Handwritten Notes

On

Digital Electronics

Unit : III (Tuesday)

Digital Electronics

A Digital Electronics or Digital circuit operate with Binary Number & in Binary Number their exist only two states which is either "zero" or "One".

→ In +ve Logic system Binary number 1 represent high state & 0 represent low state.

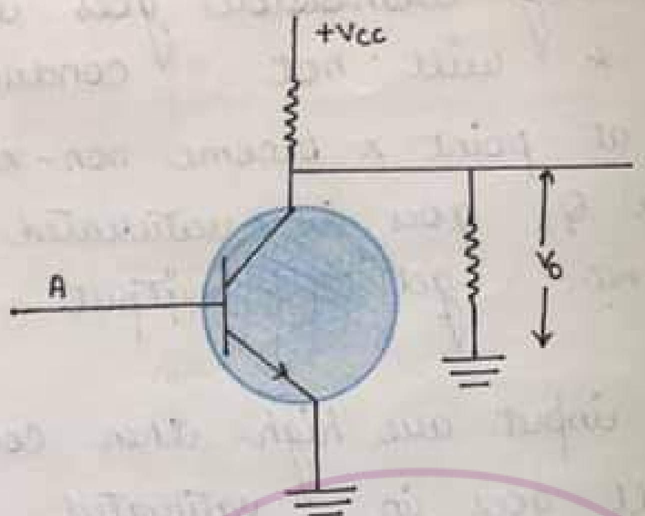
→ In -ve logic system Binary Number 0 represent high state & 1 represent low state.

→ In digital circuit the various circuit perform operation by counting digit.

Logic Gate :

The circuits which are used to provide logical output, are k/as "Logic Gate".

Transistor-Transistor Logic (TTL)

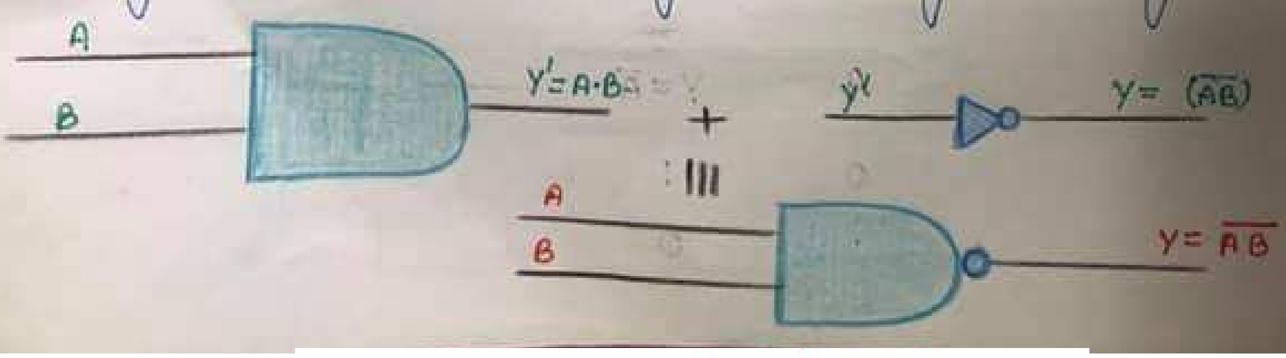


If input A is in low state then transistor goes in cutoff region & we get high output if input is in high state then transistor goes in saturated region & we get low output.

NOT Gate work as Inverter.

- **NAND Gate:**

NAND Gate is a combination of NOT - AND gate. It has two or more input & only single output. So, logic symbol for NAND gate is given by;



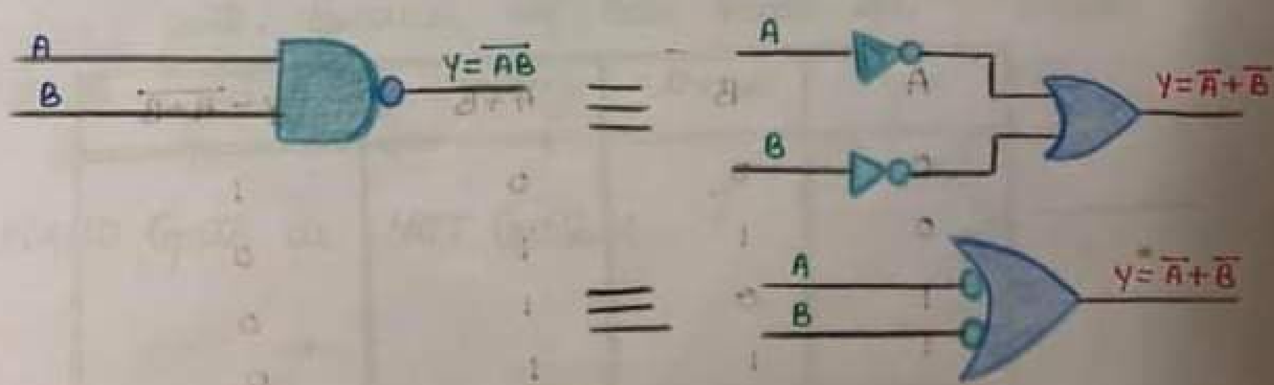
Truth Table:

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

According to Demorgan Theorem the compliment of a product is equal to sum of compliment. If A & B are input variable then, according to Demorgan theorem.

