Saroj Dahal - 191725

Date / / Page No.

Assignment - 2.

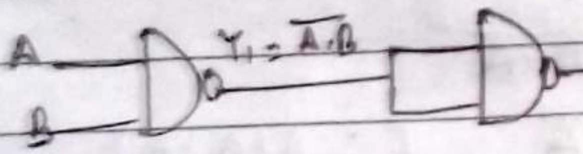
1. Discuss the universal property of NAND and NOR gate with appropriate example.

Ans NAND and NOR gates are coined as universal gates because from these any gates, any gates can be made.

Examples:

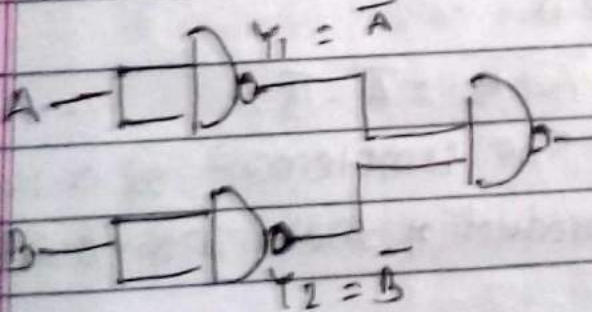
In case of NAND gates:

AND gate using NAND gate.



$Y = \overline{\overline{A.B}} = A.B.$

OR gate using NAND gate.

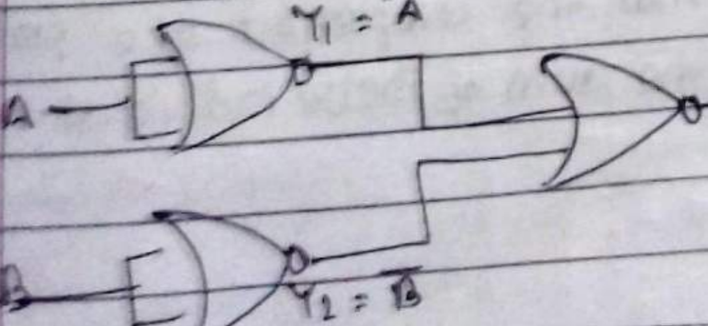


$Y = \overline{\overline{A}.\overline{B}}$

$= \overline{\overline{A}} + \overline{\overline{B}}$ (De Morgan's theorem)

$\therefore Y = A + B.$

In case of NOR gates:



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

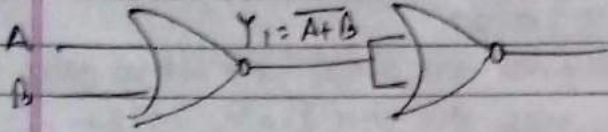
$= \overline{\overline{A}} . \overline{\overline{B}}$ (Demorgan's theorem)

$\therefore Y = A.B.$

AND gate using NOR gate

Saroj Dahal - 191725

OR gate using NOR gate:



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

$= A+B.$

NOT Gate using NOR gate.

$Y = \overline{A+A}$

$\therefore Y = \overline{A} \quad (\because A+A = A)$

2. Define De-morgan's theorem.

Ans De.Morgan's first theorem: $\overline{A+B} = \overline{A}\cdot\overline{B}$.

This theorem states that the complement of a sum of variables is equal to the product of their individual complements.

De.Morgan's second theorem: $\overline{A\cdot B} = \overline{A}+\overline{B}$.

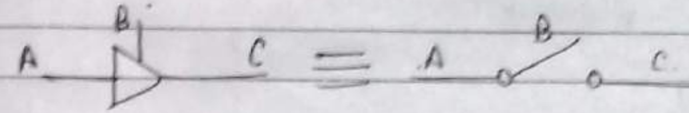
This theorem states that the complement of a product of variables is equal to the sum of their individual compliments.

3. Define tri state logic.

Ans

- In digital electronics three-state, tri-state or 3-state logic allows an output port to assume a high impedance state, effectively removing the output from the ckt, in addition to the 0 and 1 logic levels.

