

Repeated ABC product
• In here we should write only one

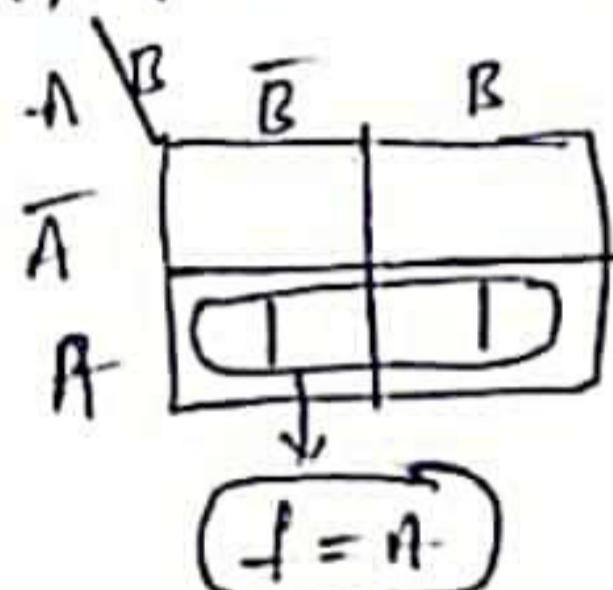
Decimal no.	A	B	C	minterms	Minterms
0	0	0	0	$\bar{A}\bar{B}\bar{C}$ (m_0)	$A + B + C$ (M_0)
1	0	0	1	$\bar{A}\bar{B}C$ (m_1)	$A + B + \bar{C}$ (M_1)
2	0	1	0	$\bar{A}BC$ (m_2)	$A + \bar{B} + C$ (M_2)
3	0	1	1	$\bar{A}BC$ (m_3)	$A + \bar{B} + \bar{C}$ (M_3)
4	1	0	0	$A\bar{B}\bar{C}$ (m_4)	$\bar{A} + B + C$ (M_4)
5	1	0	1	$A\bar{B}C$ (m_5)	$\bar{A} + B + \bar{C}$ (M_5)
6	1	1	0	ABC (m_6)	$\bar{A} + \bar{B} + C$ (M_6)
7	1	1	1	ABC (m_7)	$\bar{A} + \bar{B} + \bar{C}$ (M_7)

- ① Either group zeros & ones
- ② Diagonal mapping is not allowed
- ③ Only powers of 2, no. of cells in each group (i.e. 2, 4, 6, 8....)
- ④ Group should be as large as possible
- ⑤ Overlapping is allowed.

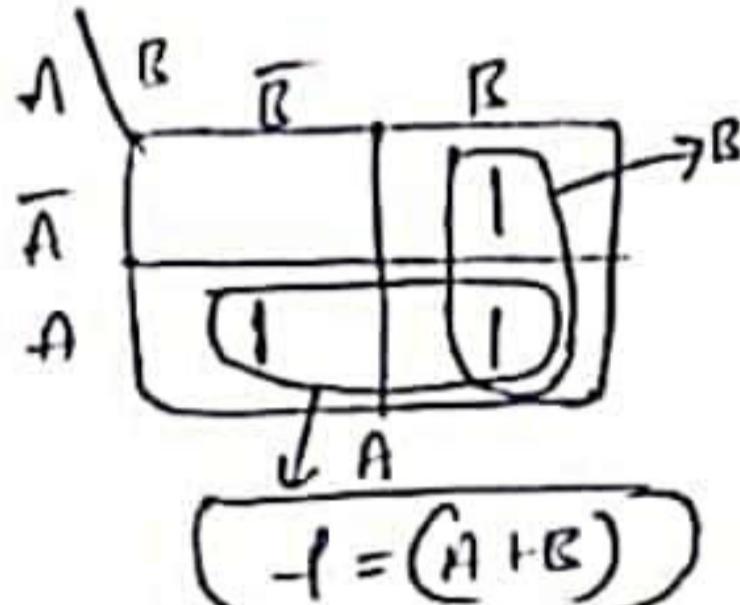
Q Problems on K-map representation

2-variable K-map (SOP)

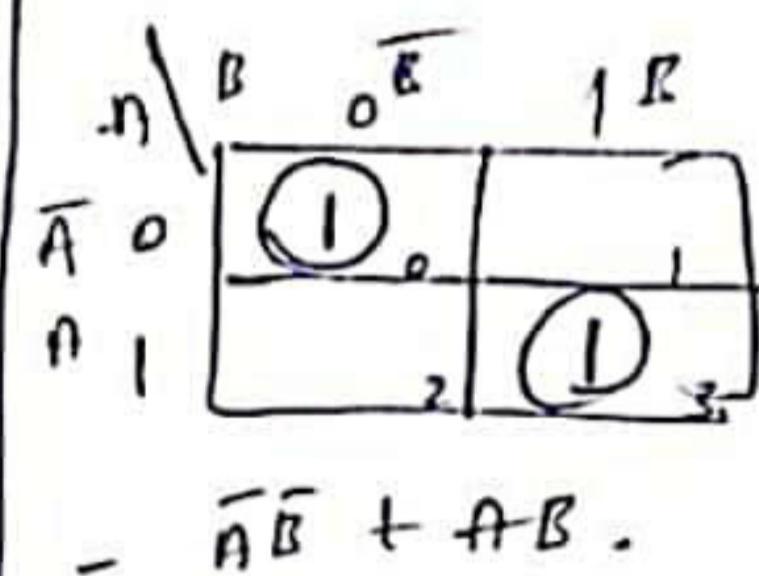
i, $f = A\bar{B} + A\cdot B$



ii, $f = A\bar{B} + AB + \bar{A}B$



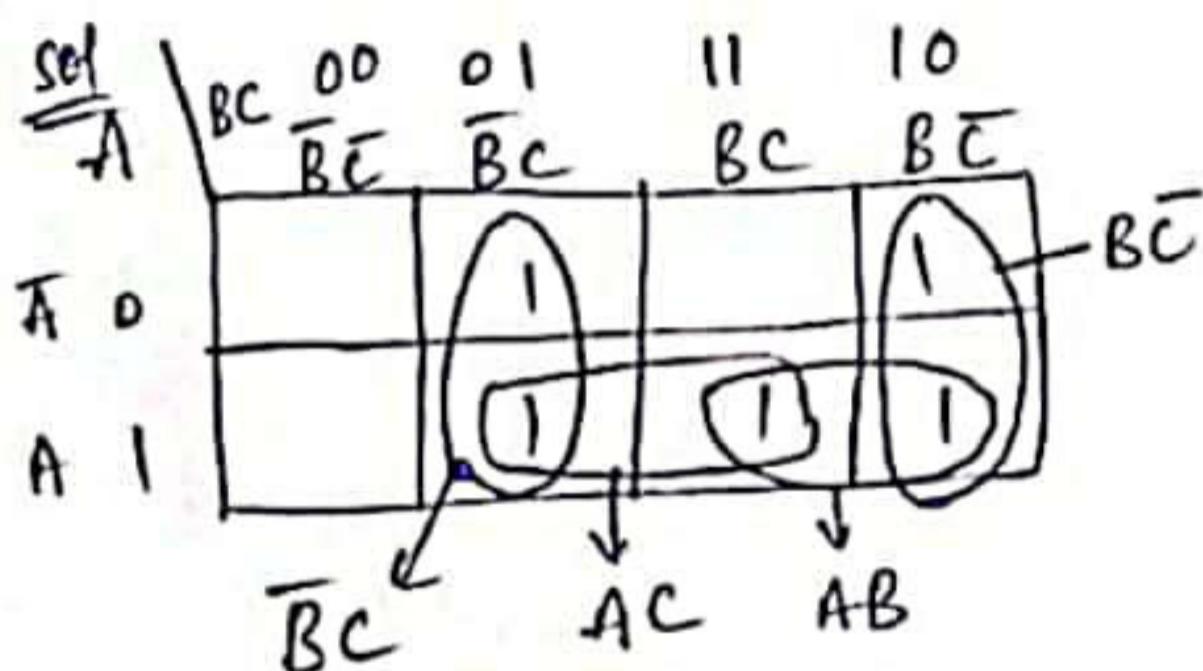
iii, $f(A, B) = (0, 3)$



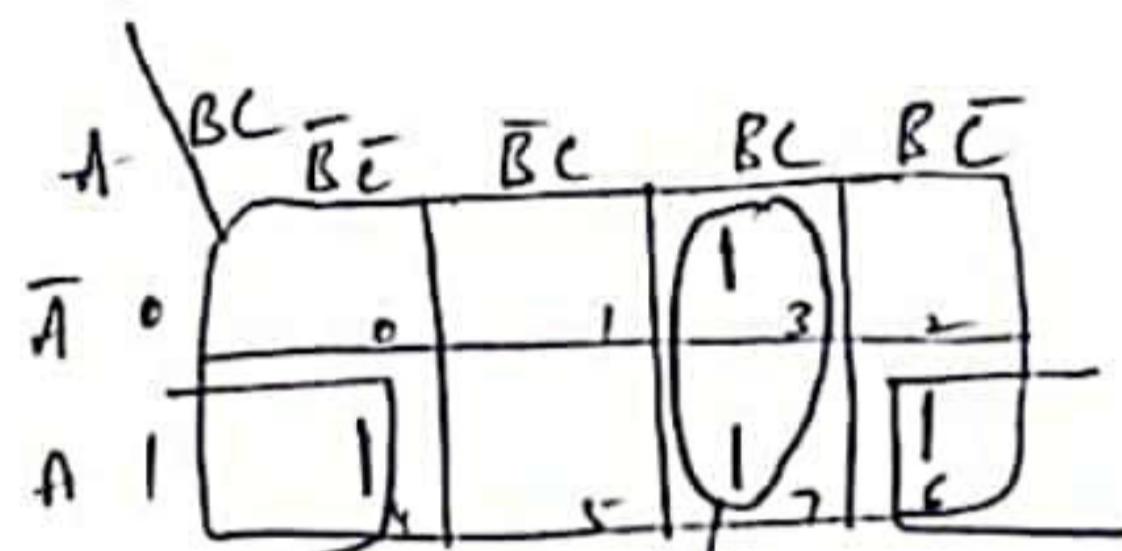
$$= \bar{A}\bar{B} + A\cdot B.$$

Q $f = \bar{A}\bar{B}\bar{C} + A\bar{B}C + \bar{A}BC + ABC + ABC$

$$f(A, B, C) = \Sigma m(3, 4, 6, 7)$$



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

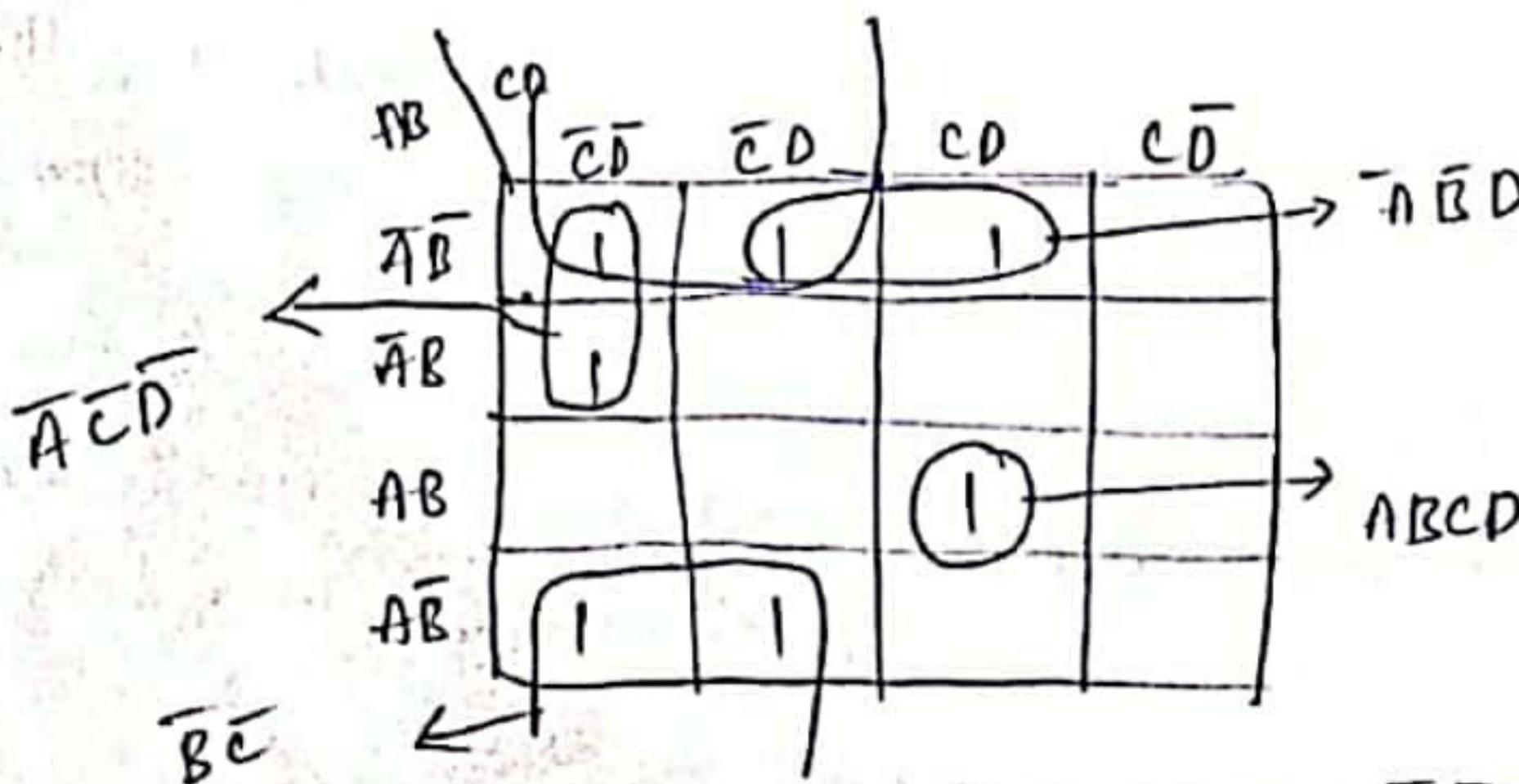


$$\bar{f} = BC + A\bar{C}$$

3-variable K-map

Q Reduce the below expression using K-map

$$f(A, B, C, D) = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}CD + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + ABCD + \bar{A}BC\bar{D}$$



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

Q

Convert the following in the SOP form and calculate the minterms

(a)

$$f(A, B) = \bar{A}B + B$$

Sol Given $f(A, B) = \bar{A}B + B$

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

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

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

If same digits

are more than
Two then it
becomes one)

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

$$= \begin{matrix} \downarrow & \downarrow \\ 0 & 1 \end{matrix} + \begin{matrix} \downarrow \\ 1 \end{matrix}$$

$$= m_1 + m_3$$

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

(b) $f(A, B, C) = ABC + A\bar{B}C + AB$

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

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

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

$$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 1 & 0 & 1 & 0 & 1 \end{matrix} \quad \begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 6 & 5 & 7 & 1 & 1 & 0 \end{matrix}$$

$$= m_6 + m_5 + m_7 + m_1$$

If B should be in proper order

$$\therefore m_1 + m_5 + m_6 + m_7$$

$$\Rightarrow \Sigma m(3, 5, 6, 7)$$

Q

convert the following in the SOP form and calculate the minterms.

(a)

$$f(A, B) = A(\bar{A} + \bar{B})$$

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

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

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

$$= M_0 \quad M_1 \quad M_2$$

$$= \Pi M(0, 1, 2)$$

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

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

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

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

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

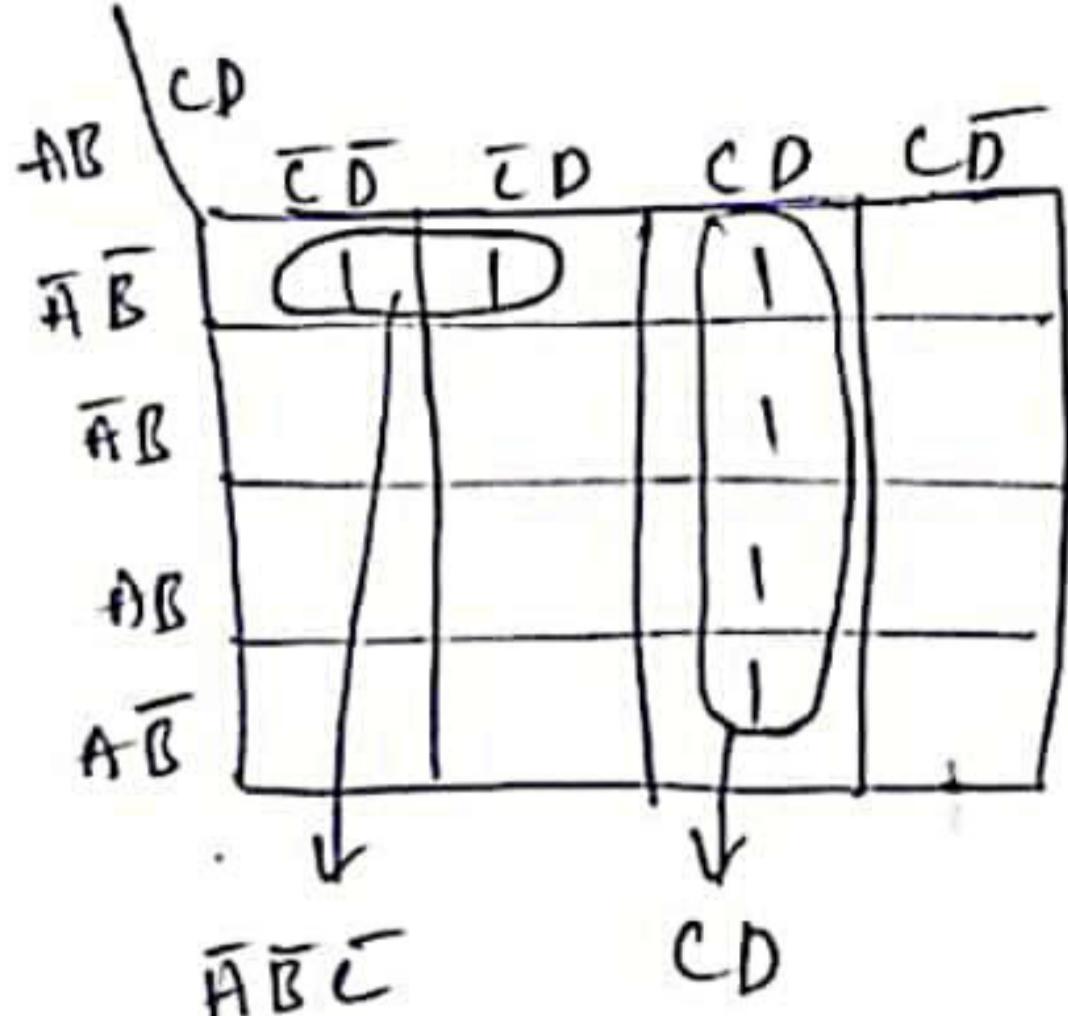
$$= (\bar{A} + B)(\bar{A} + \bar{B})(\bar{A} + \bar{B}) \quad (\because (\bar{A} + B)(\bar{A} + \bar{B}) = (\bar{A} + B))$$

$$M_0 \cdot M_1 \cdot M_2 = \Pi M(0, 1, 2)$$

Note :- K-map consist of a no. of squares. Each one of the square is cell. To do K-map minimization the expression should be in SOP form or POS form.