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

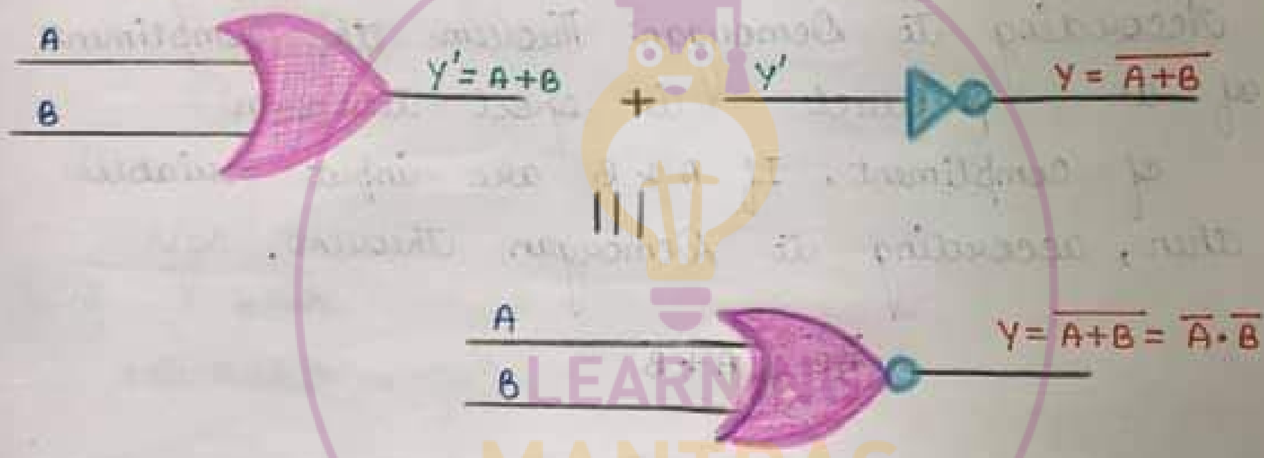
A	B	$\overline{A}$	$\overline{B}$	$\overline{A}+\overline{B}$
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

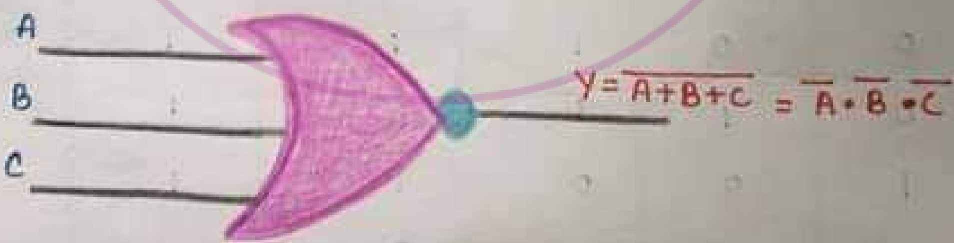


NAND Gate ≡ Bubbled OR Gate

- **NOR Gate:**

NOR gate is a combination of NOT-OR gate it has two or more input but have only single output if A & B are input variable then logic symbol can be express as;





Truth Table:

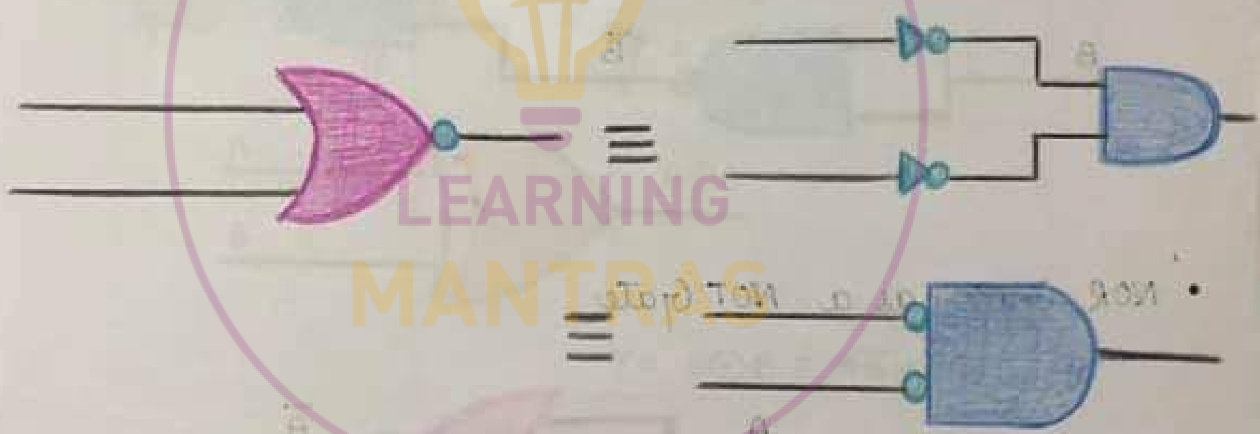
A	B	A+B	Y = $\overline{A+B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