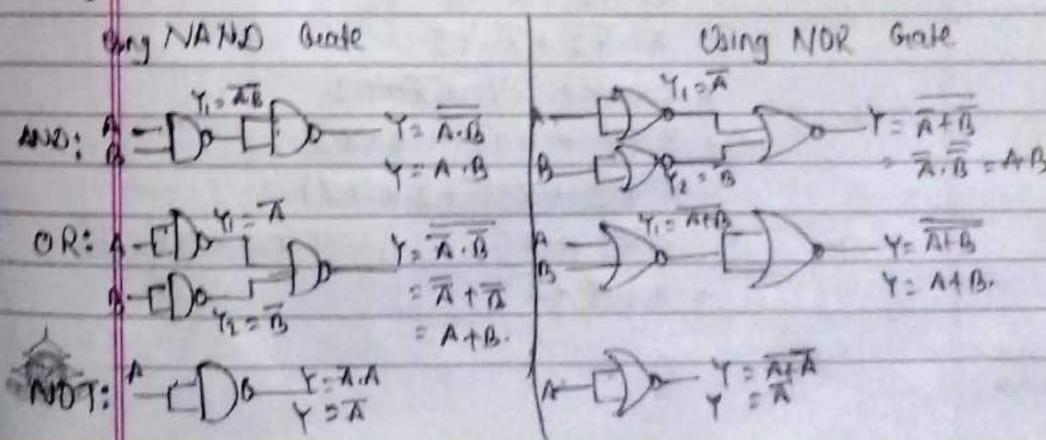
- Three-state outputs are implemented in many registers, bus drivers and flip-flops in the 7400 and 4000 series as well as in other types, but also internally in many integrated circuits.



A tristate buffer can be thought of as a switch. If B is on, the switch is closed. If B is off, the switch is open.

4. Realize the basic gate using universal gate.

Ans.



5. Find the complement of $F = X + YZ$ then show that $F \cdot \overline{F} = 0$ and $F + \overline{F} = 1$

Ans: Here,

$$F = X + YZ.$$

so, $$\overline{F} = \overline{X + YZ}$$

$$= \overline{X} \cdot \overline{YZ} \quad \text{(By De-Morgan's Theorem)}$$

Now,

$$F \cdot \overline{F} = (X + YZ) \cdot (\overline{X} \cdot \overline{YZ})$$

$$= X\overline{X} \cdot \overline{YZ} + YZ \cdot \overline{YZ}$$

$$= 0 + 0$$

$$\therefore F \cdot \overline{F} = 0$$

And,

$$F + \overline{F} = (X + YZ) + (\overline{X + YZ})$$

$$= (X + YZ) + (\overline{X} \cdot \overline{YZ})$$

$$= X\overline{X} + X \cdot \overline{YZ} + \overline{X} \cdot YZ + YZ \cdot \overline{YZ}$$

$$= 0 + X \cdot \overline{YZ} + \overline{X} \cdot YZ + (Y \cdot \overline{Y}) \cdot (Z \cdot \overline{Z})$$

$$= X \cdot \overline{YZ} + \overline{X} \cdot YZ + 0$$

$$= X \cdot \overline{YZ} + \overline{X} \cdot YZ$$

$$= \cancel{(X + \overline{X}) \cdot \overline{YZ} + \overline{X} \cdot YZ}$$

$$= \cancel{X\overline{YZ} + \overline{X}YZ + \overline{X}\,YZ}$$

$$= \cancel{X}\ \cancel{\overline{YZ}} + \cancel{\overline{X}}\ (\cancel{\overline{YZ} + YZ})$$

$$= \cancel{X\overline{YZ} + \overline{X}} \qquad \cancel{[\because \overline{YZ} + YZ = 1]}$$

$$= \cancel{X}\ (\cancel{\overline{Y} + \overline{Z}})$$

6. Simply the following Boolean function in pos form by means of 4 variable k map and don't care condition. Draw the logic diagram with i). OR-AND gates ii). NOR gate.