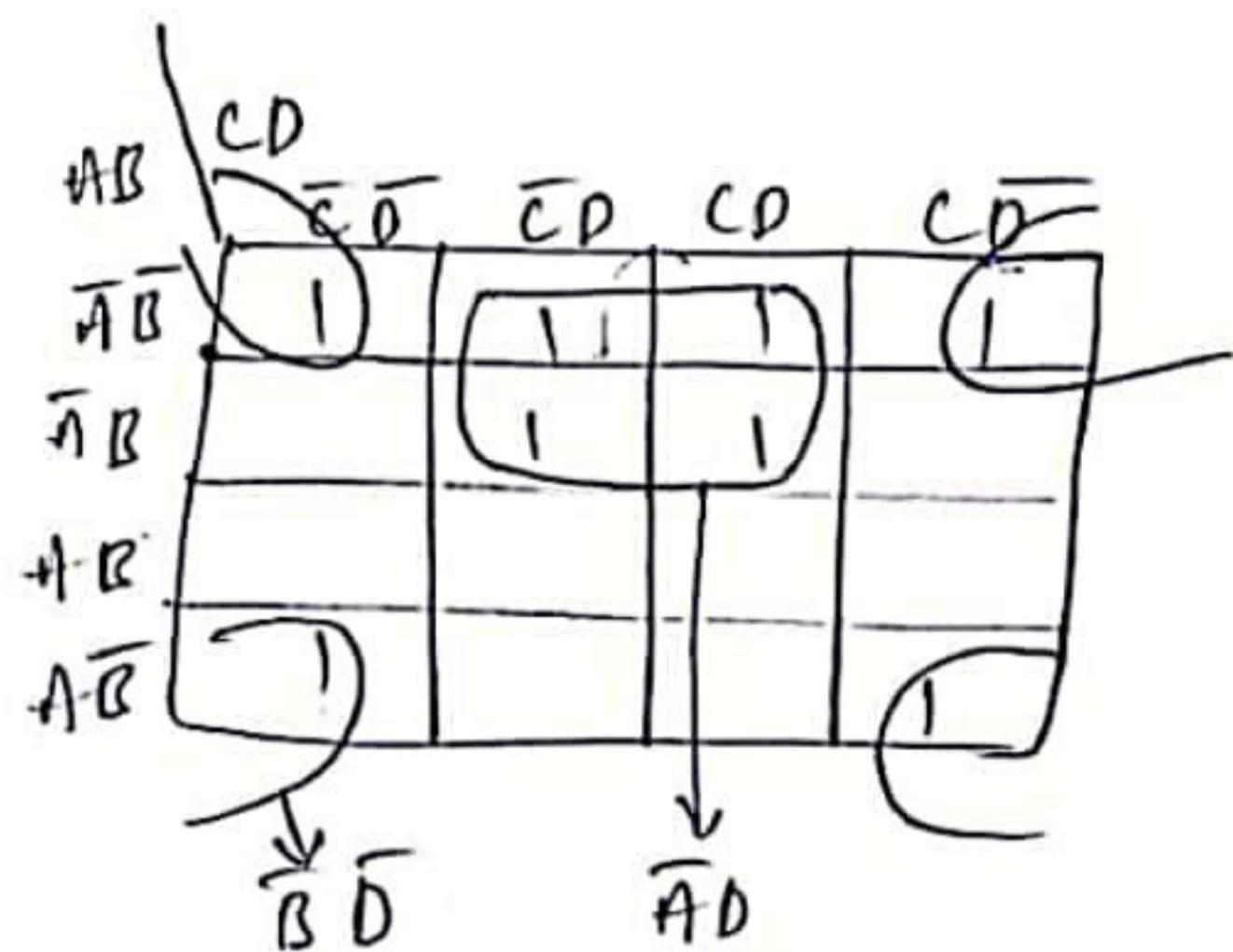
This is extremely useful and intensively used in the minimization of function of 2-variable K-map, 3-variable K-map, 4-variable K-map and so on.

Q Simplify, $f(A, B, C, D) = \Sigma m(0, 1, 3, 7, 11, 15)$



$$\therefore f = \bar{A}\bar{B}C + CD$$

Q $f(A, B, C, D) = \Sigma m(0, 1, 2, 3, 7, 8, 10)$



$$\therefore f = \bar{B}\bar{D} + \bar{A}D.$$

Minimization of Boolean Expression with help of POS by using K-map

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

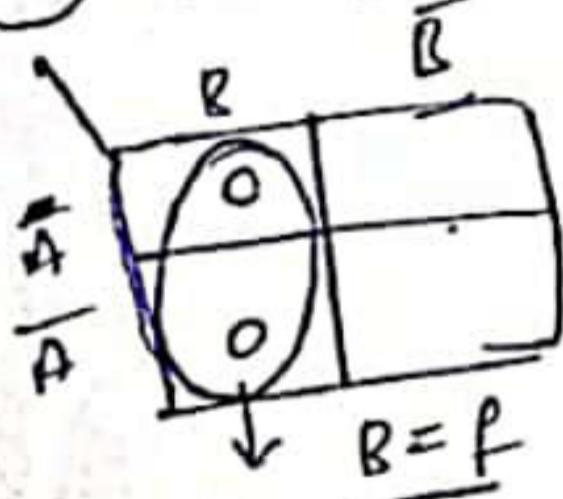
2-variable

A	B	0	1	2	3
0	$B\bar{C}$	$B+C$	$\bar{B}+C$	$\bar{B}+C$	$\bar{B}+C$
1	$A+B\bar{C}$	$A+B+C$	$A+\bar{B}+C$	$A+\bar{B}+C$	$A+\bar{B}+C$

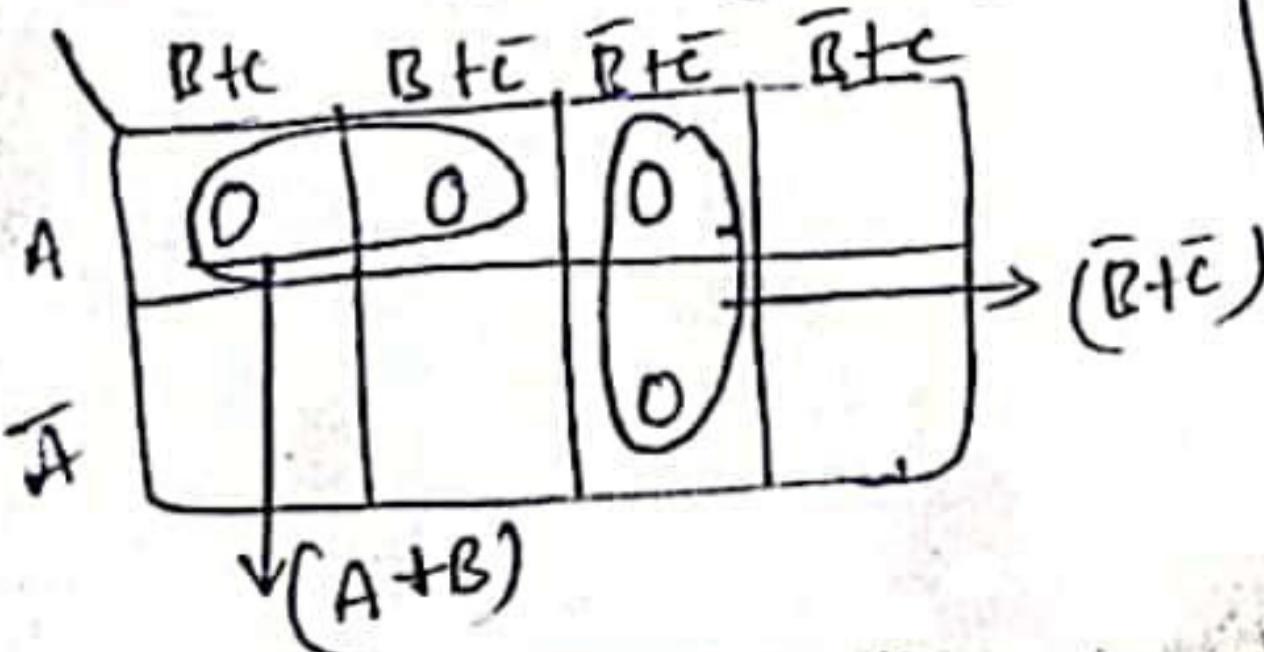
3-variable

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

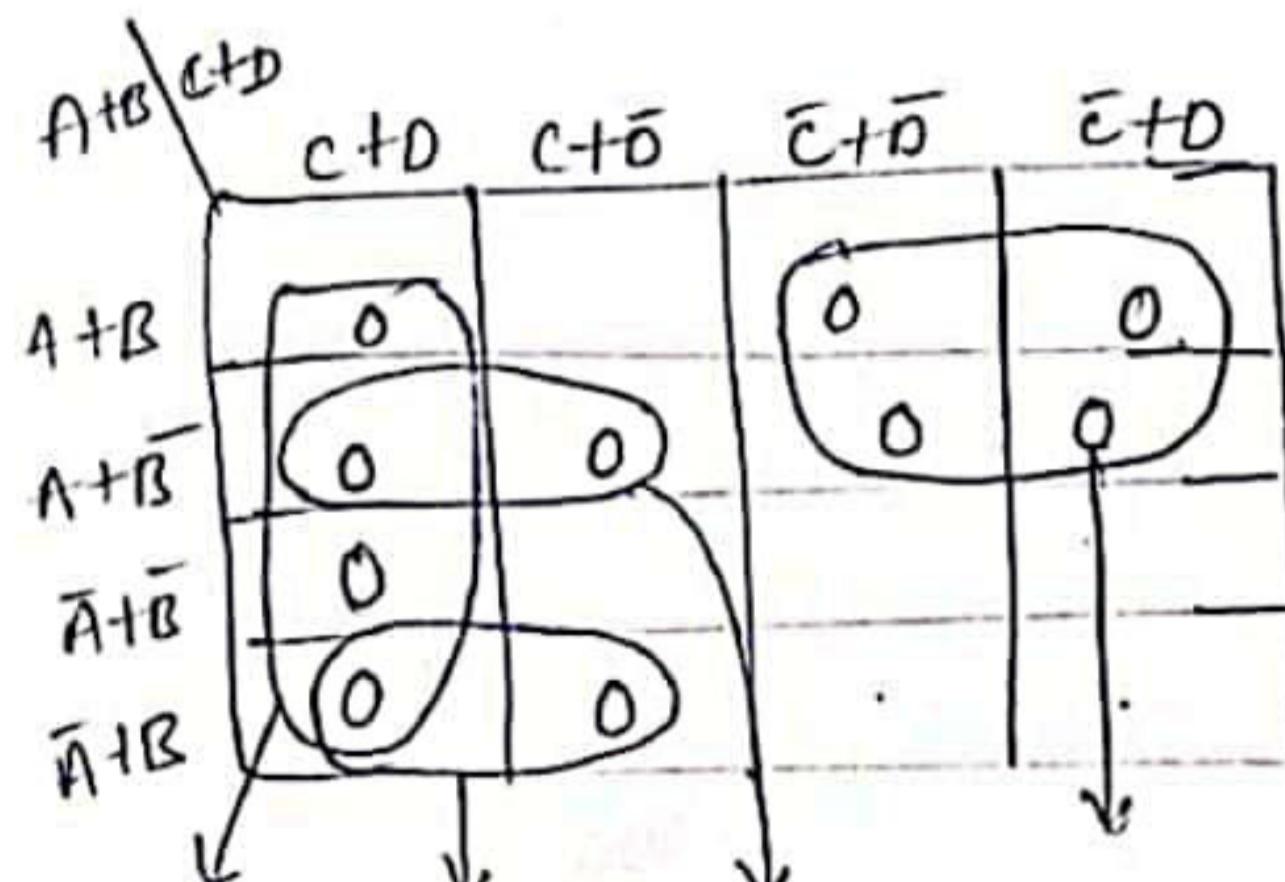
(Q) $f(A, B) = (A+B)(\bar{A}+B)$



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



Q $f = \prod M(0, 2, 3, 4, 5, 6, 7, 9, 12, 8)$



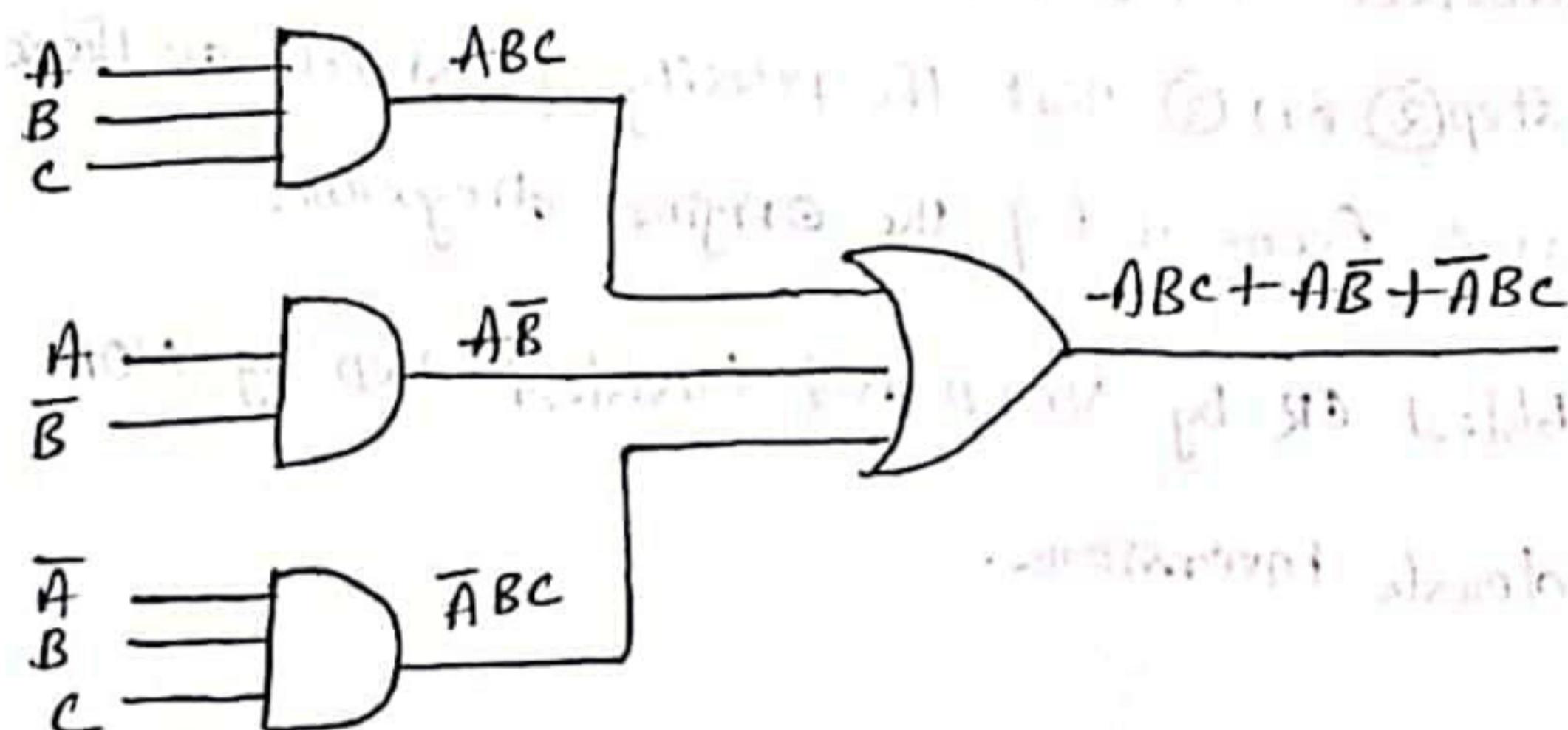
$$f = (C+D) \cdot (\bar{A}+B+C) \cdot (A+\bar{B}+C) \cdot (A+\bar{C})$$

X) AND and NOR Implementation

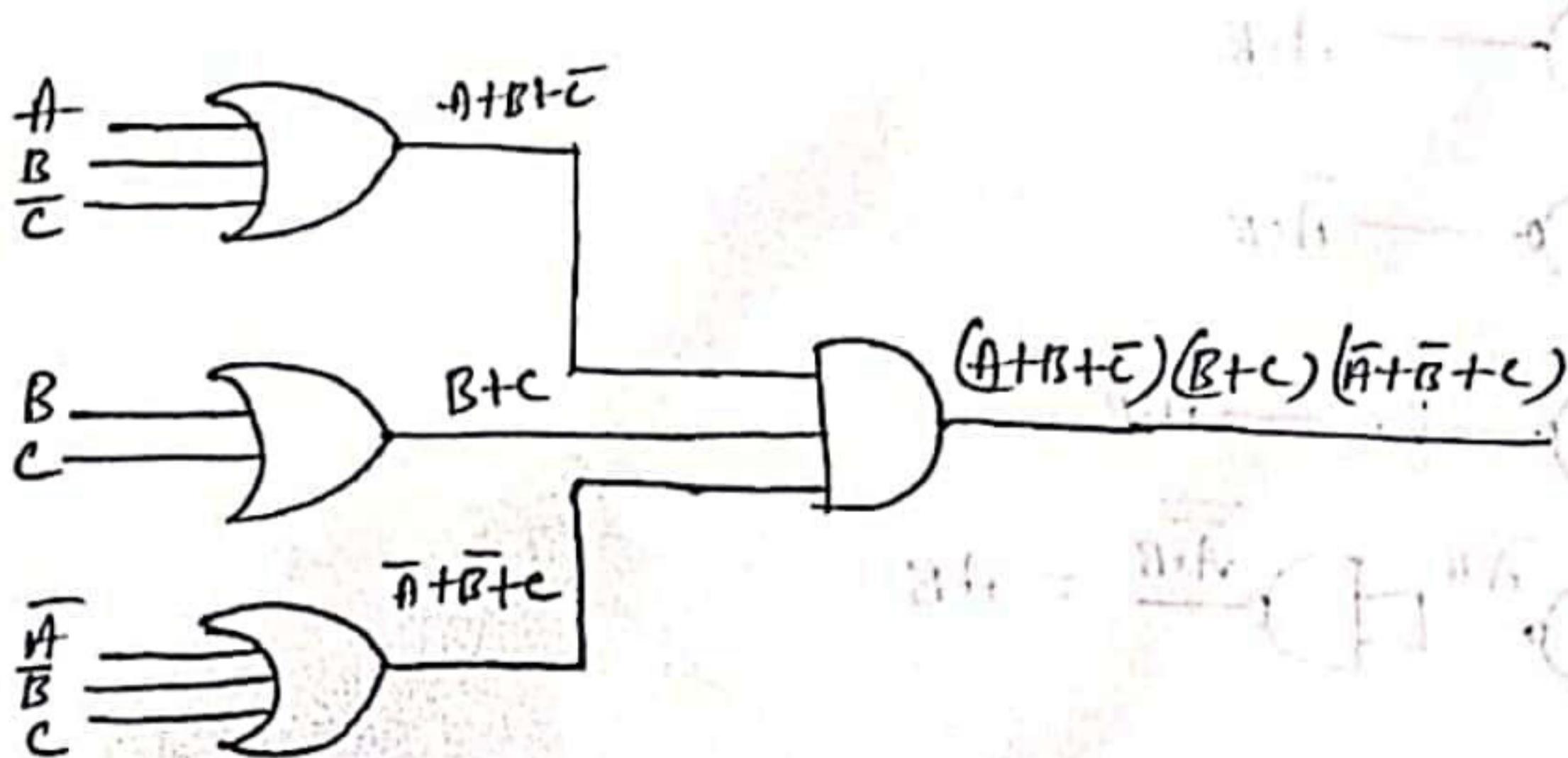
126

In the design of digital circuits, the minimal boolean expressions are usually obtained in SOP-form (or) POS-form. Sometimes the minimal expressions may also be expressed in hybrid form.