According to De-Morgan theorem

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

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

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

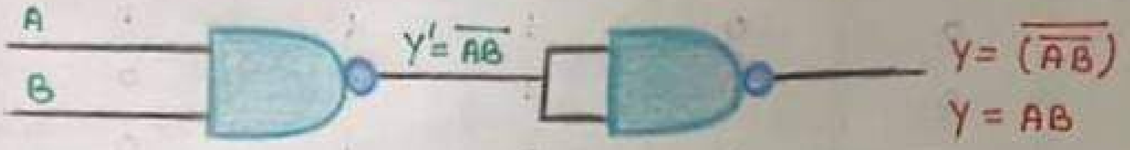


NOR Gate = Bubbled AND Gate

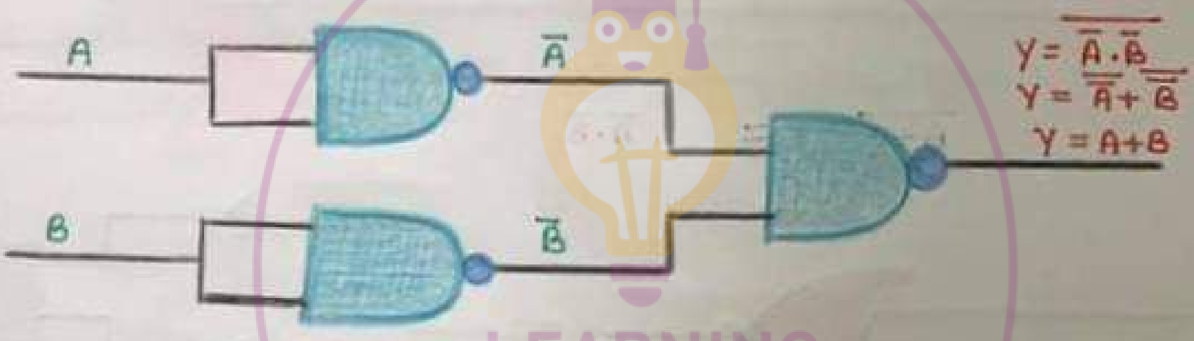
Note: "NAND" and "NOR" are "universal" logic gate. Because we can made all other logic gates by using these logic gates.

- NAND Gate as NOT Gate :

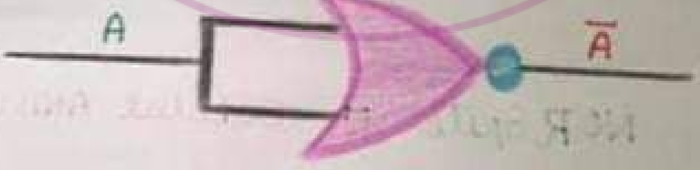
- NAND Gate as AND Gate :



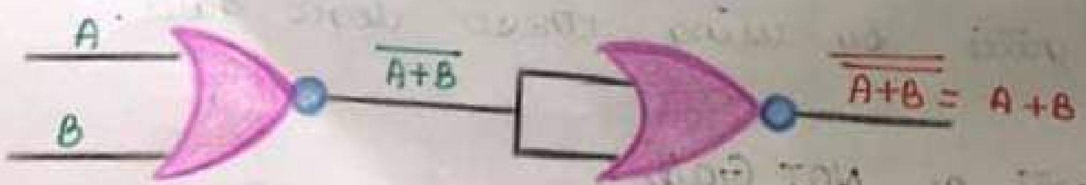
- NAND Gate as OR Gate :



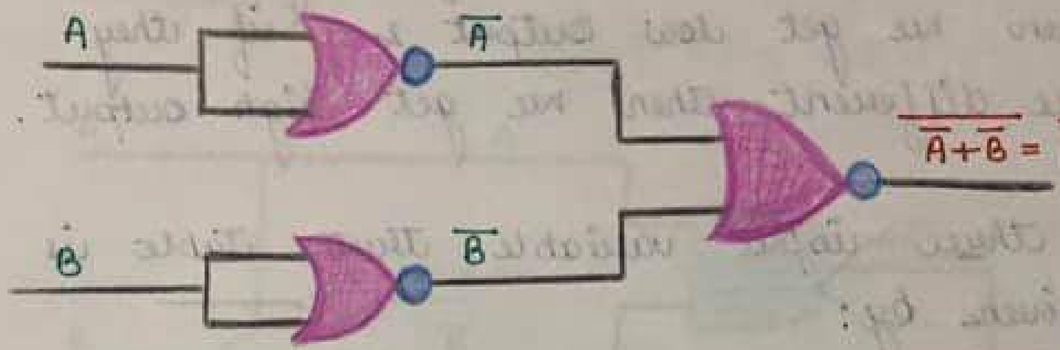
- NOR Gate as a NOT Gate



- NOR Gate as a OR Gate :



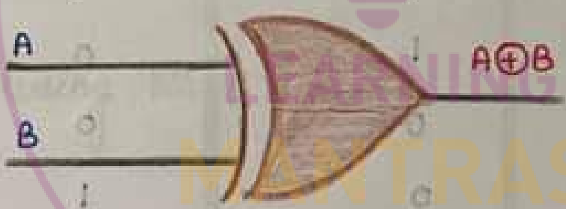
• NOR Gate as AND Gate:



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

• XOR Gate (Exclusive-OR Gate): (7th february'2019) (Thursday)

A XOR Gate have two or more input but have only single output, the logic symbol for XOR Gate is given by;



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

Truth Table:

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

For two input variable if they are same then we get low output & if they are different then we get high output.