$$F(W, X, Y, Z) = \sum m(2, 3, 4, 5, 6, 7, 11, 14, 15).$$

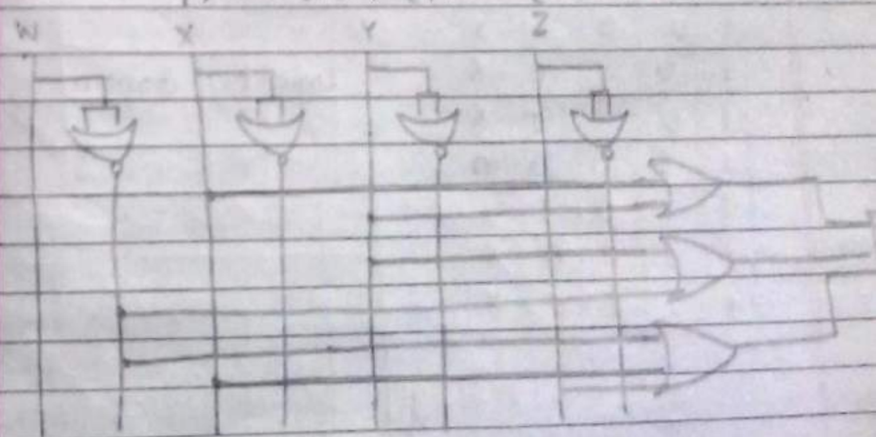
Ans:

WX \ YZ	00	01	11	10
00	0 (0)	0 (1)	1 (3)	1 (2)
01	1 (4)	1 (5)	1 (7)	1 (6)
11	0 (12)	0 (13)	1 (15)	1 (14)
10	0 (8)	0 (9)	1 (11)	0 (10)

Here, $F(W, X, Y, Z) = \sum m(2, 3, 4, 5, 6, 7, 11, 14, 15)$ which is in SOP form

so, $F(W, X, Y, Z) = \prod m(0, 1, 8, 9, 10, 12, 13)$ is in POS form.

From K-Map, $F = (Y + X) \cdot (Y + \overline{W}) \cdot (\overline{W} + X + Z)$



7. A logic ckt implement the following Boolean function, $F = \bar{A}C + A\bar{C}D$. It is found that the ckt input combination $A = C = 1$ can never occur. Using K-map with proper don't care condition find a simplified expression and implement it using NAND gates only.

Ans Here,

Given Boolean function, $F = \bar{A}C + A\bar{C}D$

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	X
1	0	1	1	X
1	1	0	0	0
1	1	0	1	1
1	1	1	0	X
1	1	1	1	X

K-Map

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	1	X	X
10	0	1	X	X

→ C

→ AD

$F = C + AD$

Logic ckt diagram

$\overline{\bar{C} \cdot \overline{AD}}$

$\bar{\bar{C}} + \overline{\overline{AD}}$

$C + AD$

8. Reduce the given expression in minimum number of literals using Boolean algebra and derive the truth table and implement in NAND logic. $A + B[AC + \{(AC + (B + \bar{C})D\}]$

Ans

$A + B[AC + \{(AC + (B+\bar{C})D\}]$

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

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

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

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

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

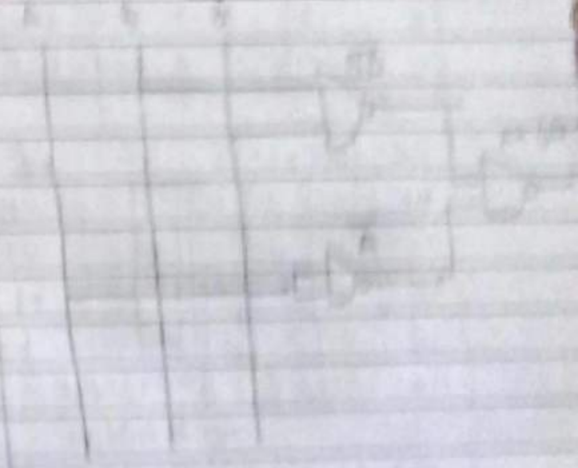
$= A + ABC + BD \quad [\because 1 + \bar{C} = 1]$

$= A + BD$

Truth table

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	[illegible]	1

Logic Diagram



9. Design a combinational ckt that converts a decimal digit from 2421 code to 8 4 -2 -1 code to binary.

Ans Truth table:

Decimal no:		2	4	2	1	8	4	-2	-1
		A	B	C	D	A_1	B_1	C_1	D_1
0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	1	0	1	1	1
2	2	0	0	1	0	0	1	1	0
3	3	0	0	1	1	0	1	0	1
4	4	0	1	0	0	0	1	0	0
5	11	1	0	1	1	1	0	1	1
6	12	1	1	0	0	1	0	1	0
7	13	1	1	0	1	1	0	0	1
8	14	1	1	1	0	1	0	0	0
9	15	1	1	1	1	1	1	1	1

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	x	x	x
11	1	1	1	1
10	x	x	1	x