For example $f = ABC + A\bar{B} + \bar{A}BC$
 Given example is in SOP-form. So, SOP expression can be implemented by using AND/OR logic as shown below.



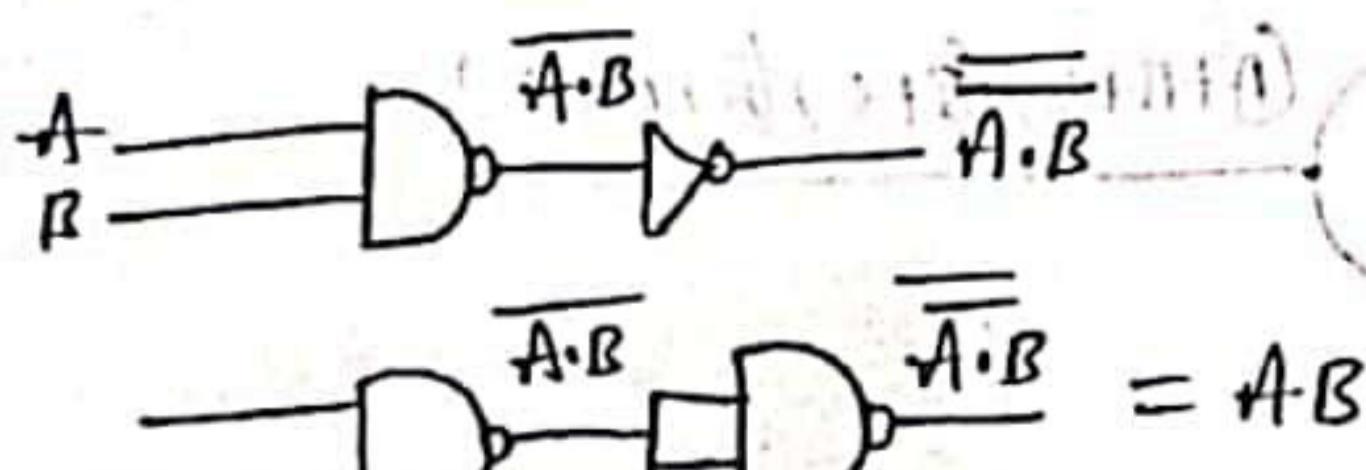
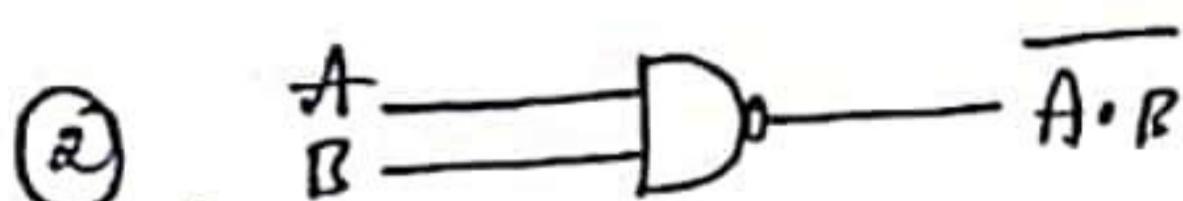
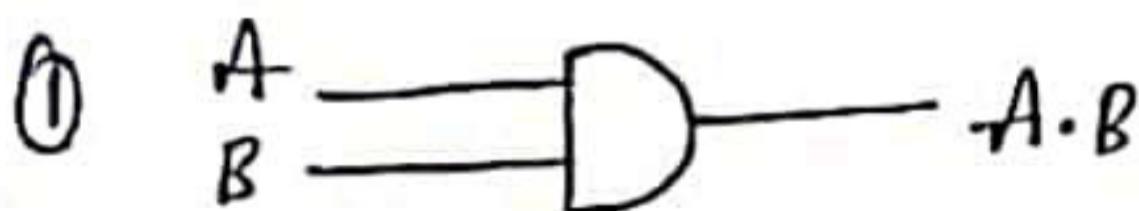
The form of given expression is $f = (A+B+\bar{C})(B+C)(\bar{A}+\bar{B}+C)$. This can be implemented using OR/AND logic as shown below.



The procedure to convert an AOI logic to NAND logic (or) NOR logic is given below.

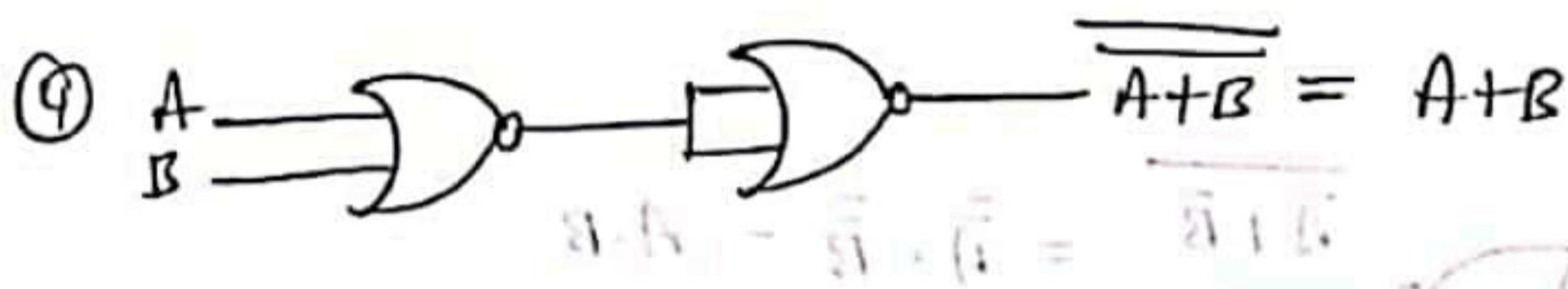
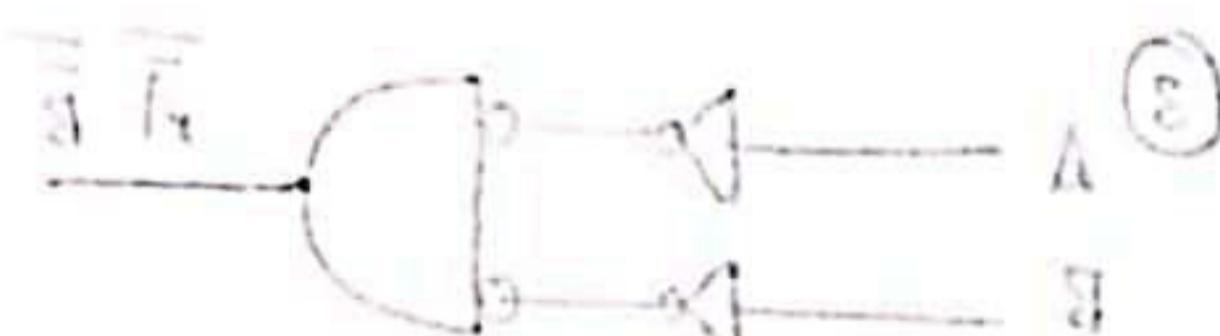
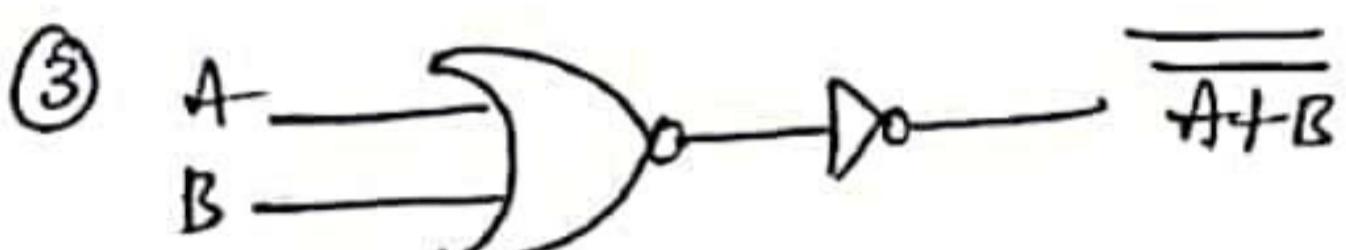
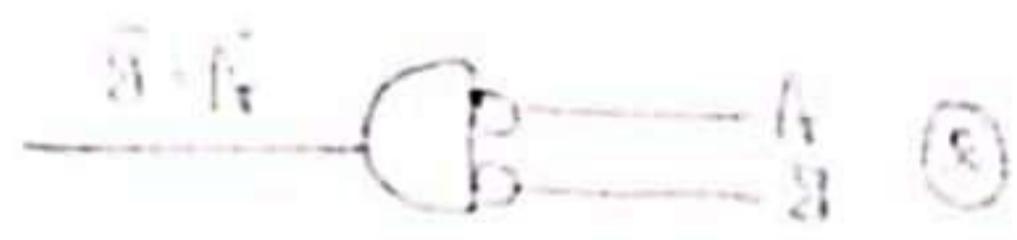
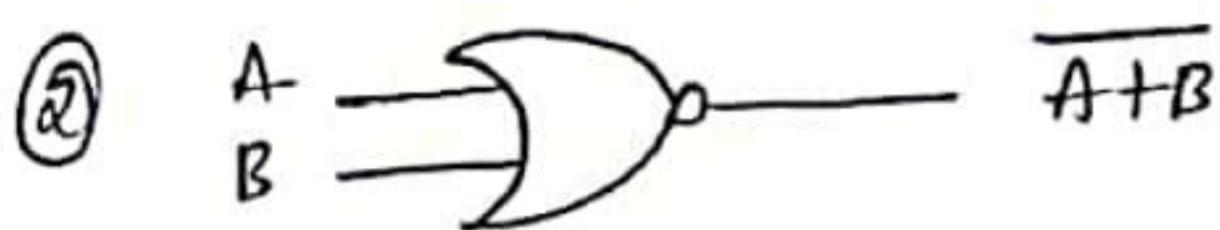
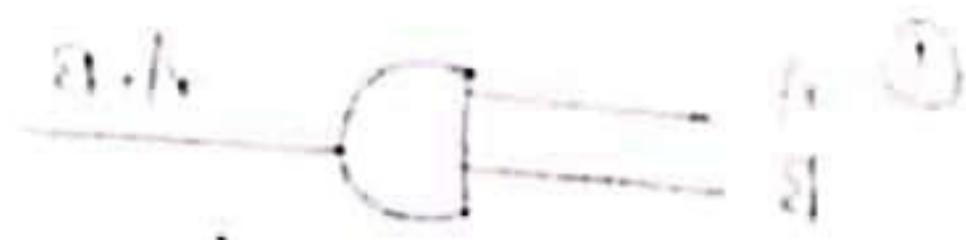
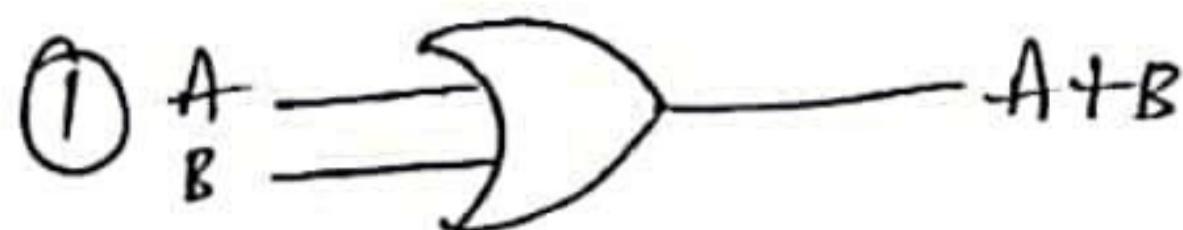
- ① Draw the circuit to AOI logic
- ② If NAND hardware is chosen, add a circle at the output of each AND gate and at the inputs to all the OR gates.
- ③ If NOR hardware is chosen, add a circle at the output of each OR gate and at the inputs to all the AND gates.
- ④ Add (or) subtract an inverter on each line that received a circle in step ② or ③ that the polarity of signals on those lines remains from that of the original diagram.
- ⑤ Replace bubbled OR by NAND and bubbled AND by NOR
- ⑥ Eliminate double inversions.

① Implementation of AND gate Using NAND Gate :-



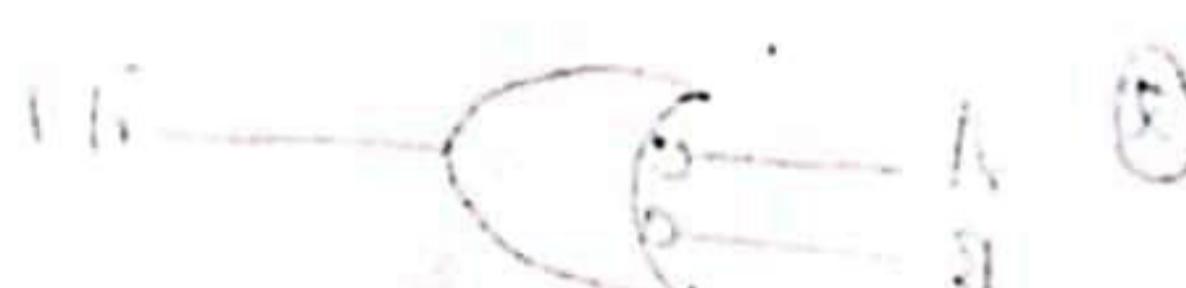
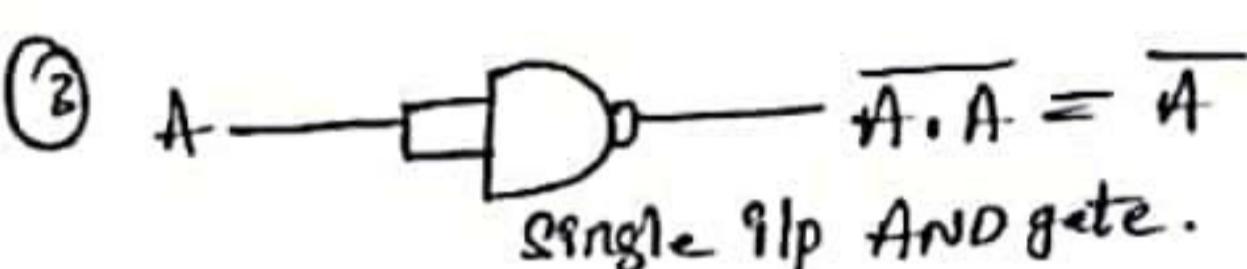
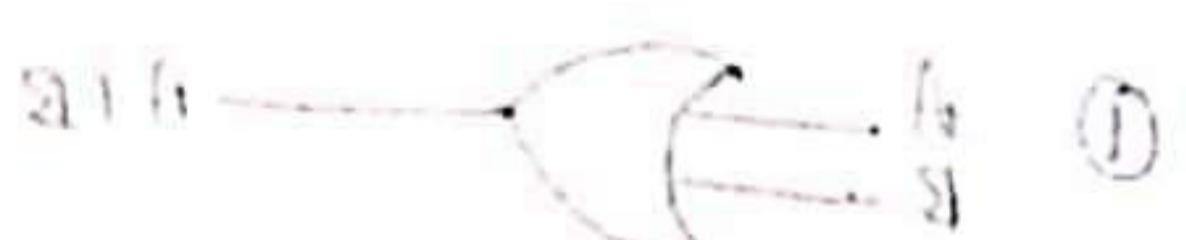
④ OR gate using NOR

128

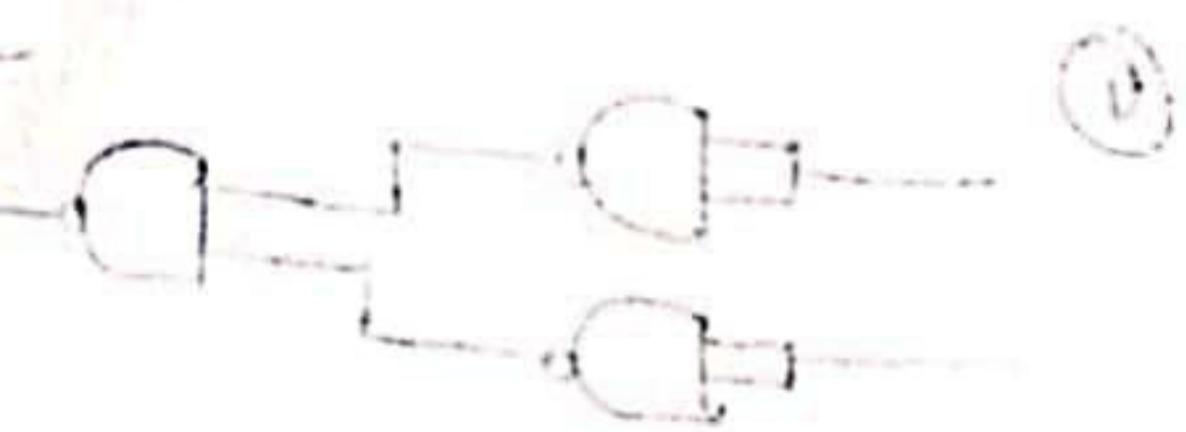
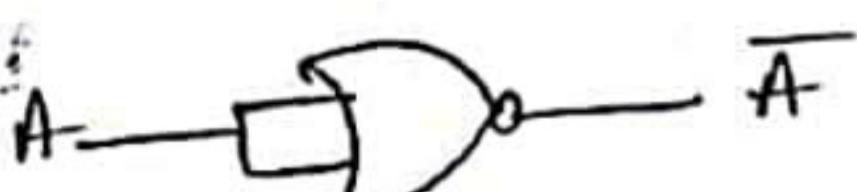
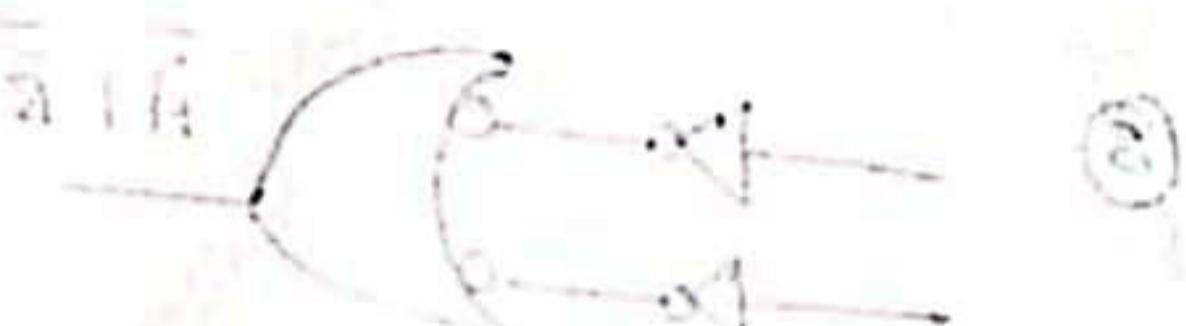
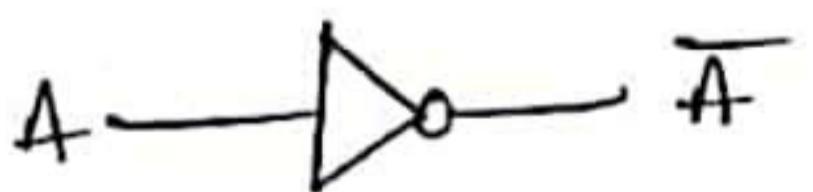