For three input variable truth table is given by:

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

The another logic expression of XOR Gate

$$Y = A \oplus B = A\overline{B} + B\overline{A} = (A+B)\overline{AB}$$

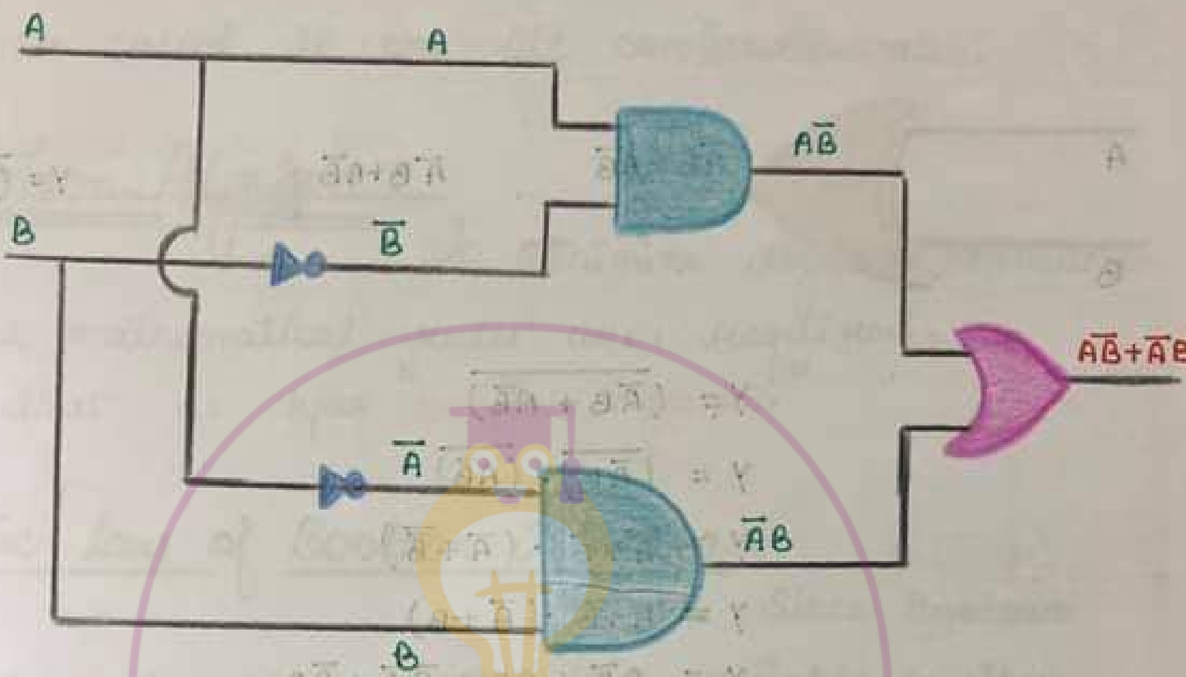
$$= (A+B)(\overline{A}+\overline{B})$$

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

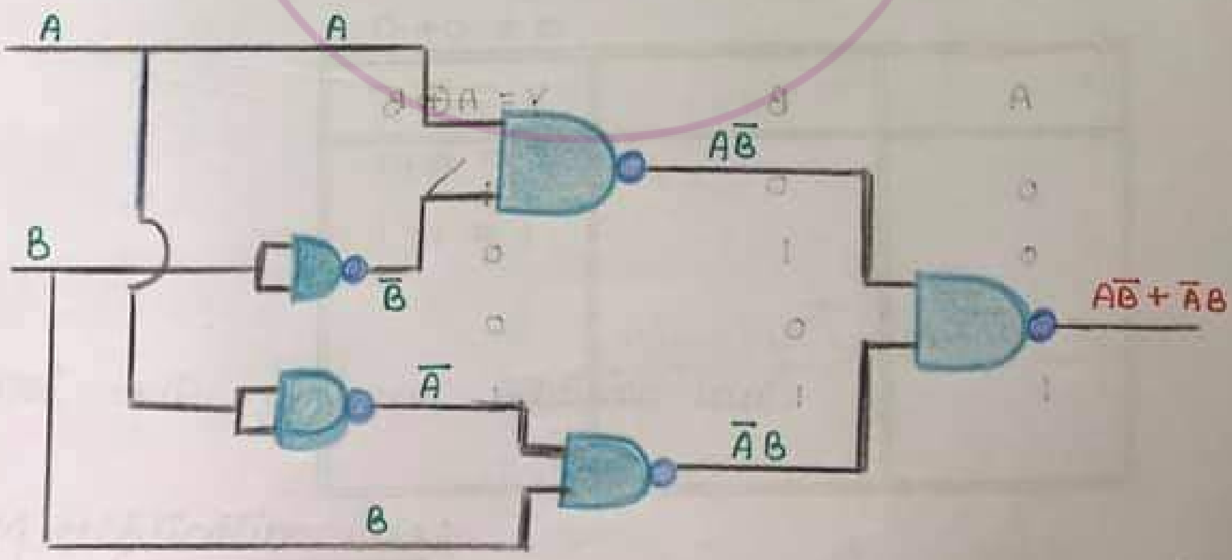
$$\Rightarrow A\overline{A} = B\overline{B} = 0$$

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

Equivalent Circuit of XOR Gate using NOT, AND & OR Gate.