From KMap: $A_1 = A$

AB \ CD	00	01	11	10
00	0	1	1	1
01	1	x	x	x
11	0	0	1	0
10	x	x	0	x

From KMap. $B_1 = \bar{A}B + \bar{A}D + \bar{A}C + BCD$

AB \ CD	00	01	11	10
00	0	1	0	1
01	0	x	x	x
11	1	0	1	0
10	x	x	1	x

From KMap. $C_1 = \bar{A}\bar{C}\bar{D} + \bar{A}C\bar{D} + A\overline{CD} + ACD$

AB \ CD	00	01	11	10
00	0	1	1	0
01	0	x	x	x
11	0	1	1	0
10	x	x	1	x

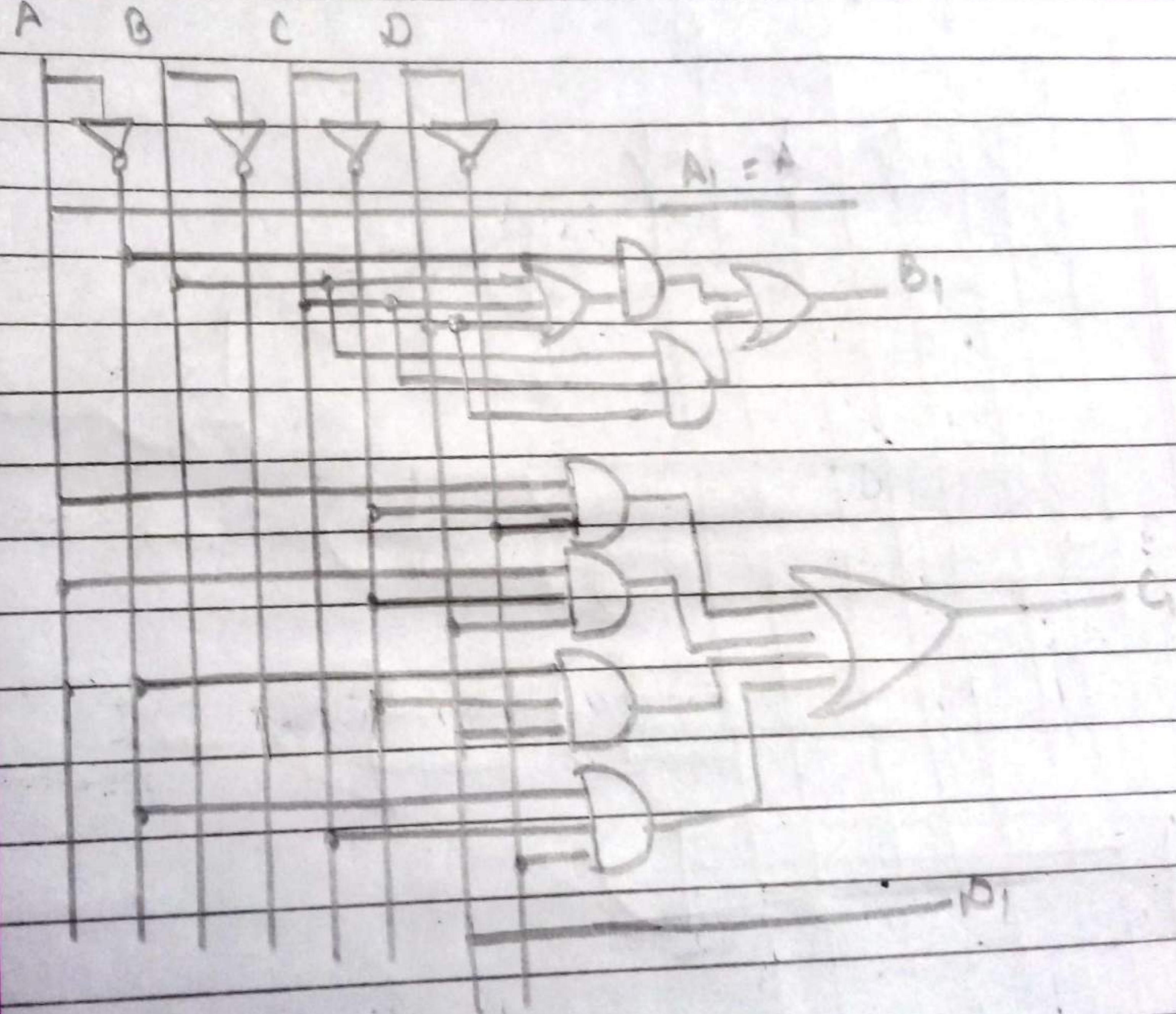
From KMap, $D_1 = D$

The Boolean expression for 24 21 to 8 4 -2 -1 code converter are:

$A_1 = A$, $B_1 = \bar{A}B + \bar{A}C + \bar{A}D + BCD = \bar{A}(B+C+D) + BCD$

$C_1 = A\overline{CD} + ACD + \bar{A}CD + \bar{A}C\bar{D}$, $D_1 = D$

9. Logic Diagram.



Saroj Dahal - 191725

Date / /

Page No.