$$A \cdot B = \overline{\overline{A} + \overline{B}} = \overline{A + B}$$

⑤ Realization of NOT gate Using NAND & NOR



⇒ Using NOR

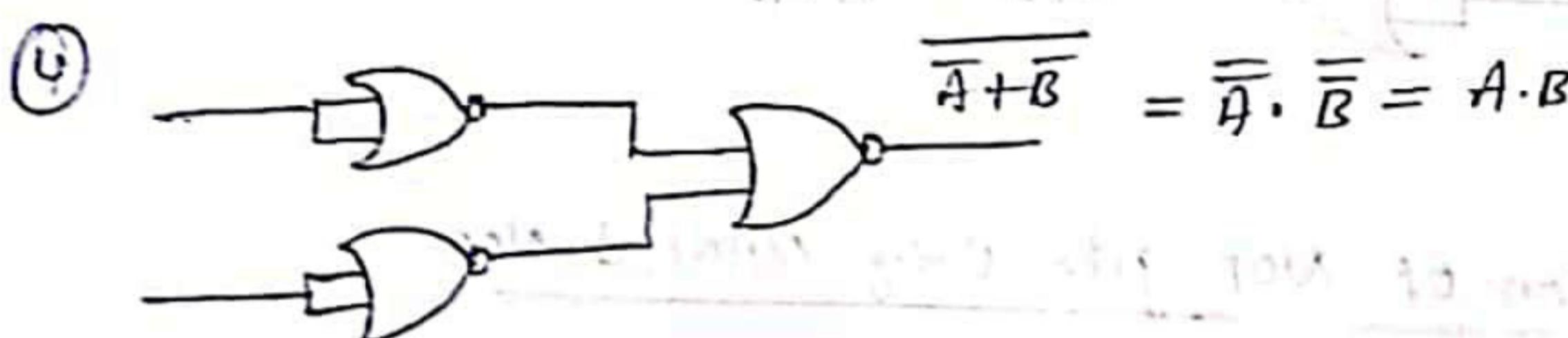
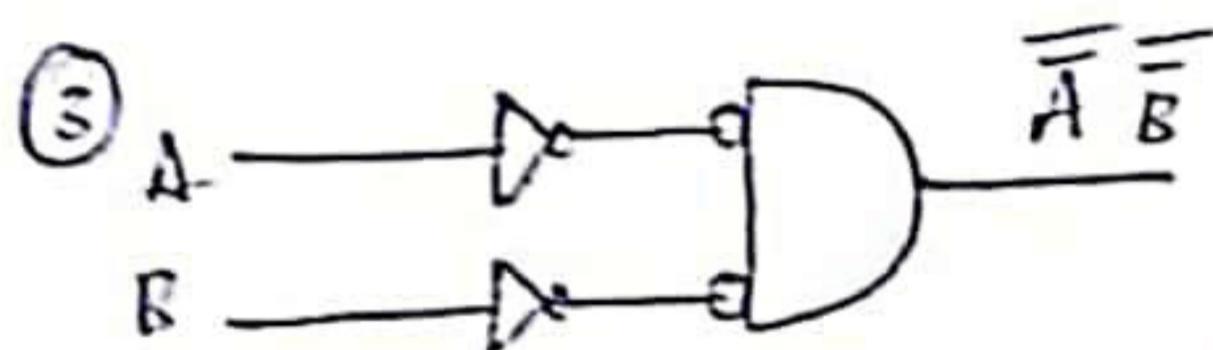
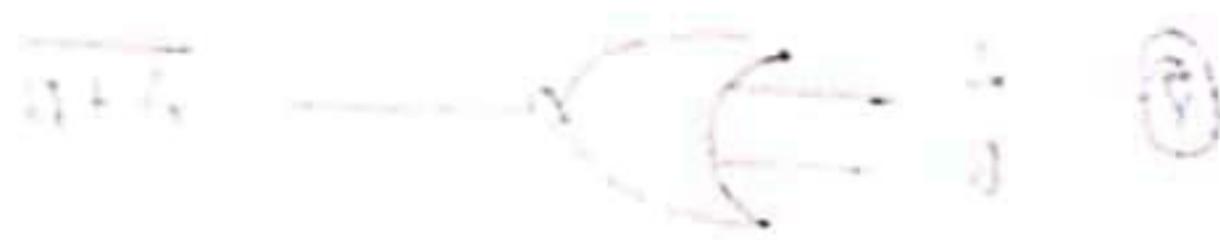
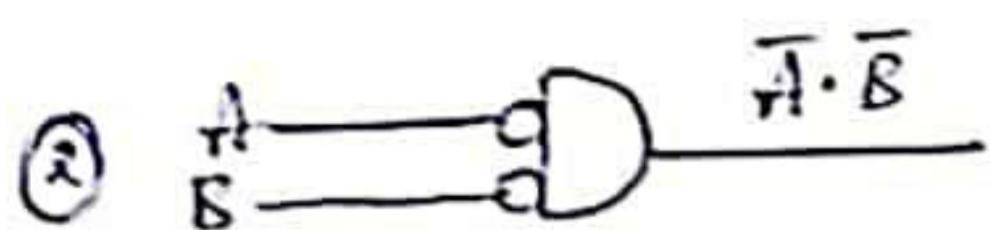
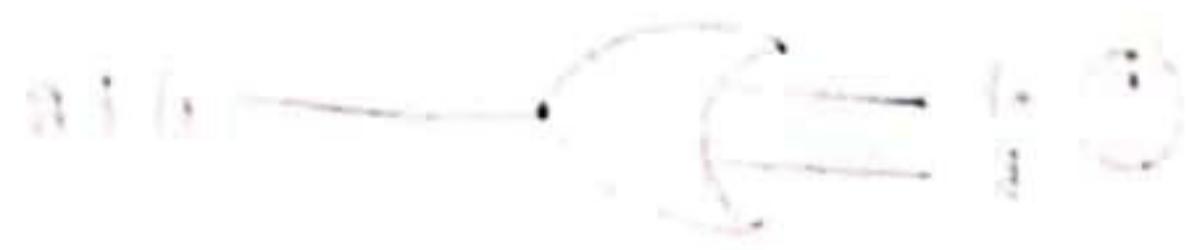
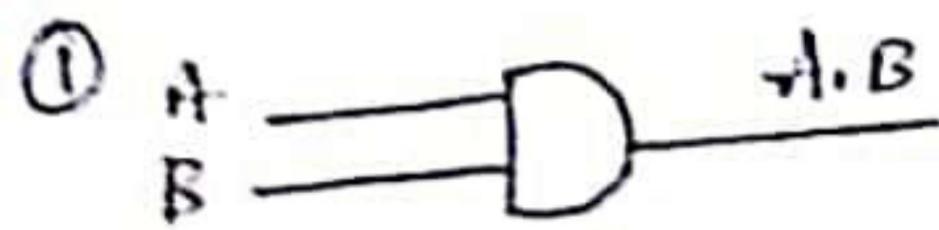


$$A \rightarrow \overline{A} = \overline{A \cdot A} = \overline{A} + \overline{A} = \overline{A}$$

Single input NOR gate.

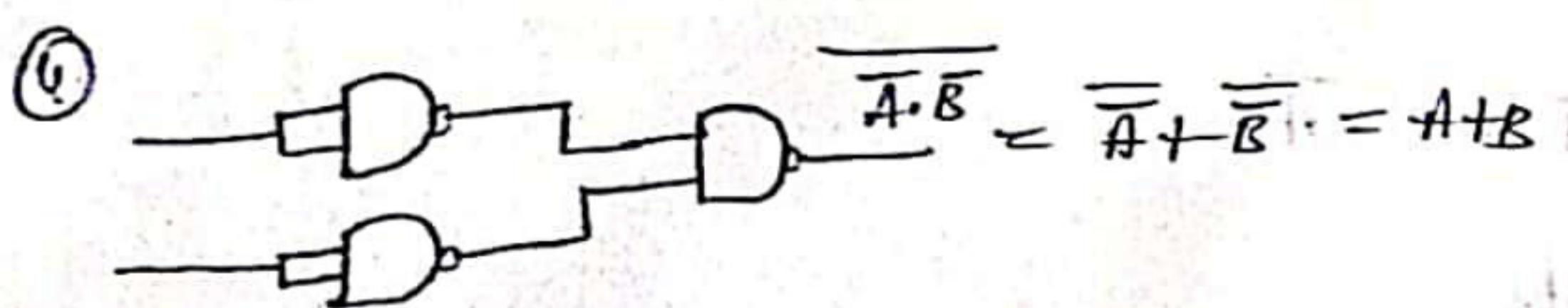
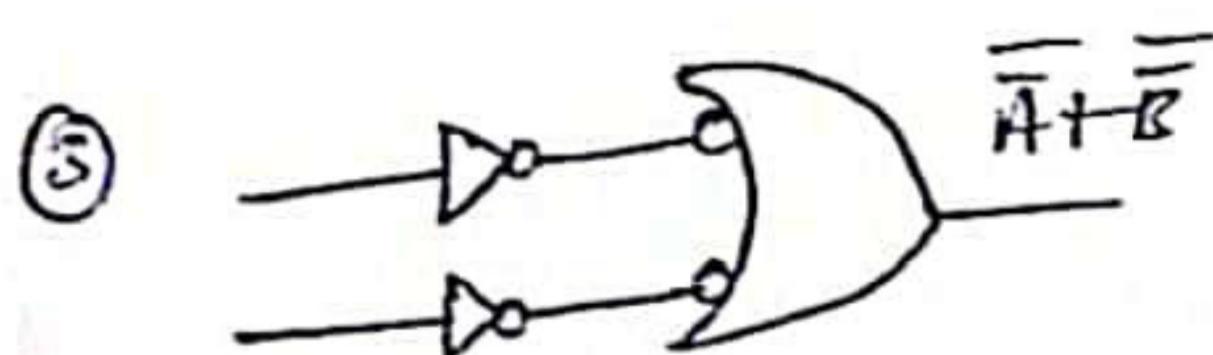
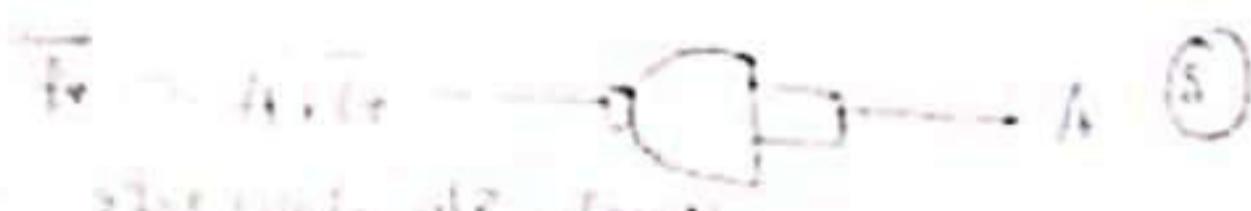
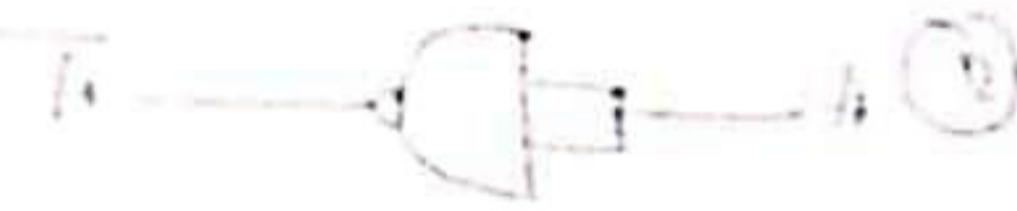
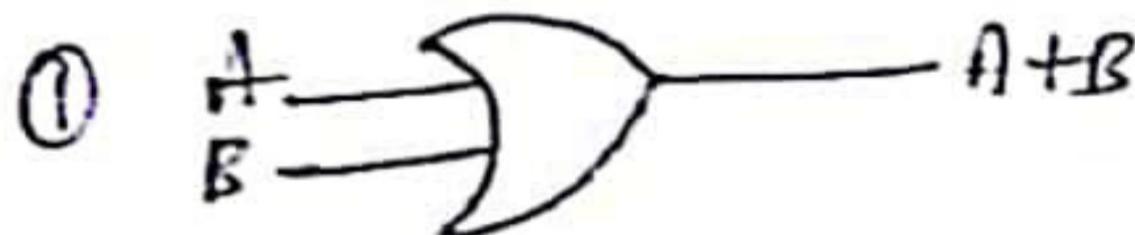
② Realization of AND gate Using NOR Gate :-

129



$$A \cdot B = \overline{\overline{A} + \overline{B}} = \overline{\overline{A}} \cdot \overline{\overline{B}} = A \cdot B$$

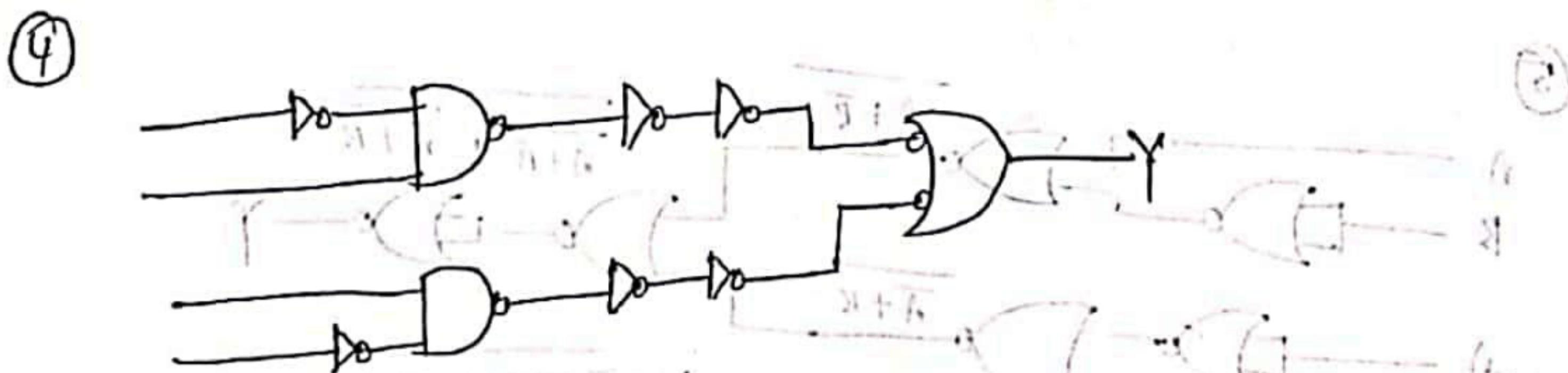
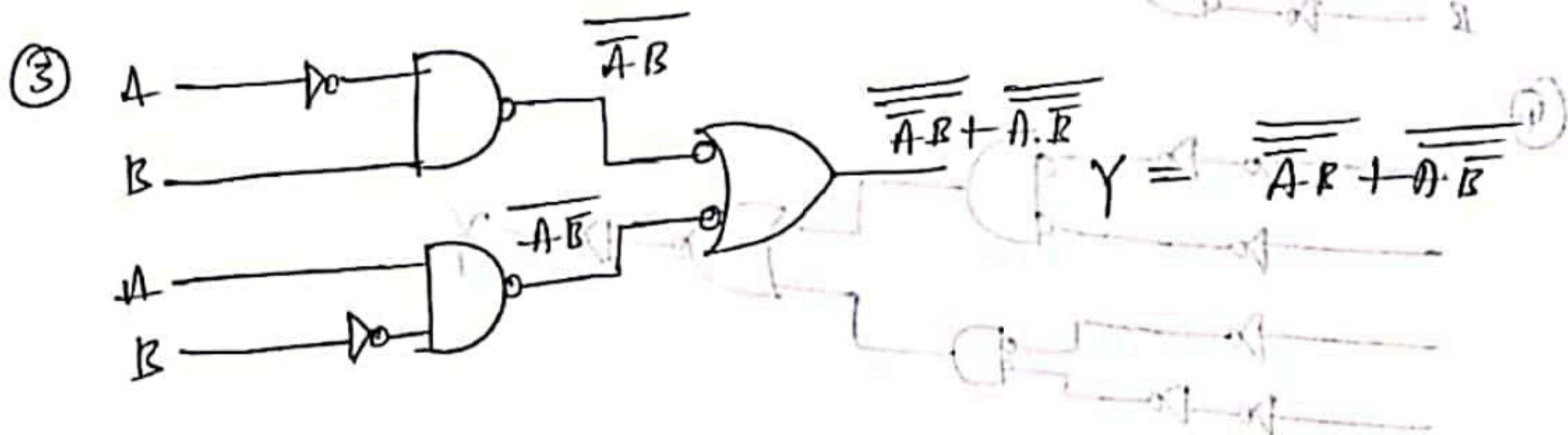
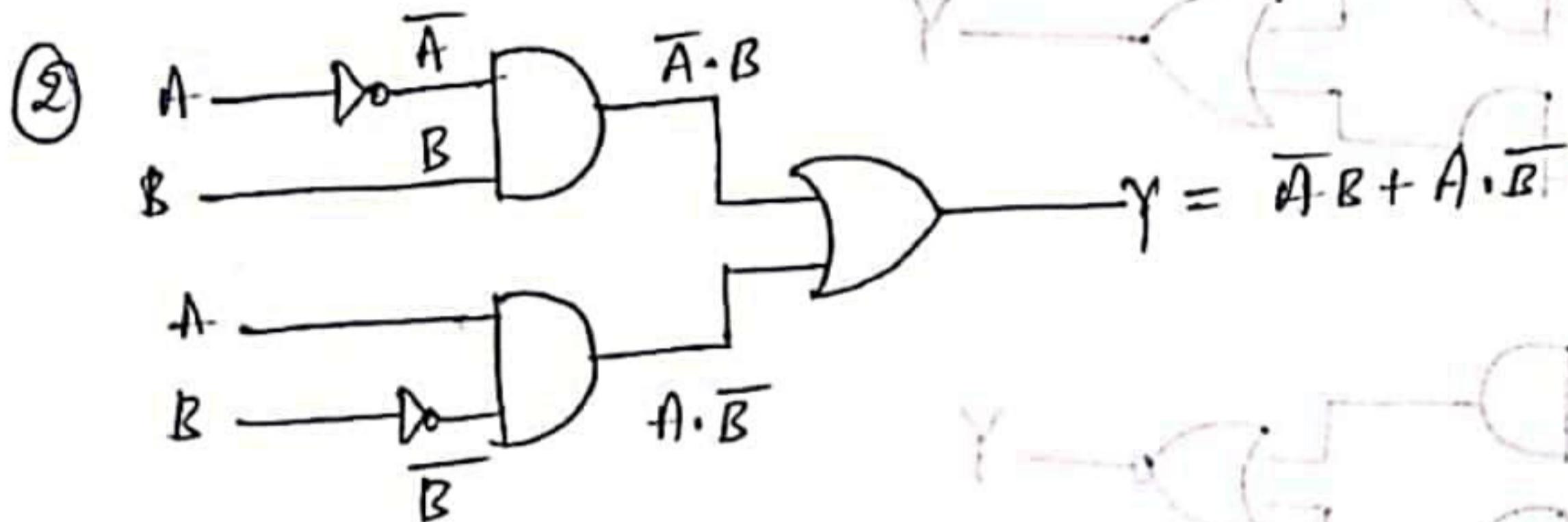
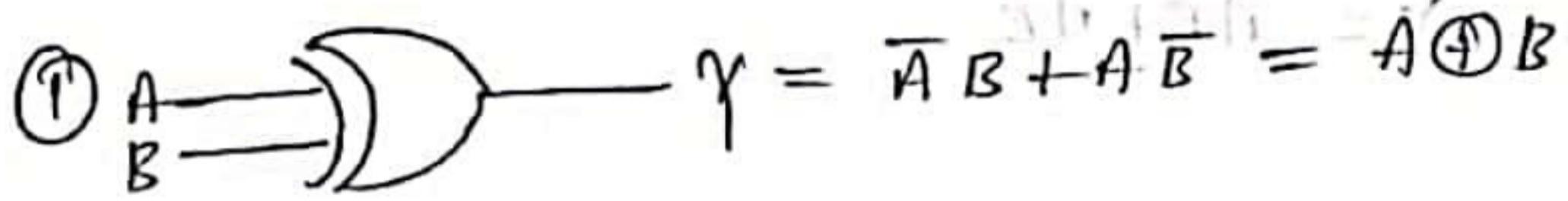
③ Realization of OR gate Using NAND & NOR gates :-