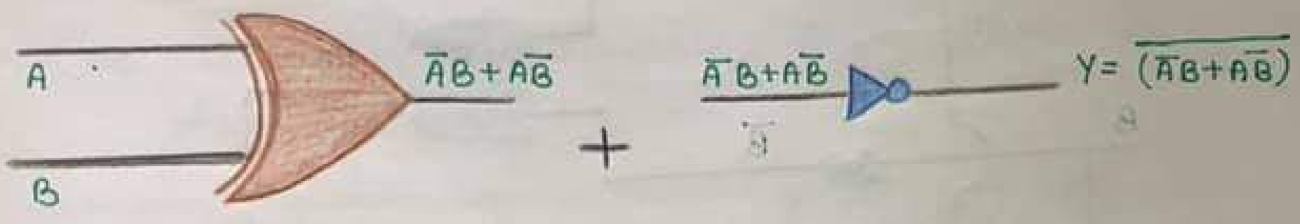


XOR Gate using NAND Gate :



• Exclusive NOR Gate (X-NOR Gate):

X-NOR = X-OR + NOT



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

$$Y=(\overline{\bar{A}B})\cdot(\overline{A\bar{B}})$$

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

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

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

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

Truth Table:

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

From truth table output is high only when both input are same [low or high].

If A=B then Y=1
If A≠B then Y=0

So, we called it one bit comparator also.

· Boolean Algebra:

To minimize logical expression some mathematical rules are required, that is k/as "Boolean Algebra."

· Basic Law of Boolean Algebra:

Since Boolean algebra depend on arthematic variable & they have value either zero & one.

→ Addition Law:

$0+0=0$
$0+1=1$
$1+0=1$
$1+1=1$

'OR' Gate perform Addition Law.

→ Multiplication Law:

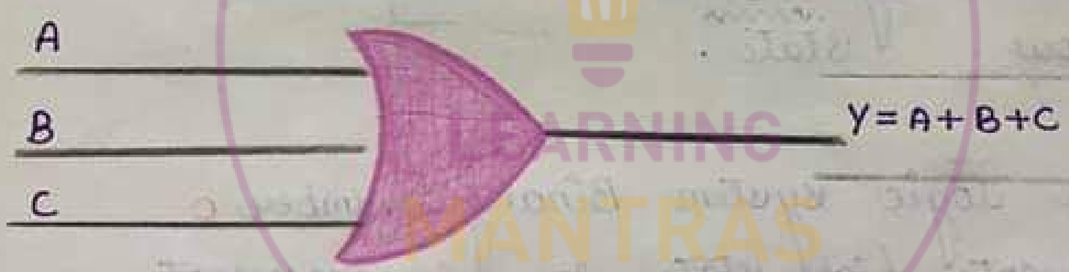
$0\cdot 0=0$
$0\cdot 1=0$
$1\cdot 0=0$
$1\cdot 1=1$

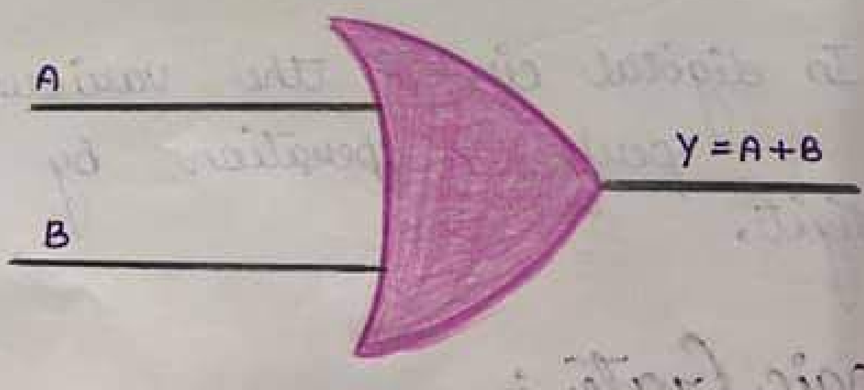
Logic Gate have many input but only one output. We can perform arthematic operation by using Logic Gate Circuit..

Example: AND, OR, NOT, NAND, NOR, X-OR, X-NOR....

- **OR-Gate:**

OR-Gate have two or more input but have only one output. The output of OR-Gate is high when any one of input is high. If A & B are input variable of OR gate then, output Y is represented by;





"OR" Gate perform Plus (+) operation.

so, Boolean Multiplication is same as AND operation.