10. Define literal and term. Find the canonical SOP for the expression. $F = ac + ab + bc$.

Ans A literal is the variable or complement of a variable used in boolean functions.

A term is the expression formed by literal and operation.

For example:

$F = XY + X\bar{Y}Z + \bar{X}YZ$, which has three variables, (X, Y, Z), 8 literals($X, Y, X, \bar{Y}, Z, \bar{X}, Y, Z$) and 4 terms [$XY, X\bar{Y}Z, \bar{X}YZ$ and OR(+)].

* $F = ac + ab + bc$

For Canonical SOP, $F = a(b+\bar{b})c + ab(c+\bar{c}) + (a+\bar{a})bc$

$= abc + a\bar{b}c + abc + ab\bar{c} + abc + \bar{a}bc$

$= abc + \bar{a}bc + a\bar{b}c + ab\bar{c}$,

is required canonical SOP form.

11. Design a circuit of a ~~9 bit~~ 3 bit parity generator and the circuit of 4 bit parity checker for odd parity

Ans. Parity bit is an extra bit included with binary message to make the number either odd or even.

3-bit parity generator:

Let three inputs be represented by X, Y, Z respectively and output by P.

So, we have:

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

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

Date / /

Page No.

X	Y	Z	P (odd)
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

K-Map

X \ YZ	00	01	11	10
0	1	0	1	0
1	0	1	0	1

$\therefore P = \bar{X}\bar{Y}\bar{Z} + X\bar{Y}Z + XY\bar{Z} + \bar{X}YZ$

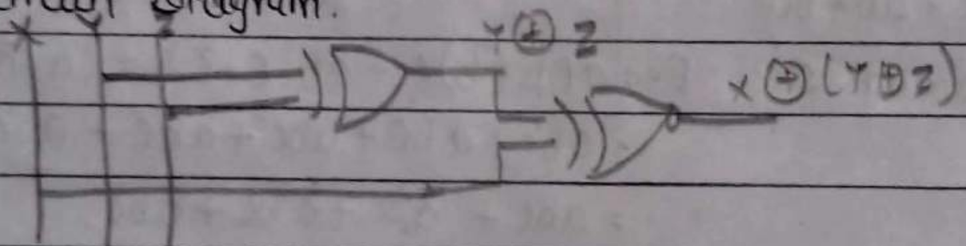
$= \bar{X}\bar{Y}\bar{Z} + \bar{X}YZ + X\bar{Y}Z + XY\bar{Z}$

$= \bar{X}(\bar{Y}\bar{Z} + YZ) + X(\bar{Y}Z + Y\bar{Z})$

$= X(Y \oplus Z) + \bar{X}(\overline{Y \oplus Z})$

$= X \oplus (Y \oplus Z)$

Circuit Diagram.



* Designing a 4 bit parity checker for odd parity.

4-bit message Parity checker.

X	Y	Z	P	C
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

K-Map.

XY \ ZP	00	01	11	10
00	1	0	1	0
01	0	1	0	1
11	1	0	1	0
10	0	1	0	1

$\therefore C = \bar{X}\bar{Y}\bar{Z}\bar{P} + \bar{X}\bar{Y}ZP + \bar{X}Y\bar{Z}P + \bar{X}YZ\bar{P} +$