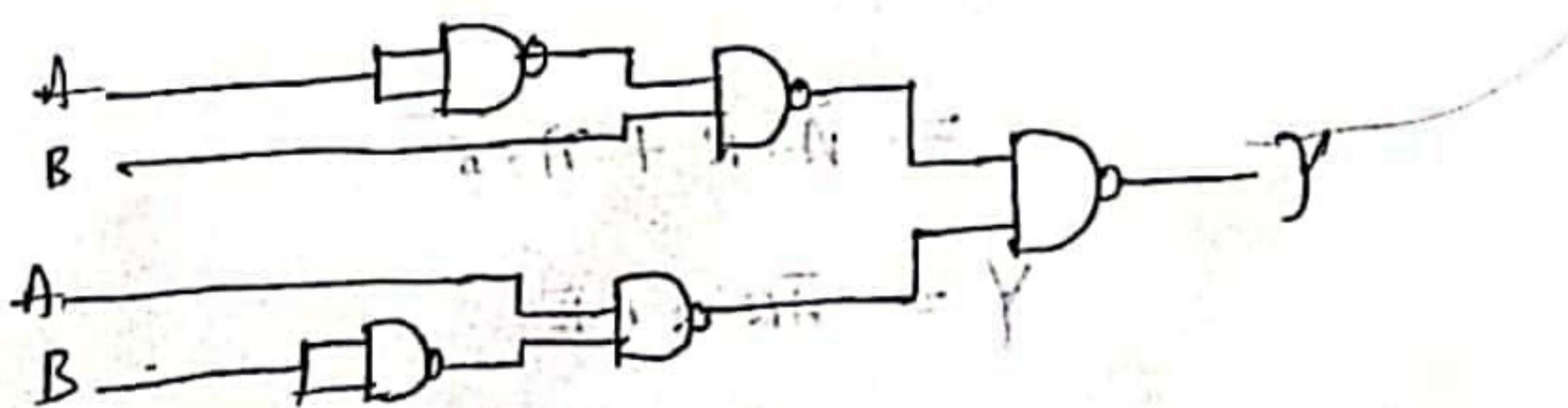
(27)

Realization of EX-OR gate Using NAND & NOR Using NAND gate :-

(130)

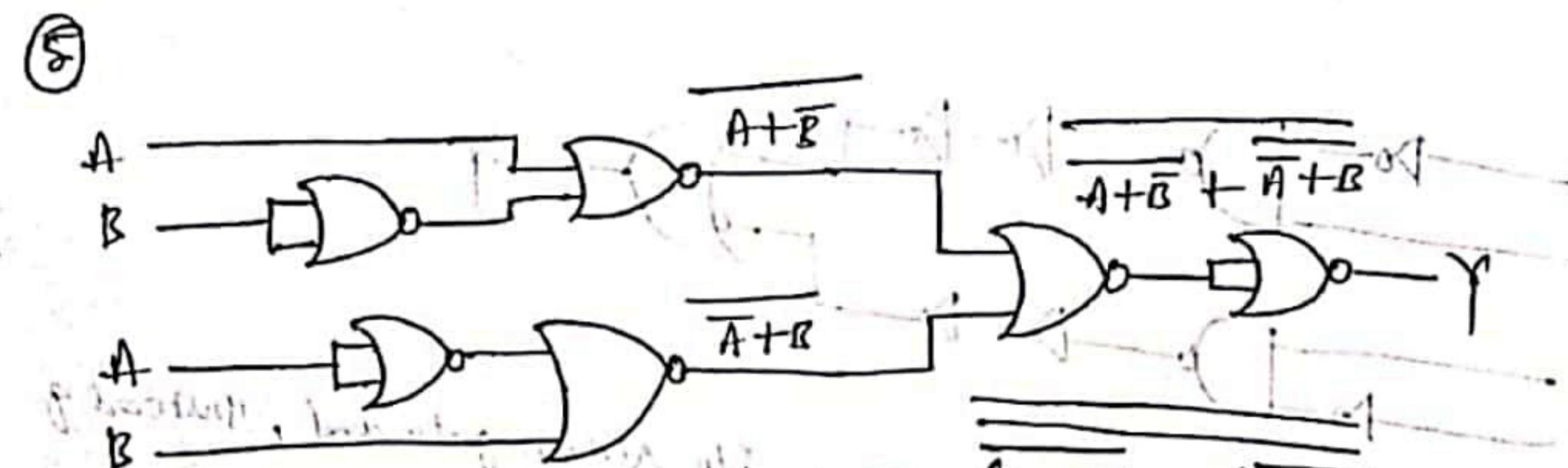
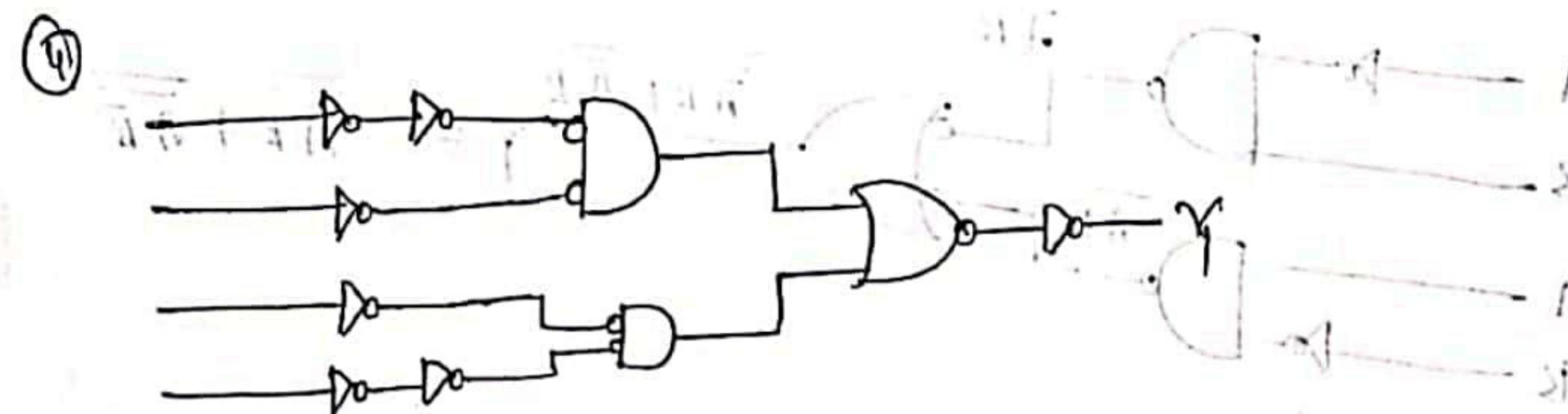
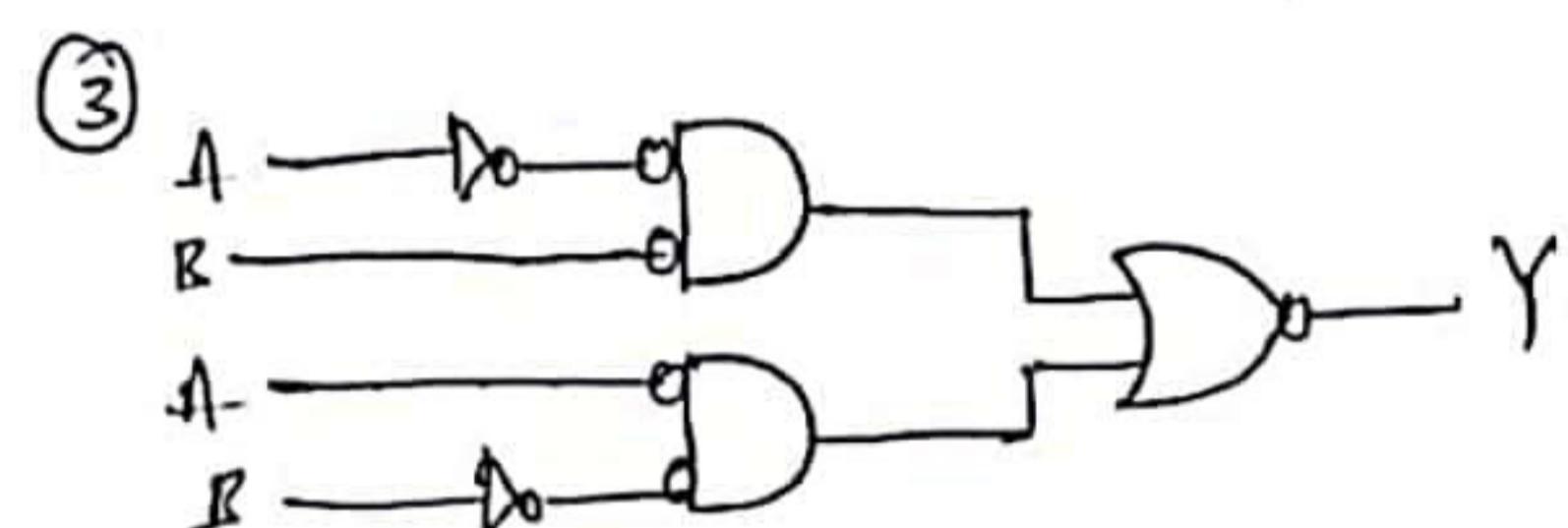
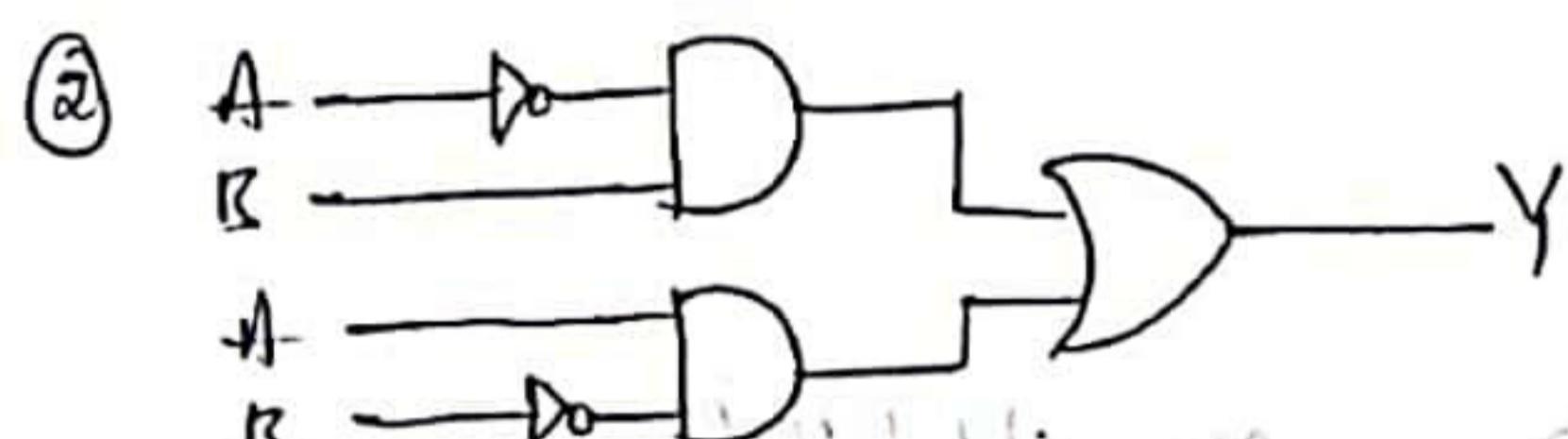
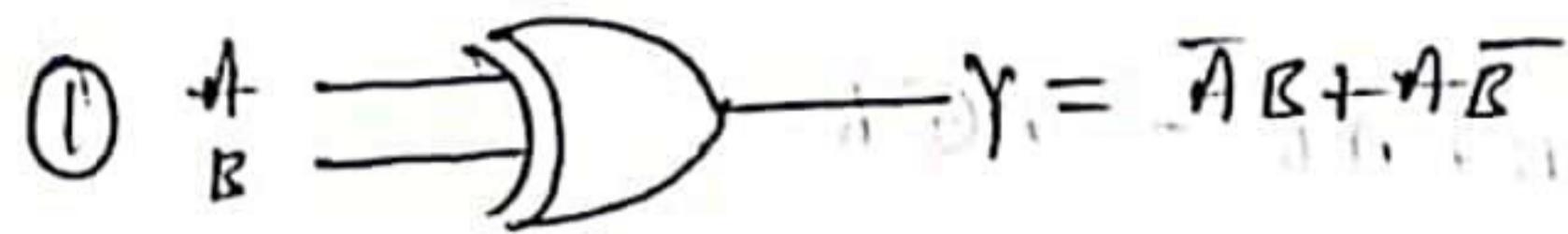


⑤ Instead of not gate place single Bi/p NAND gate and, Instead of
bulleted OR gate place NAND gate



131

Using NOR Gate:

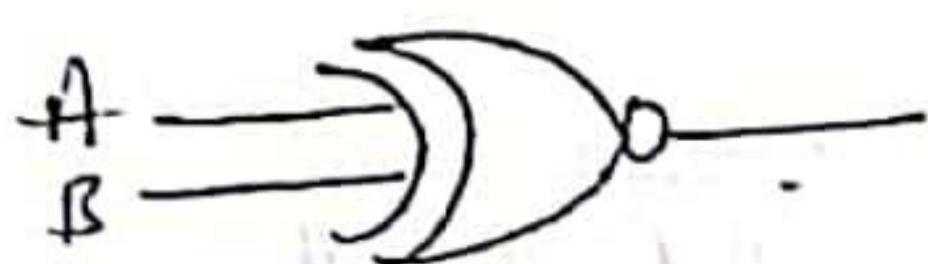


$$= (\overline{A} + \overline{B}) + (\overline{A} + B)$$

$$= A \circ B + A \cdot \overline{B}$$

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

→ Realization of X-NOR gate Using NAND & NOR 28

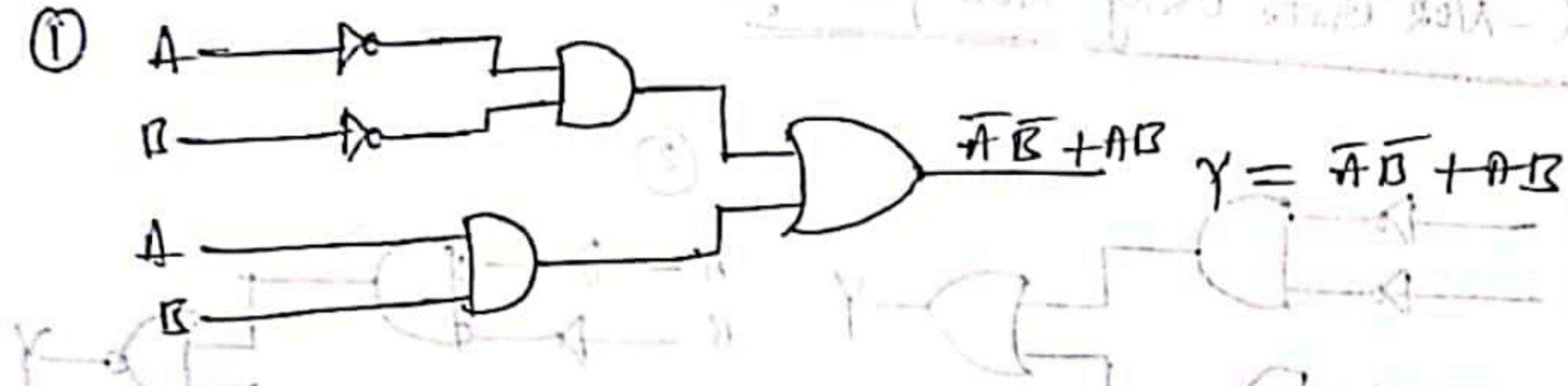


$$\overline{A} \cdot \overline{B} + AB = A \oplus B$$

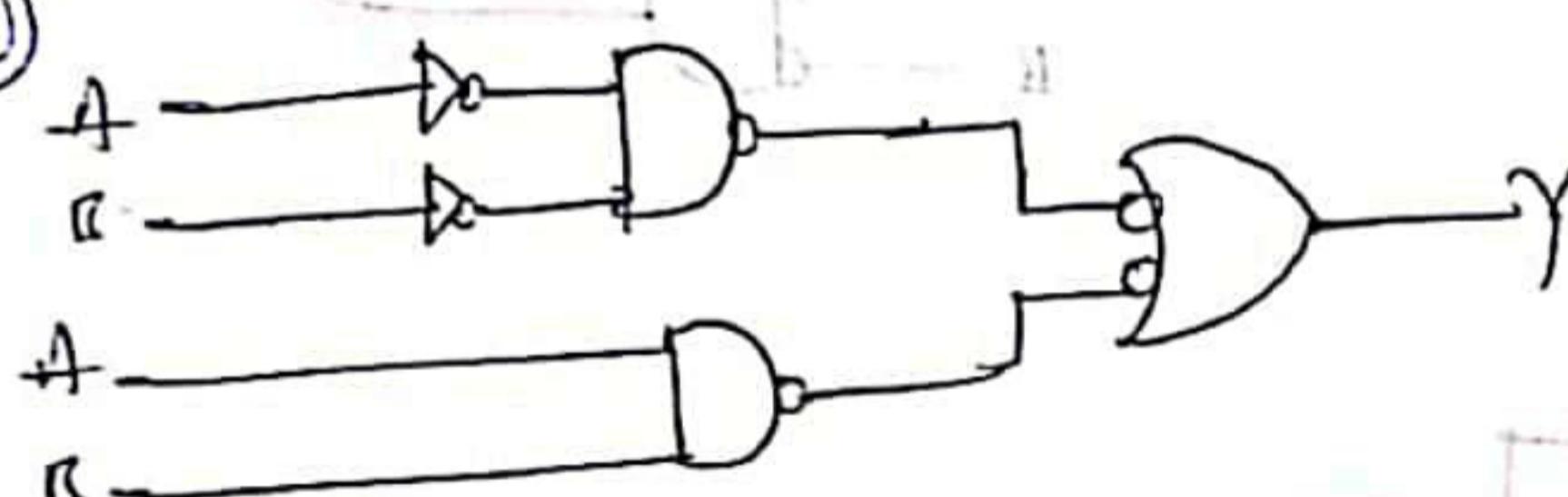
132

→ Using NAND Implementation

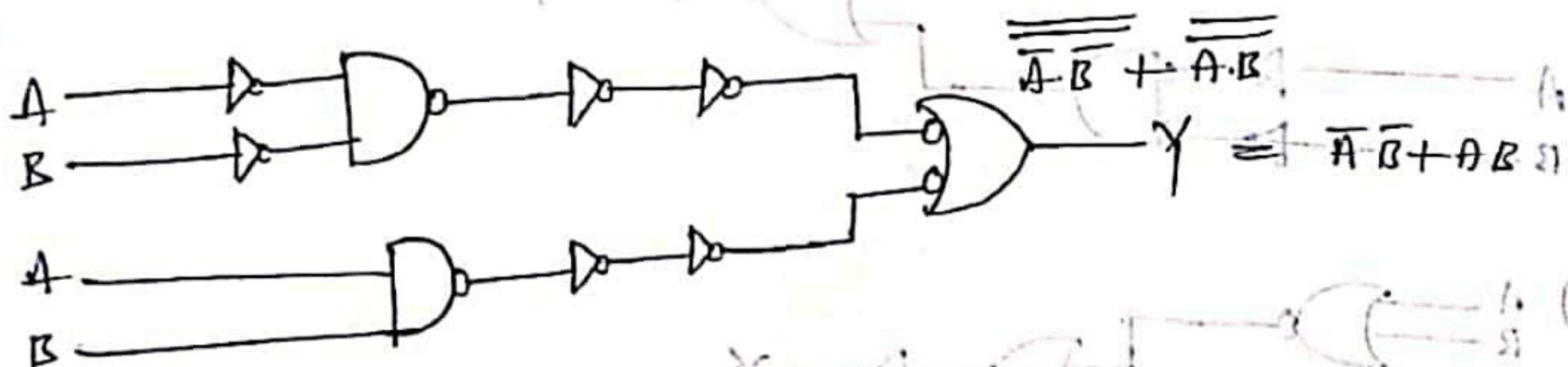
①



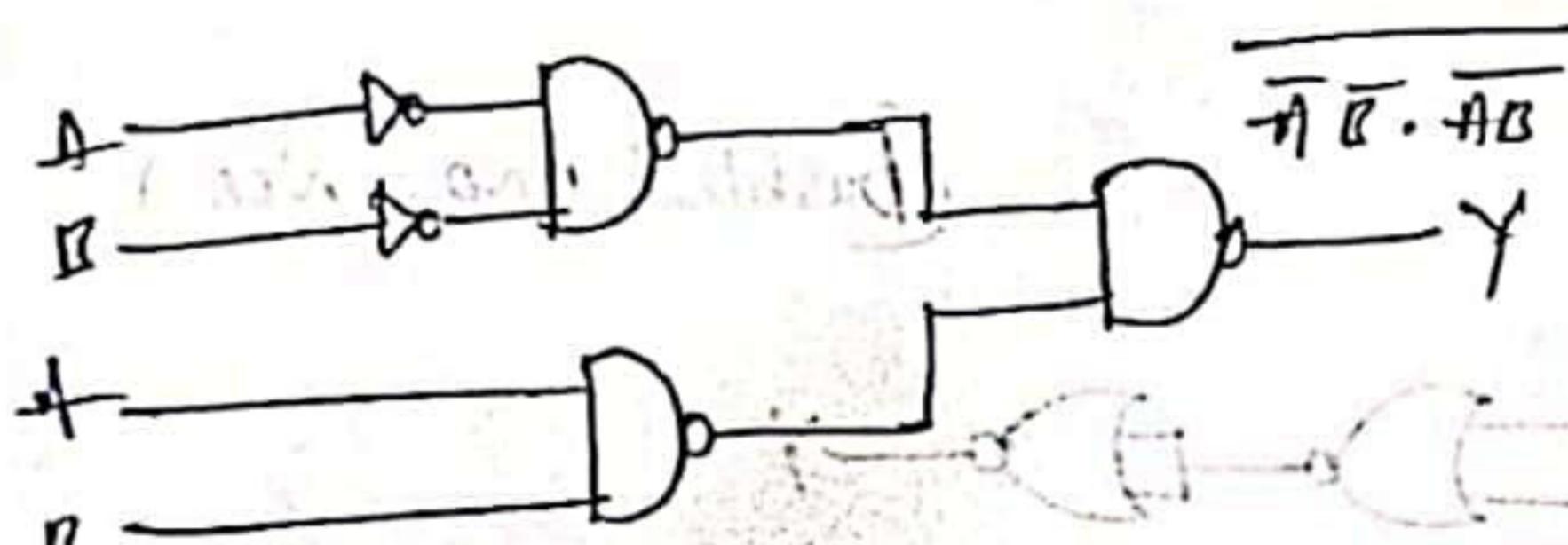
②



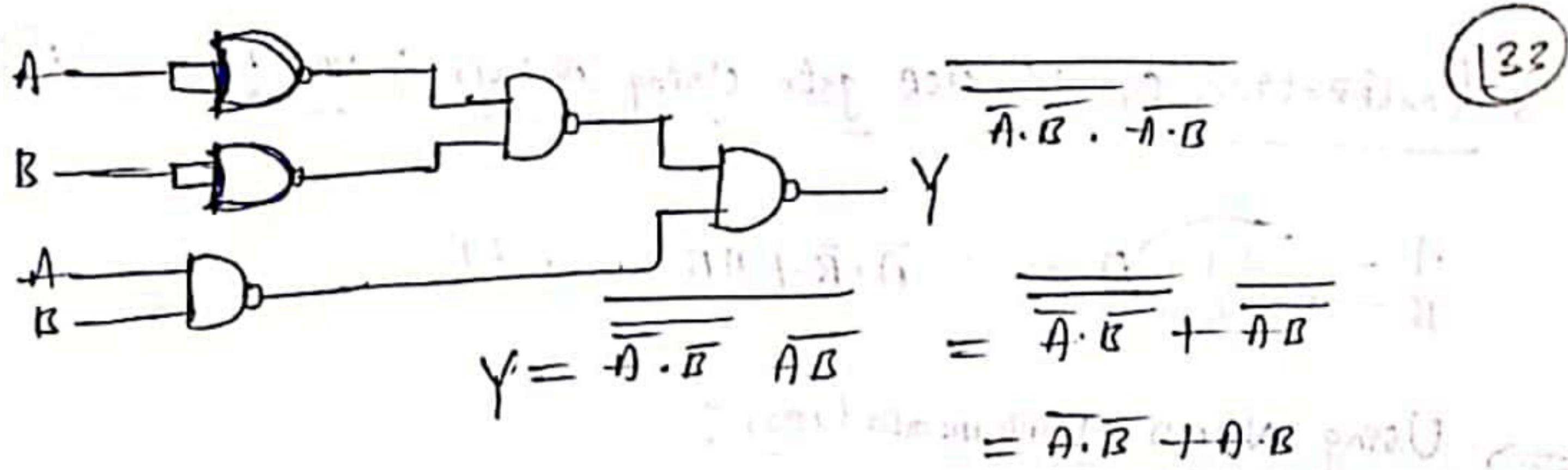
③



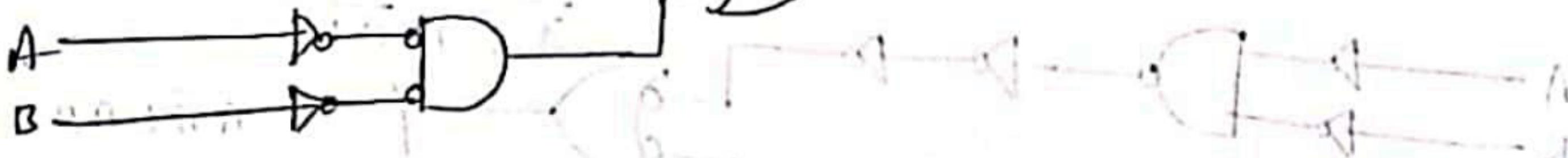
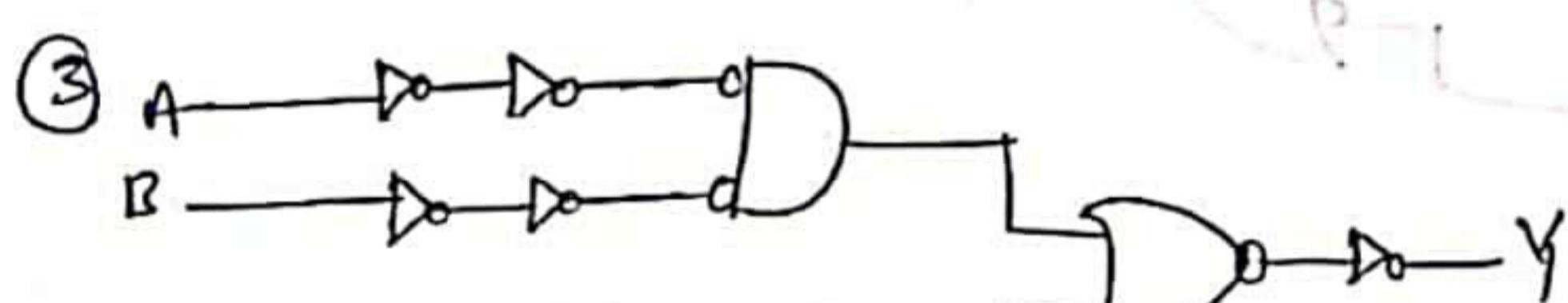
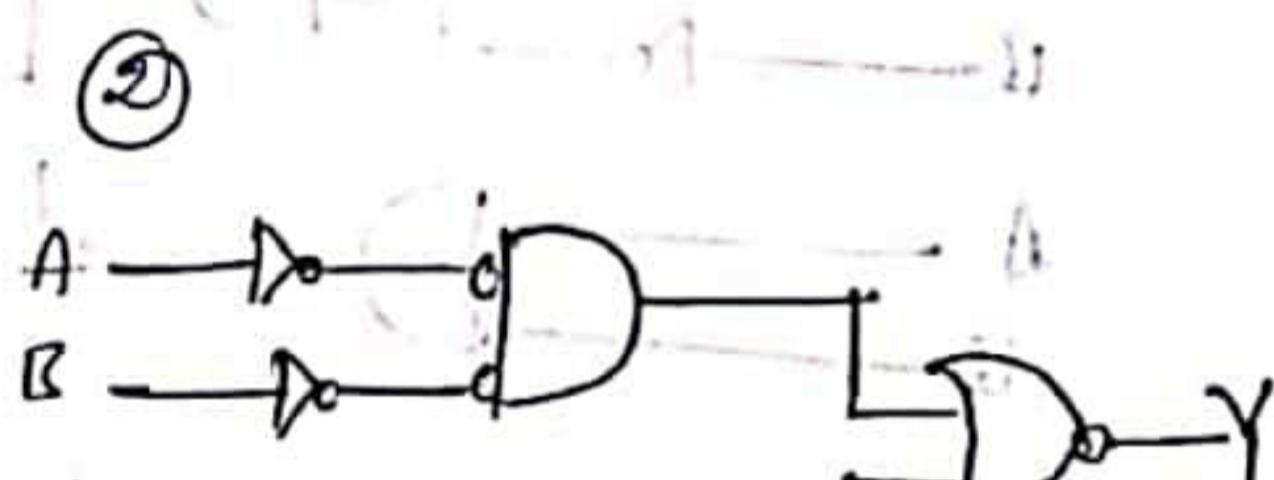
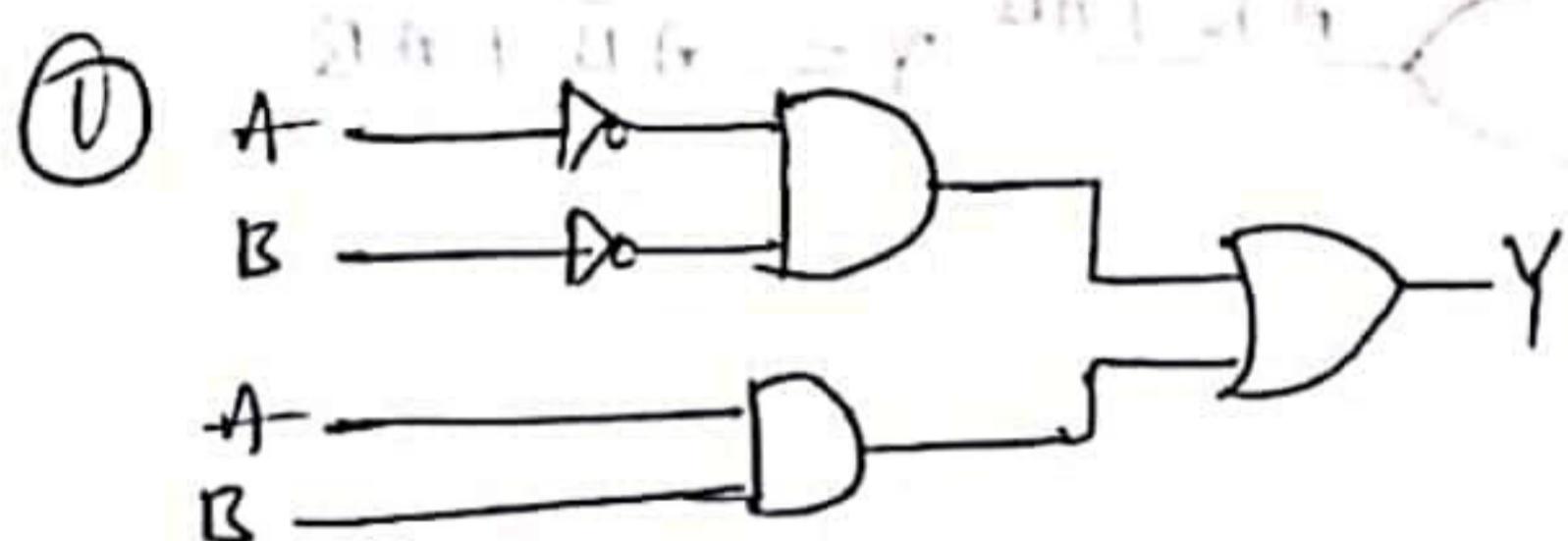
④



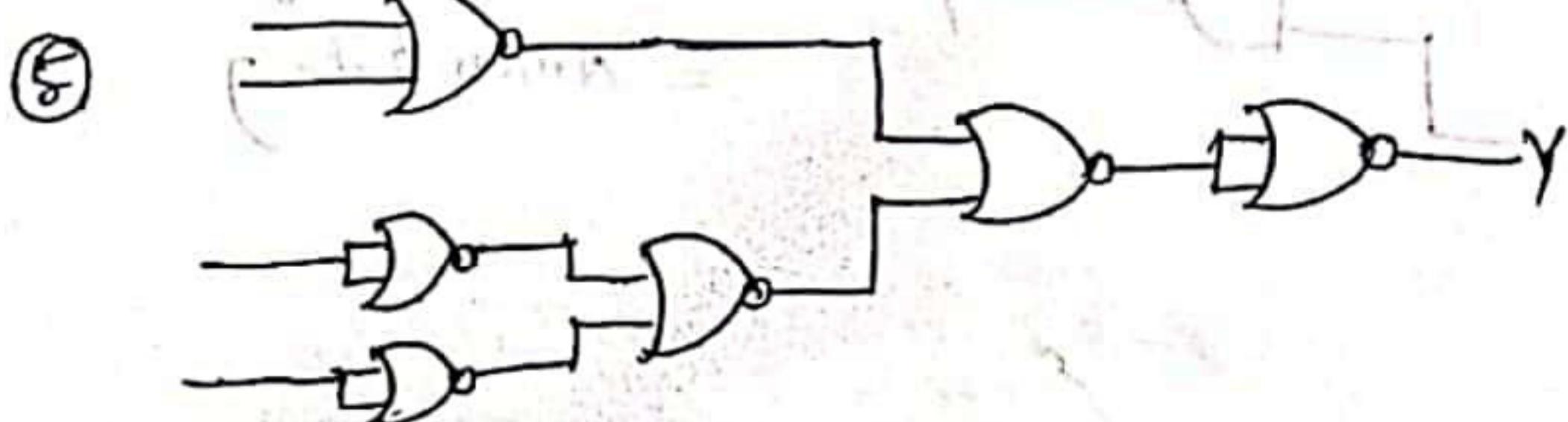
(Bubbled OR gate
= NAND gate)



\Rightarrow X-NOR Gate using NOR gate :-



(Bubbled AND = NOR)



(29)

⇒ Implement the following functions using NAND Gates.