Properties of Boolean Algebra:

Boolean Algebra is a Mathematical system which consist two or more variable and depend on two operation which is denoted by OR(+) and AND(·).

Property:

→ **Commutative Property:** Boolean Algebra follow commutative law for both addition law and Multiplication law.

$$A+B = B+A$$

$$A \cdot B = B \cdot A$$

→ **Associative Property:** The associative property for addition & multiplication is given by

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

$$(A \cdot B) \cdot C = A \cdot (B \cdot C)$$

→ **Distributive Property:** The Boolean Addition is distributive over Boolean multiplication which is given by

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

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

→ **Absorption law:**

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

→ Idempotent law :

$A + A = A$, $A \cdot A = A$

→ Double Inversion law :

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

OR operation	AND operation
$A + 0 = A$	$A \cdot 0 = 0$
$A + 1 = 1$	$A \cdot 1 = A$
$A + \overline{A} = 1$	$A \cdot \overline{A} = 0$

De-Morgan Theorem:

De Morgan has two statement.

Statement : I

Compliment of the sum of A & B is equal to the product of compliment of A & B.

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

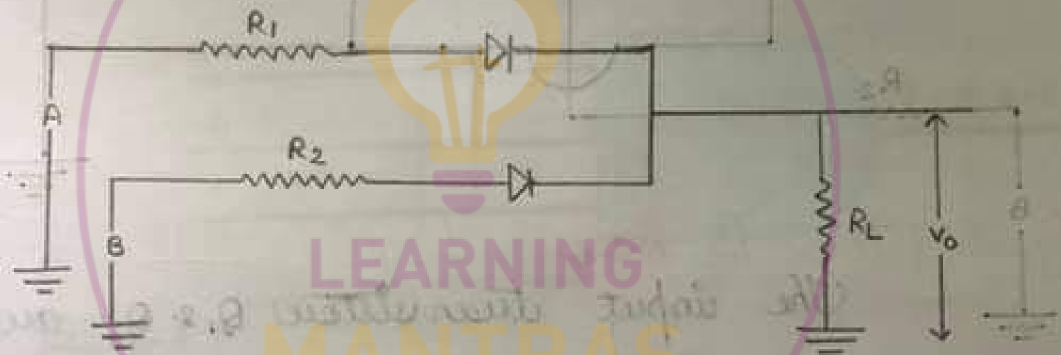
Statement : II

Compliment of product of A & B is equal to the sum of compliment of A & B.

Truth Table:

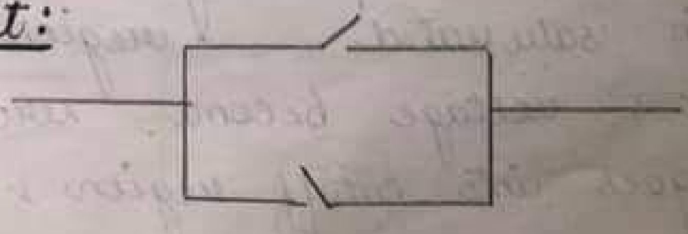
A	B	Y = A + B
0	0	0
0	1	1
1	0	1
1	1	1

OR-Gate Using Diode Logic : (6th february 2019) Wednesday

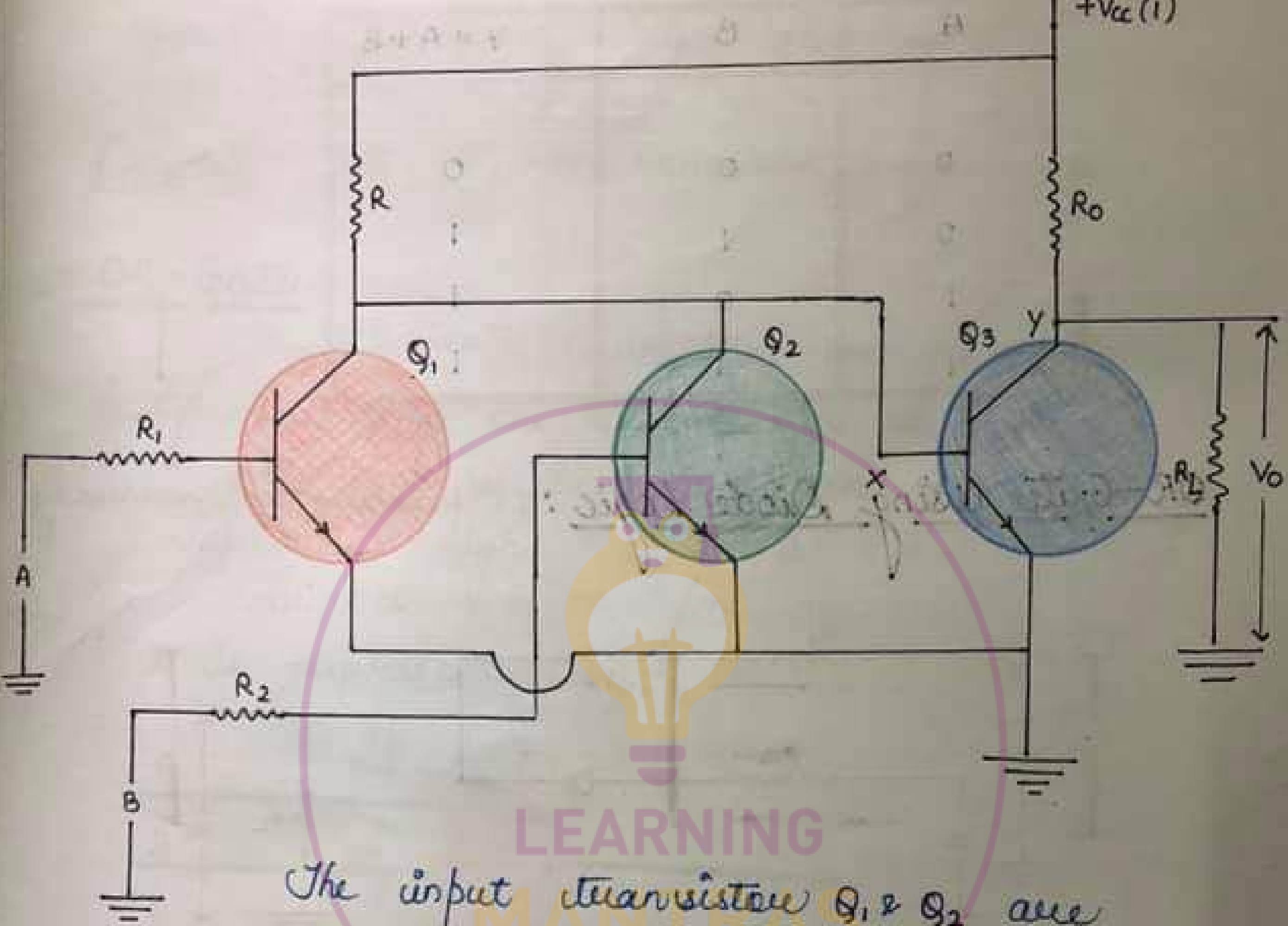


If both input are low then both diode are in reverse bias & will not conduct so, we get "0" output.
If any 1 input is high then corresponding diode in forward bias & we get high output.

Switching Circuit:



• Transistor - Transistor Logic (TTL) :



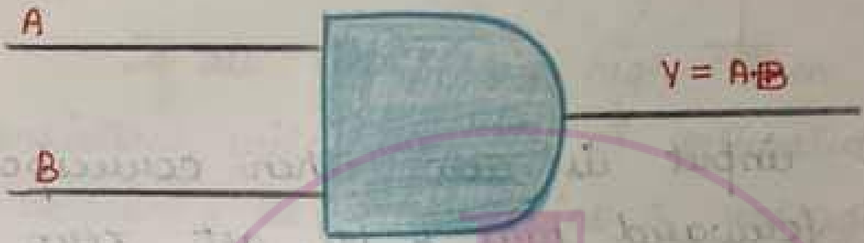
The input transistor Q_1 & Q_2 are connected || to eachother if both input are low then transistor Q_1 & Q_2 are in cutoff region so, voltage at point x is not equal to zero & transistor Q_3 work in saturated region & we get zero output.

If any one of the input is high then corresponding transistor is in saturated region & at point x voltage become zero. So, transistor Q_3 goes into cutoff region & we get high output.

• AND Gate:

'AND' Gate perform multiply operation. It provide output only when all inputs are high logic [1].

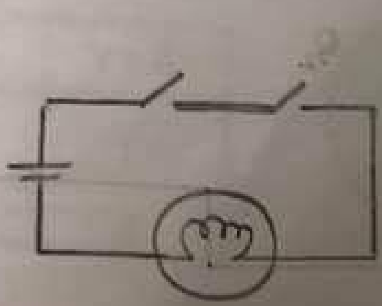
It is represented by;



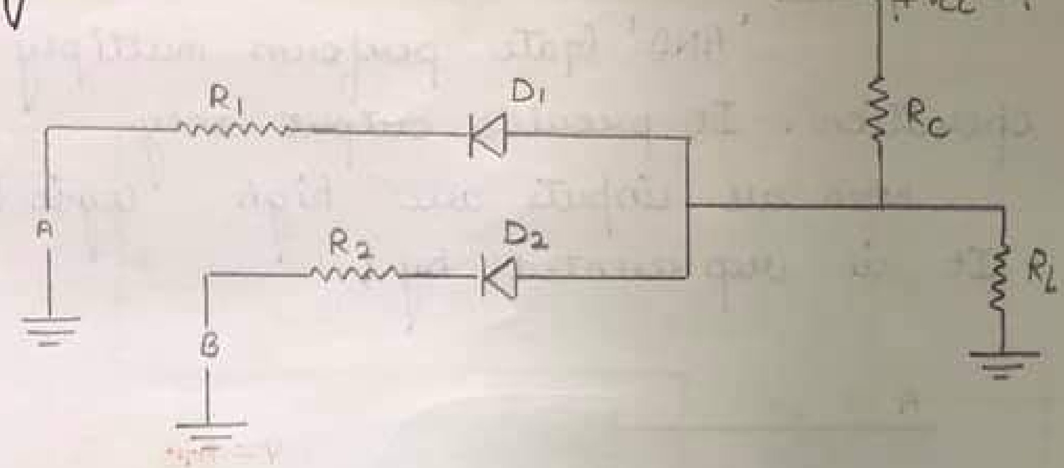
Truth Table:

A	B	Y = A·B
0	0	0
0	1	0
1	0	0
1	1	1

Switching Circuit:

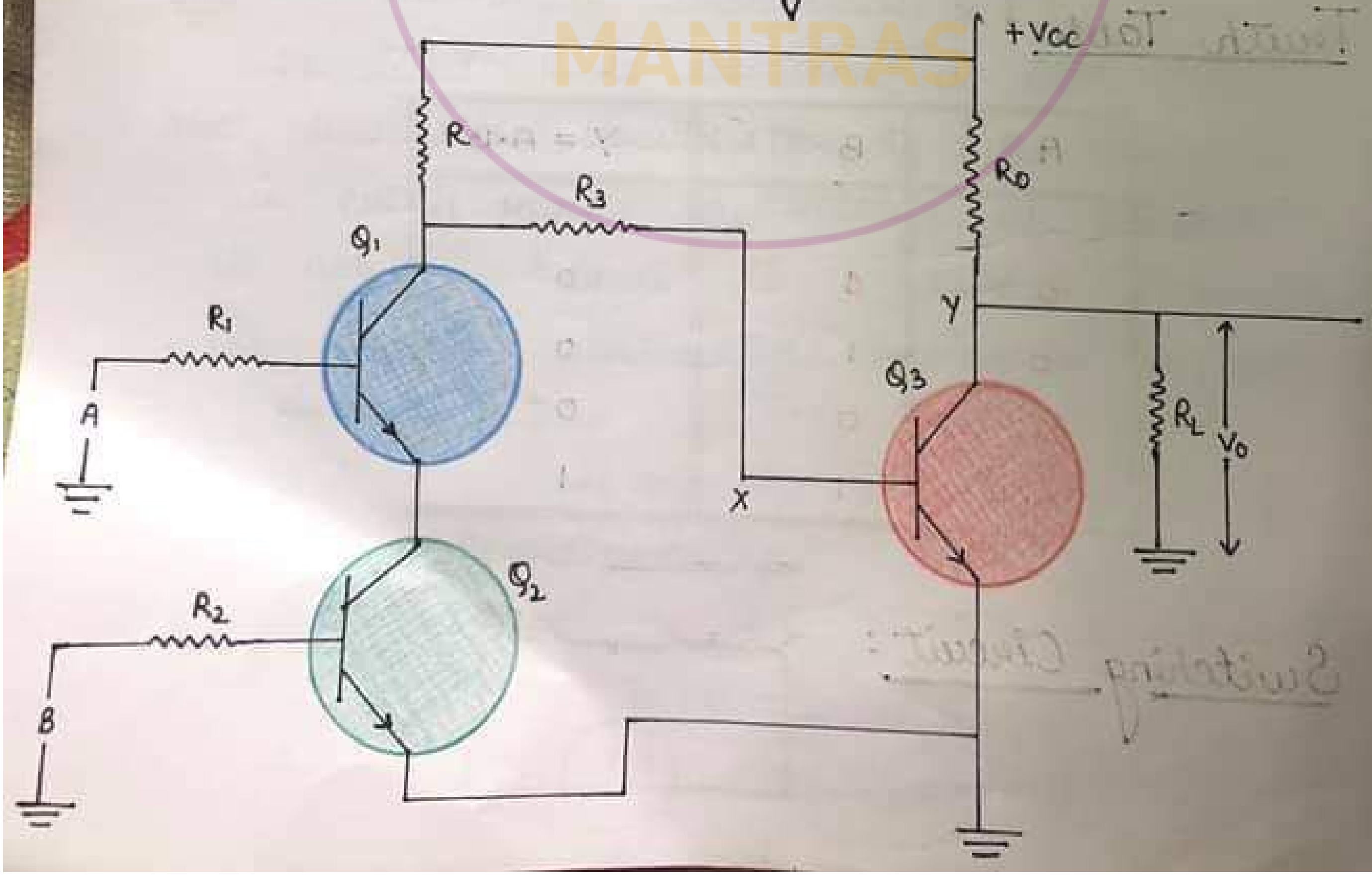


Diode Logic Circuit:



If any input is low then corresponding diode in forward bias & we get zero output if all input are high then corresponding all diode are in reverse bias & we get high output.

Transistor-Transistor Logic (TTL):



If any one of the input is low then corresponding transistor goes in cutoff region & will not conduct so, voltage at point x become non-zero & transistor Q_3 goes in saturated region & we get zero output.

If all input are high then corresponding transistor will goes in saturated region. So, at point x voltage become zero & transistor Q_3 goes in cutoff region & we get high output.

- NOT Gate:

'NOT' Gate perform reverse operation means when input is low then output is high and vice-versa.

It is represented by;

Truth Table:

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