$XY\bar{Z}\bar{P} + X\bar{Y}\bar{Z}P + X\bar{Y}Z\bar{P} + XYZP$

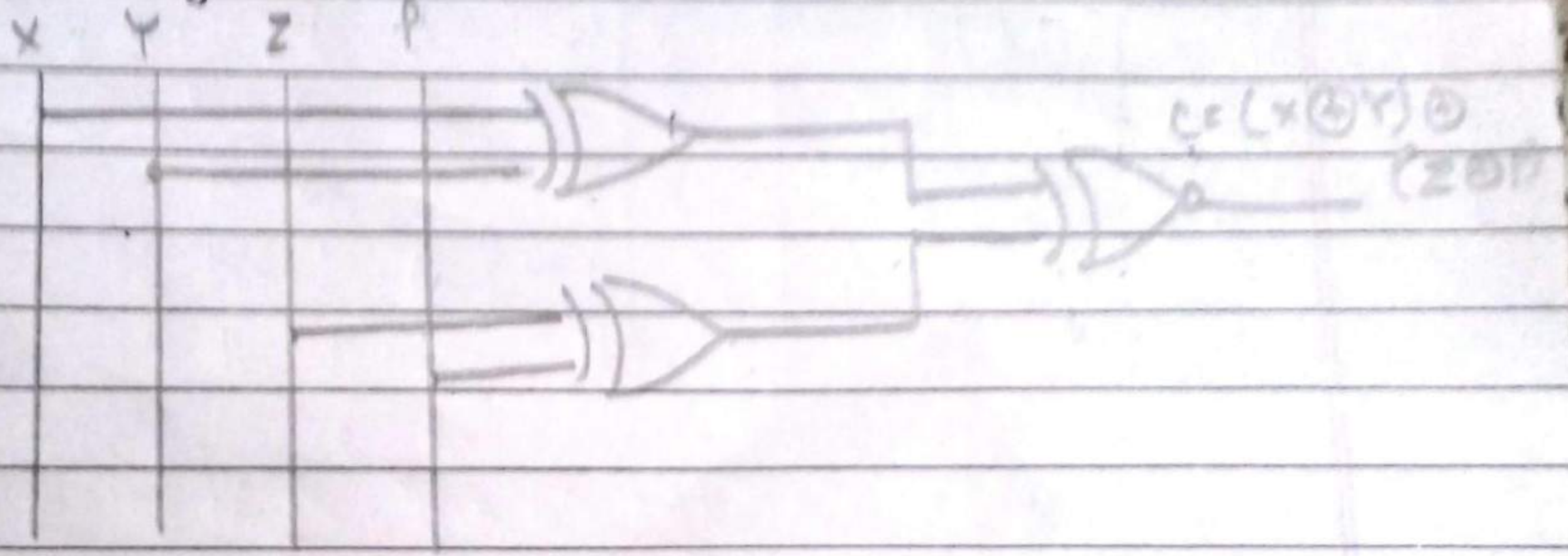
$= \bar{X}\bar{Y}(\bar{Z}\bar{P} + ZP) + \bar{X}Y(\bar{Z}P + Z\bar{P}) +$

$X\bar{Z}(Y\bar{P} + \bar{Y}P) + XZ(\bar{Y}\bar{P} + YP)$

$= (X \oplus Y) \oplus (Z \oplus P)$

Saroj Dahal - 191725

11. Circuit Diagram.



Saroj Dahal - 191725

12. Design a combinational circuit that has four inputs and two outputs, one of the output is high when majority of inputs are high and second output is high only when all inputs are of same type.

Ans. Let four inputs be A, B, C, D and two outputs be Y and Z.
So, The truth table is,

A	B	C	D	Y	Z
0	0	0	0	0	1
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	0
0	1	0	1	0	0
0	1	1	0	0	0
0	1	1	1	1	0
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	0	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	1	1

K-Map for Output Y.

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	0	1	1	1
10	0	0	1	0

$Y = AB(C+D) + CD(A+B)$

K-Map for Output Z.