(134)

$$\textcircled{a} \quad F_1 = A(B+CD) + \overline{BC}$$

$$\textcircled{b} \quad F_2 = w\bar{x} + \bar{y}z(\bar{x}+\bar{w})$$

Sol

$$\textcircled{a} \quad F_1 = A(B+CD) + \overline{BC}$$

$$= A\cdot B + A\cdot C\cdot D + \overline{B}\cdot \overline{C}$$

$$= A\cdot B + A\cdot C\cdot D + \overline{B} + \overline{C}$$

$$= \underline{A\cdot B + \overline{B}} + \overline{C} + A\cdot C\cdot D$$

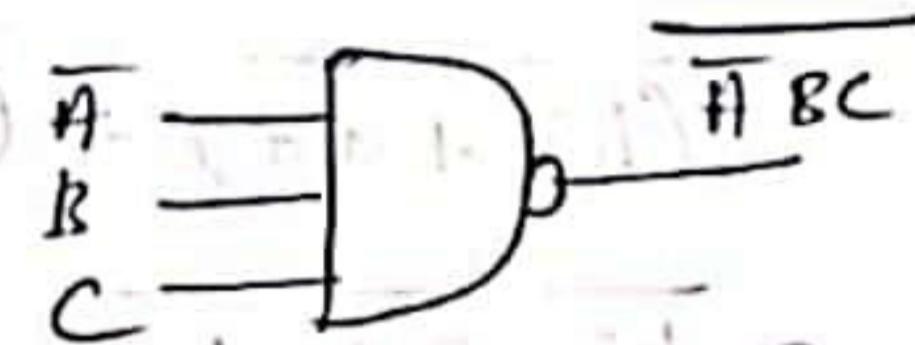
$$= \overline{B} + A + \overline{C} + A\cdot C\cdot D$$

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

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

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

$$= \overline{\overline{A} \cdot \overline{B} \cdot \overline{C}}$$



$$\therefore \overline{B} + A\overline{B} = \overline{B} + A$$

$$\overline{C} + A\cdot C\cdot D = \overline{C} + A\cdot D$$

Redundant Law) R.L.R

$$\therefore HD = 1$$

\textcircled{b}

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

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

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

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

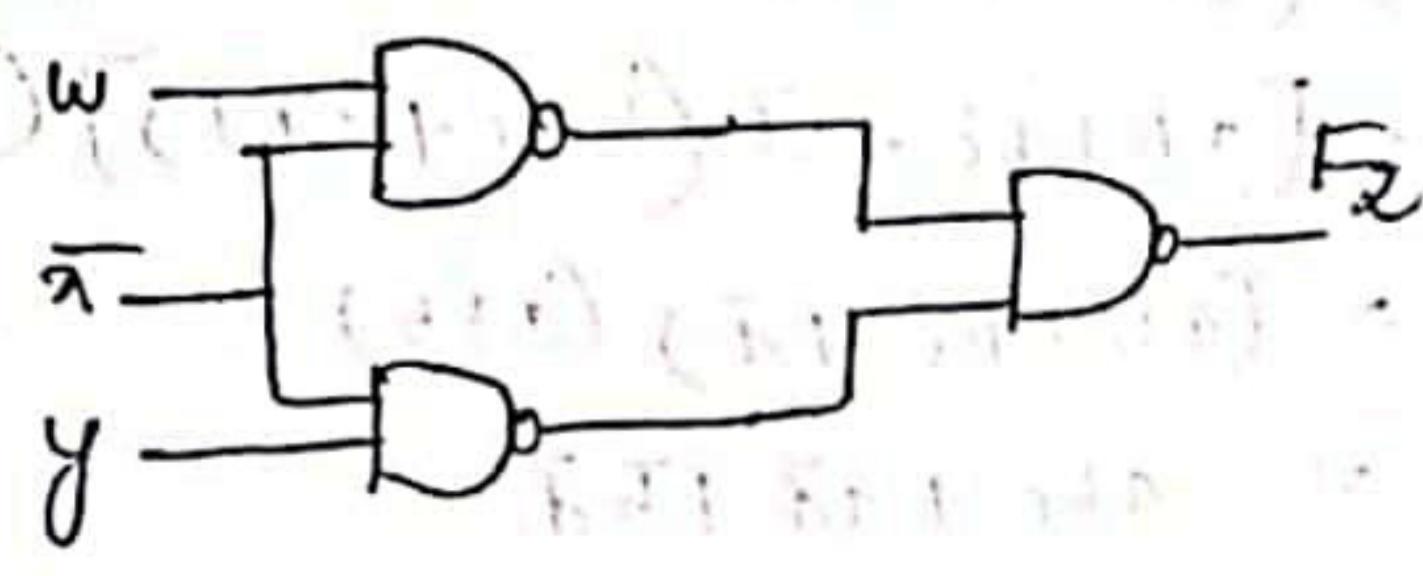
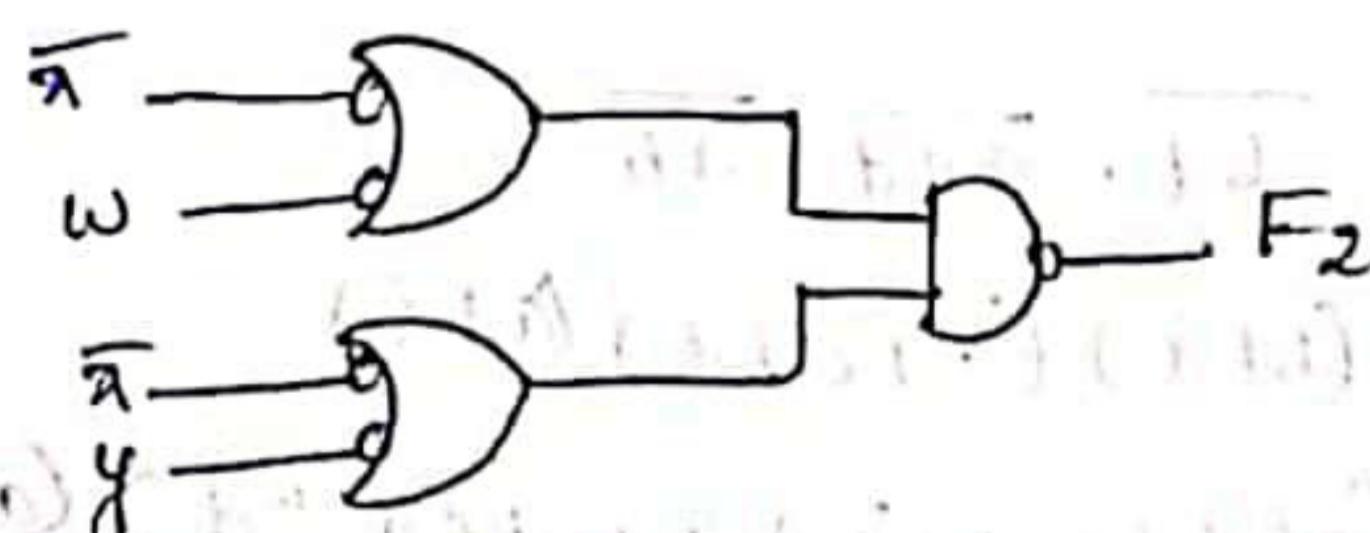
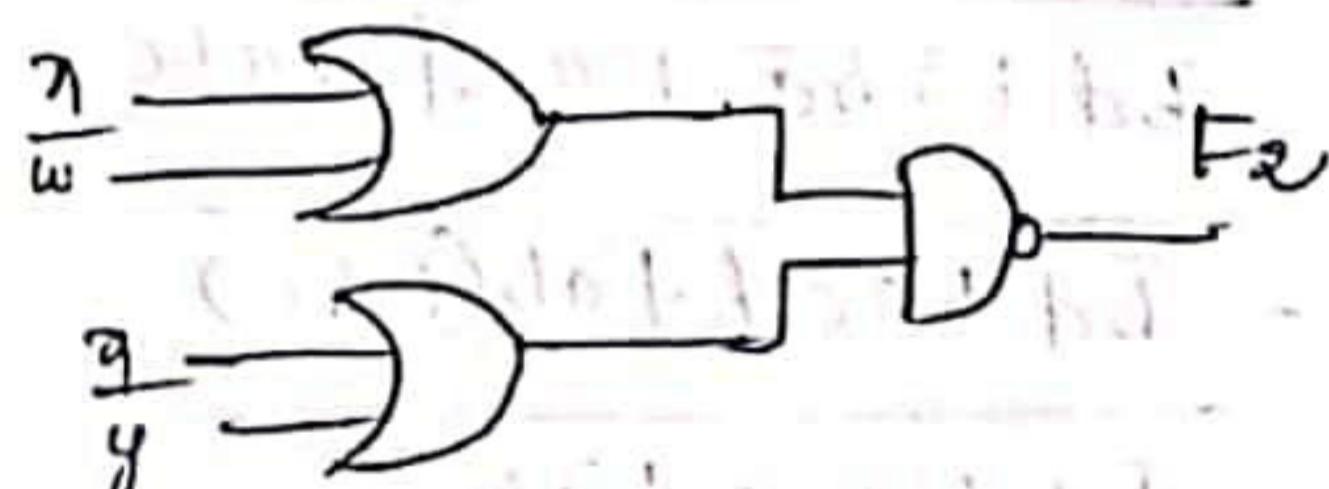
$$= \bar{x}w + \bar{y}z + \bar{y}z\bar{x}$$

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

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

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

$$= \frac{\bar{x}(w+y)}{\bar{x}(w+y)}$$



⇒ Find the complement of the following boolean function & Reduce them to minimum no. of literals.

(135)

$$\textcircled{a} (b\bar{c} + \bar{a}\bar{d})(a\bar{b} + c\bar{d})$$

$$\textcircled{b} (\bar{b}d + \bar{a}\bar{b}\bar{c} + a\bar{c}\bar{d} + \bar{a}\bar{b}c)$$

$$\underline{\text{Sof}} \quad \textcircled{a} \quad \overline{(b\bar{c} + \bar{a}\bar{d})(a\bar{b} + c\bar{d})}$$

$$= \overline{(b\bar{c} + \bar{a}\bar{d})} + \overline{(a\bar{b} + c\bar{d})}$$

$$= \overline{b\bar{c}} \cdot \overline{\bar{a}\bar{d}} + \overline{a\bar{b}} \cdot \overline{c\bar{d}}$$

$$= (\bar{b} + c)(a + \bar{d}) + (\bar{a} + b)(\bar{c} + d)$$

$$= a\bar{b} + \bar{b}\bar{d} + ac + c\bar{d} + \bar{a}\bar{c} + \bar{a}\bar{d} + b\bar{c} + bd$$

$$= a\bar{b} + ac + \bar{b}\bar{d} + c\bar{d} + \bar{a}\bar{c} + \bar{a}\bar{d} + b\bar{c} + bd$$

$$= 1$$

$$\textcircled{b} \quad \overline{\bar{b}d + \bar{a}\bar{b}\bar{c} + a\bar{c}\bar{d} + \bar{a}\bar{b}c}$$

$$= \overline{\bar{b}d + a\bar{c}\bar{d} + \bar{a}\bar{b}(\bar{c} + c)}$$

$$= \overline{\bar{b}d + a\bar{c}\bar{d} + \bar{a}\bar{b}}$$

$$= \overline{\bar{b}d} \cdot \overline{a\bar{c}\bar{d}} \cdot \overline{\bar{a}\bar{b}}$$

$$= (b + \bar{d})(\bar{a} + \bar{c} + \bar{d})(a + \bar{b})$$

$$= (\bar{a}b + b\bar{c} + b\bar{d} + \bar{a}\bar{d} + \bar{c}\bar{d} + \bar{d})(a + \bar{b})$$

$$= [\bar{a}b + b\bar{c} + \bar{d}(b + \bar{a} + \bar{c} + 1)](a + \bar{b})$$

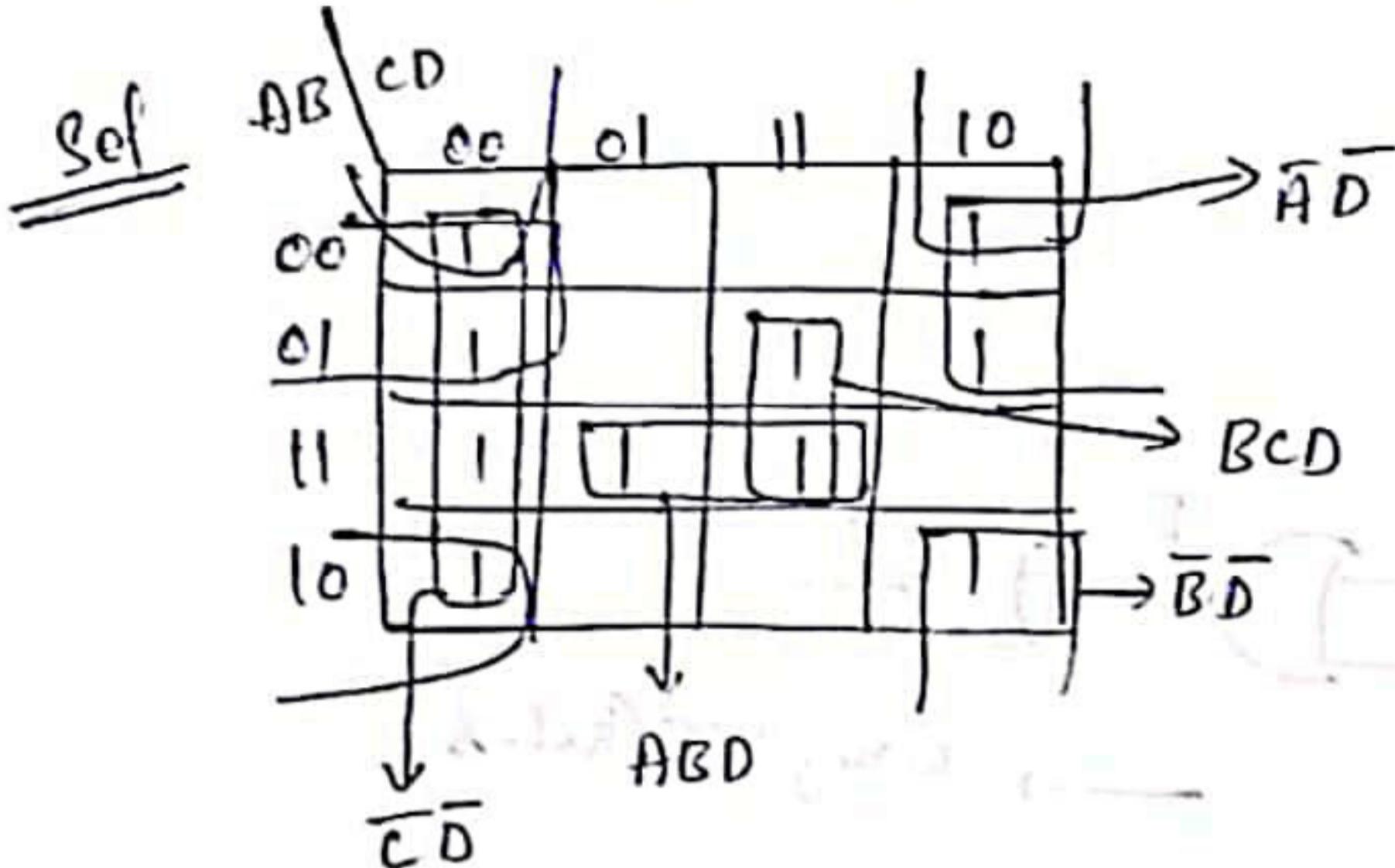
$$= (\bar{a}b + b\bar{c} + \bar{d})(a + \bar{b})$$

$$= ab\bar{c} + a\bar{d} + \bar{b}d$$

Reduce using mapping the following expression and implement the real minimal expression in universal logic

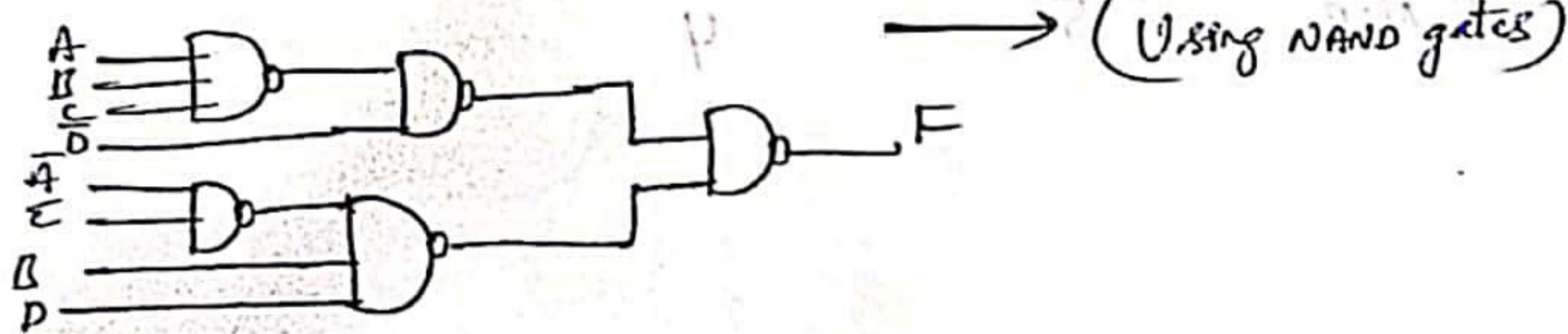
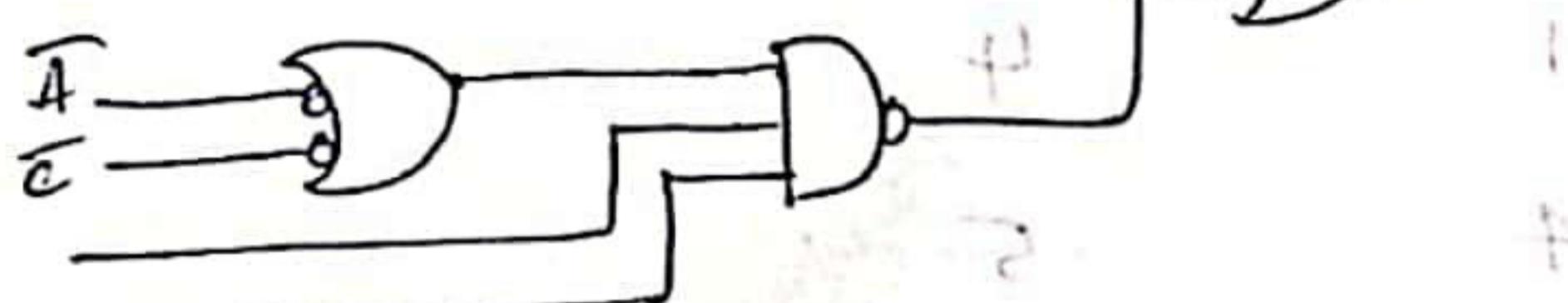
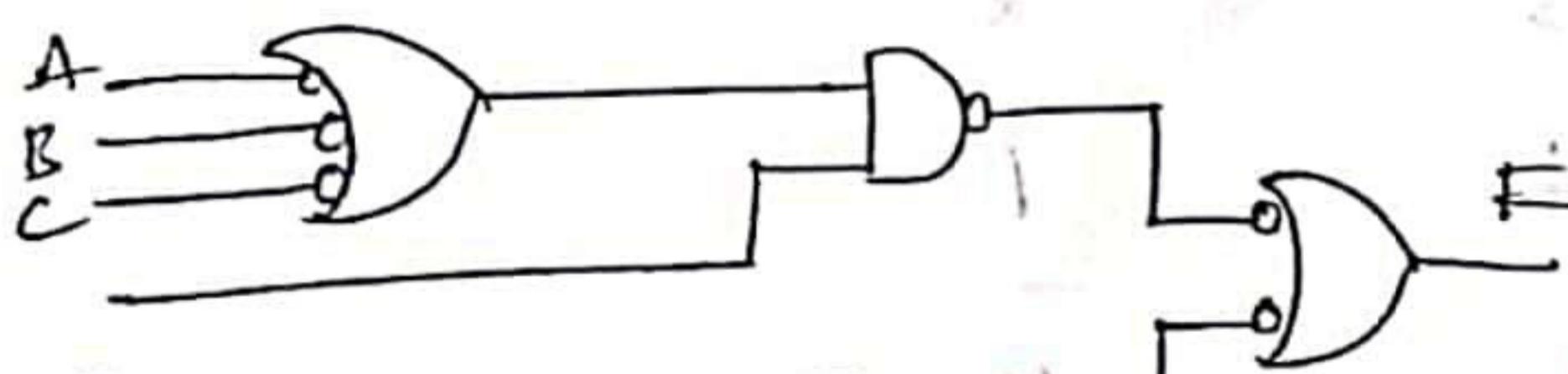
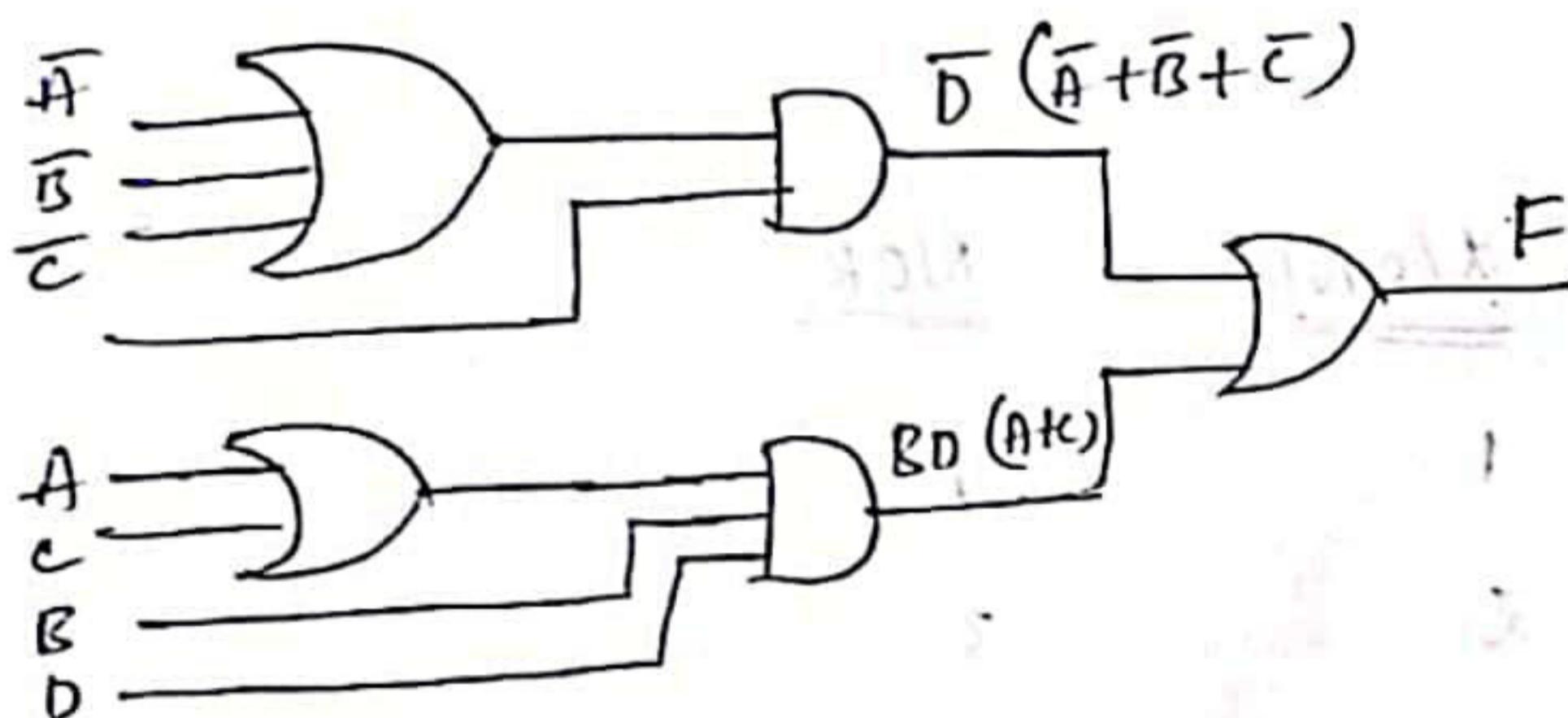
136

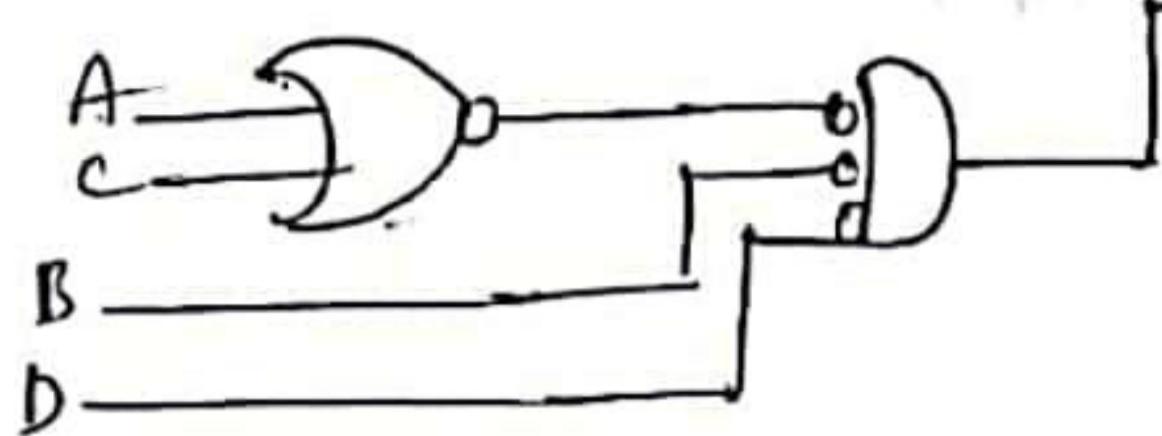
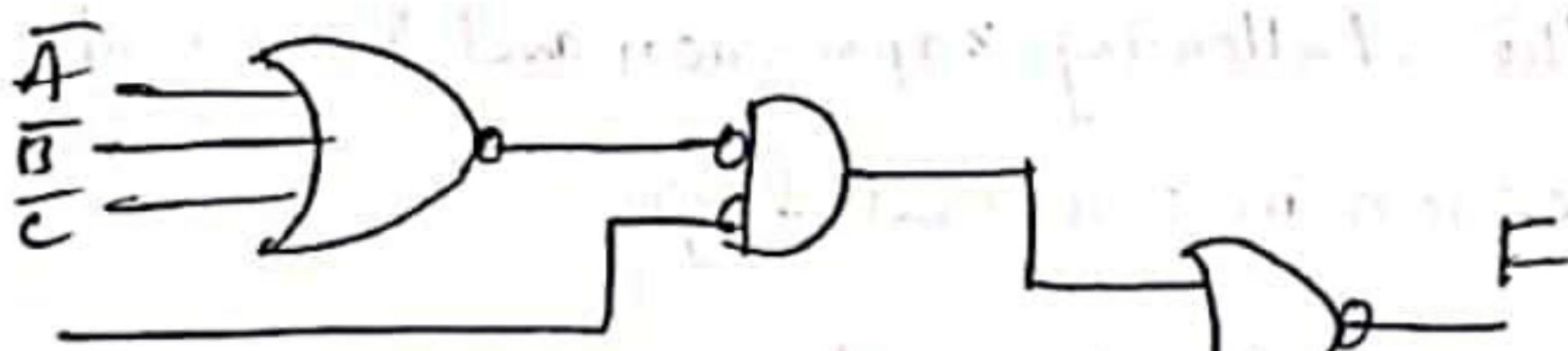
$$F = \Sigma m(0, 2, 4, 6, 7, 8, 10, 12, 13, 15)$$



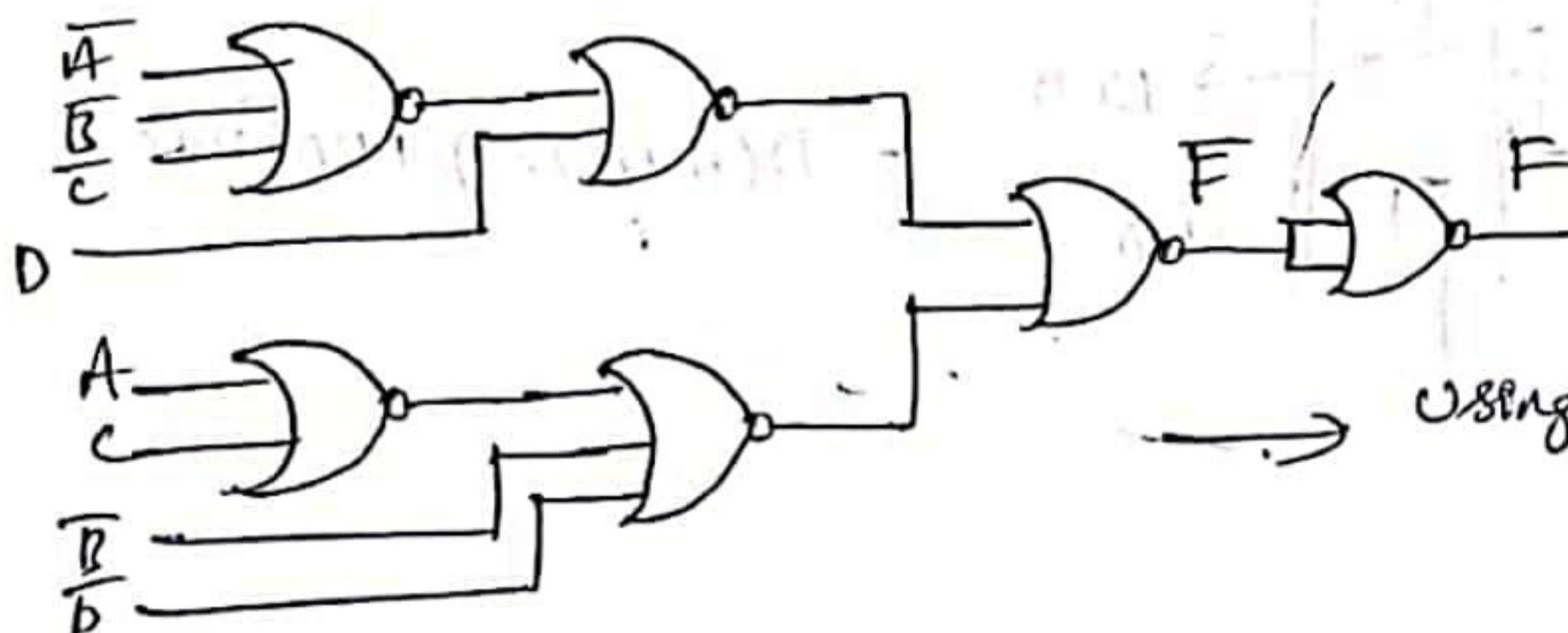
$$F = C̄D̄ + ĀD̄ + B̄D̄ + ĀB̄D̄ + B̄C̄D̄$$

$$= D̄(Ā + B̄ + C̄) + B̄D̄(Ā + C̄)$$





ANOTHER IMPLEMENTATION



using NOR Gates

NOT

1

AND

2

OR

3

NOR

4

NAND

1

X-OR

4

X-NOR

5

NOR

1

3

2

4

5

4

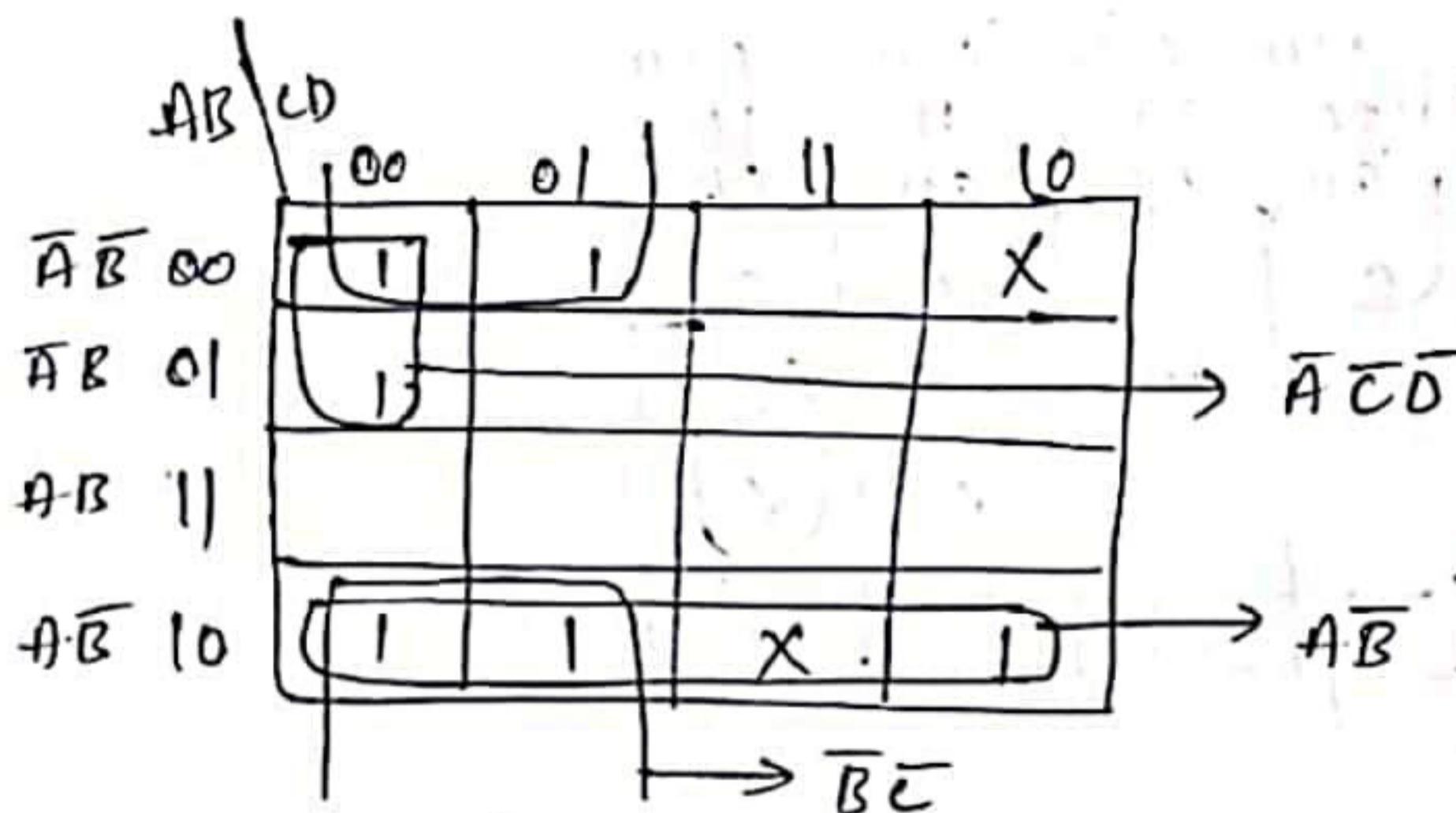


Q. Reduce the following expression Using K-map and implement it using NAND gate.

(138)

$$\text{TM}(3, 5, 6, 7, 12, 13, 14, 15) + \text{d}(2, 11)$$

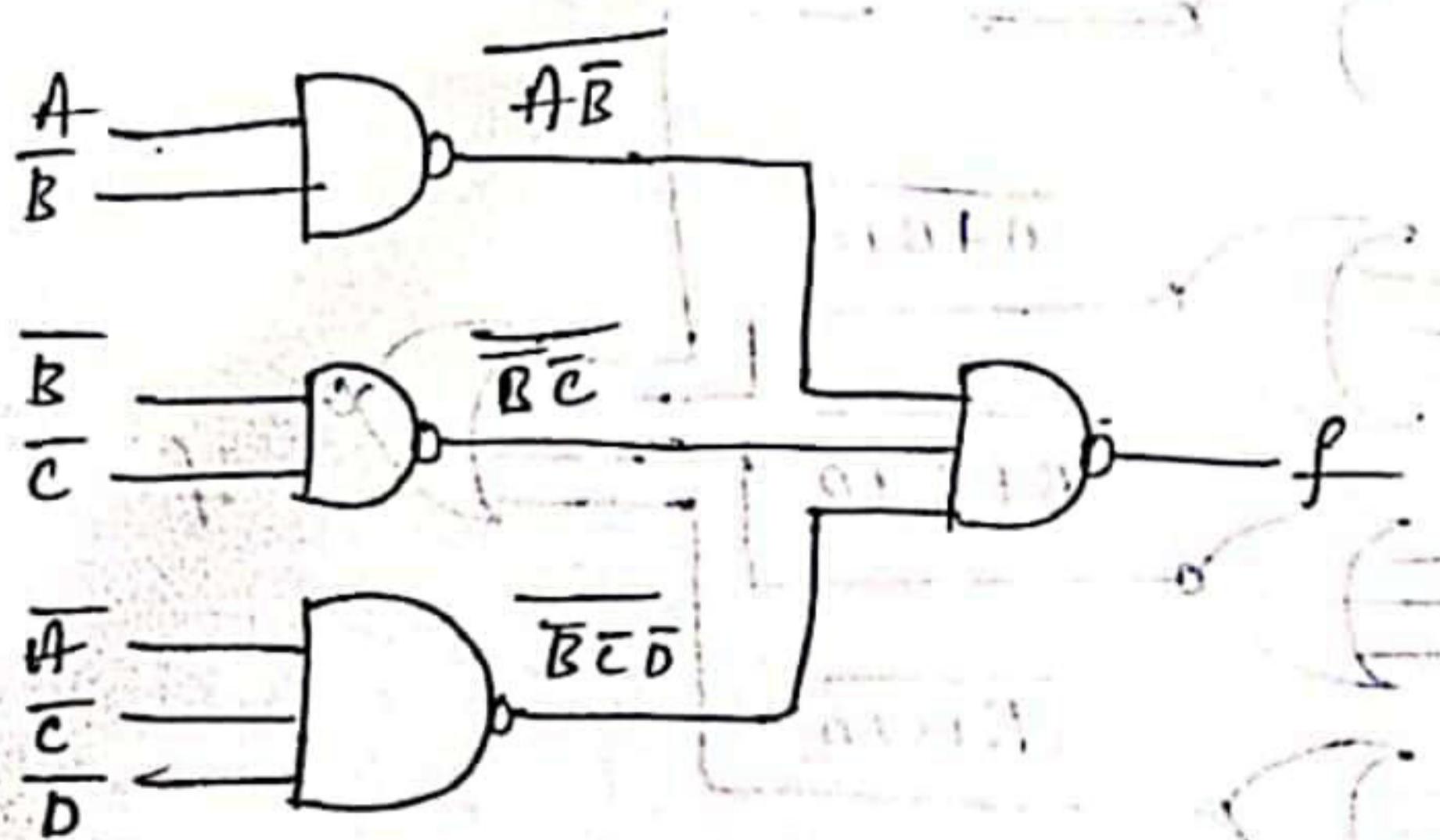
$$\Rightarrow \text{Em}(0, 1, 4, 8, 9, 10) + \text{d}(2, 11)$$



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

$$\text{W.K.T } f = \overline{\bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}\bar{C}\bar{D}}$$

$$f = \overline{\bar{A}\bar{B} \cdot \bar{B}\bar{C} \cdot \bar{A}\bar{C}\bar{D}}$$



Q Minimize the expression Using K-map and implement it
Using NOR gate

139

$$f(A, B, C, D) = \Sigma_m(1, 4, 7, 10, 11, 12, 13) + \Sigma_d(5, 14, 15)$$

$$= \Pi_M(\bar{C}, 2, 3, 6, 8, 9) + \Sigma_d(5, 14, 15)$$

AB	CD	$C+D$	$C+\bar{D}$	$\bar{C}+D$	$\bar{C}+\bar{D}$
$A+B$	00	0	0	0	0
$A+\bar{B}$	01	X			
$\bar{A}+\bar{B}$	11		X		X
$\bar{A}+B$	10	0	0		

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

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

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