AB \ CD	00	01	11	10
00	1	0	0	0
01	0	0	0	0
11	0	0	1	0
10	0	0	0	0

$\bar{A}\bar{B}.\bar{C}\bar{D}$ $ABCD$

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

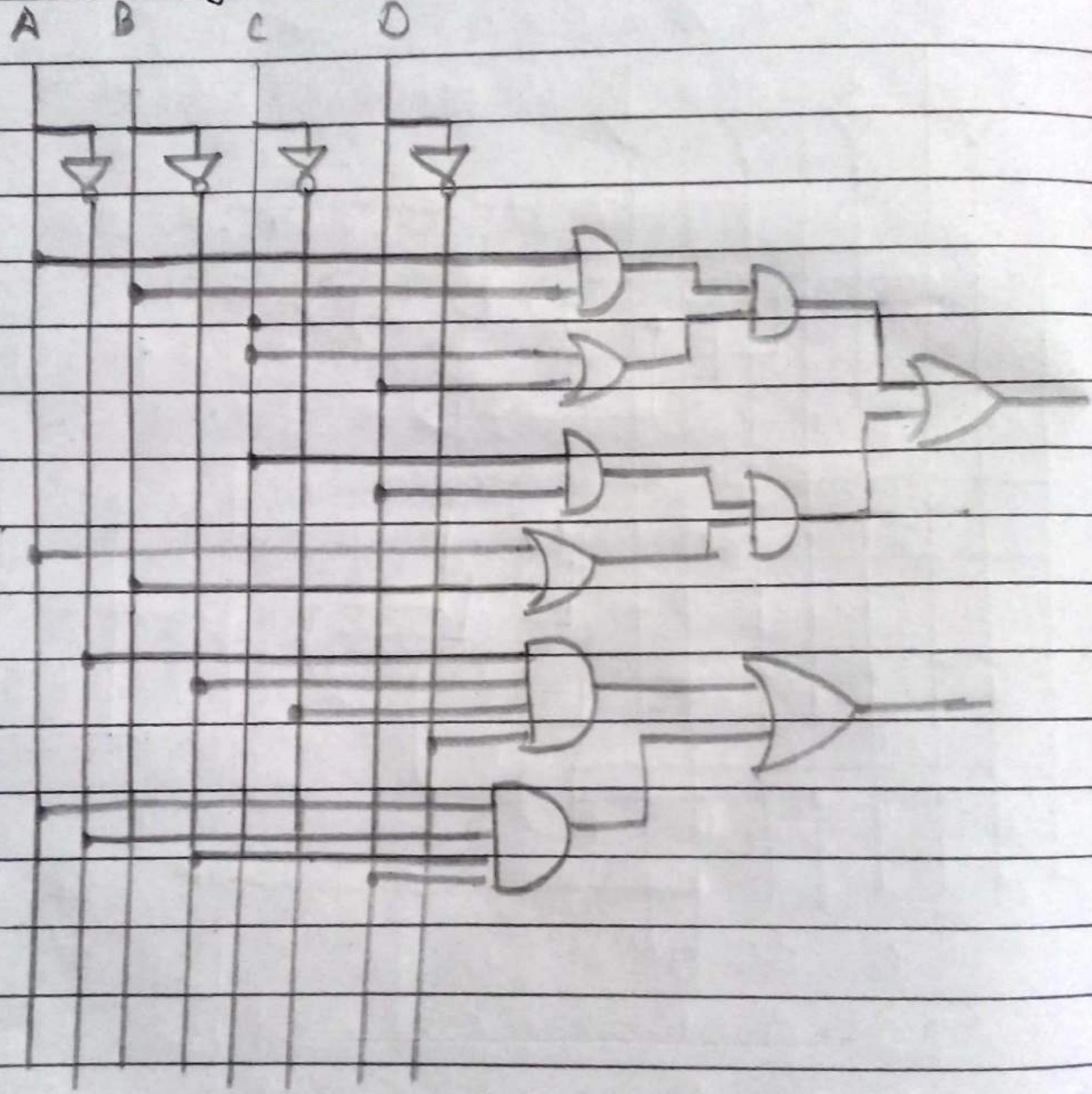
For K-Map of Y, $Y = BCD + ABD + ABC + ACD$

$= ACD + BCD + ABC + ABD$

$= CD(A+B) + AB(C+D)$

Date / /
Page No.

12. Circuit Diagram.



Date / /
Page No.

13 Use K.Map to simplify the given Boolean function with don't care condition and realize it using only basic gates. $F = \Sigma(1, 4, 8, 12, 13, 15)$ $d = (3, 7, 11, 14)$.

Ans Let A, B, C, D be inputs.

$\bar{A}\bar{B}\bar{C}D$; (CD) ; AB ; $B\bar{C}\bar{D}$; $A\bar{C}\bar{D}$

AB \ CD	00	01	11	10
00	0 [0]	1 [1]	x [3]	0 [2]
01	1 [4]	0 [5]	x [7]	0 [6]
11	1 [12]	1 [13]	1 [15]	x [14]
10	1 [8]	0 [9]	x [11]	0 [10]

so we get, $AB + CD + B\bar{C}\bar{D} + A\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D$