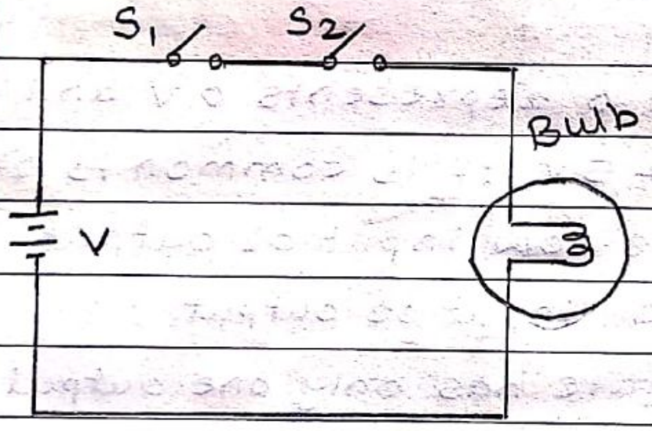
UNIT-II

LOGIC GATES

A digital circuit with one or more input signals but only one output signal is called a logic gate.

Since a logic gate is a switching circuit (i.e. a digital circuit), its output can have only one of the two possible states viz, either a high voltage (1) or a low voltage (0) - it is either ON or OFF. Wheather the output voltage of a logic gate is high (1) or low (0) will depend upon the conditions at its input. Below Fig. shows the basic idea of a logic gate using switches.



S_1	S_2	Bulb
open	open	OFF
open	closed	OFF
closed	open	OFF
closed	closed	ON.

Truth Table

S_1	S_2	output
0	0	0
0	1	0
1	0	0
1	1	1

Three Basic Logic Gates

Logic gate is a circuit that has one or more input signals but only one output signal. All logic gates can be analysed by constructing a truth table. A truth table lists all possibilities and the corresponding output for each input. The three basic logic gates that make up all digital circuits are

<I> OR gate

<II> AND gate

<III> NOT gate.

The following points may be noted about logic gates

(i) A binary 0 represents 0 V and binary 1 represents +5 V. It is common to refer to binary 0 as LOW input or output and the binary 1 as HIGH input or output.

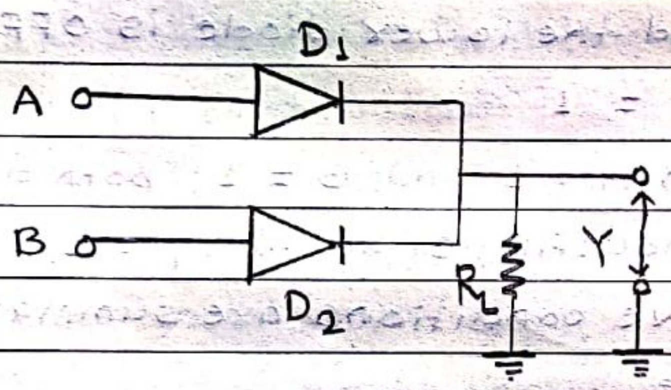
(ii) A logic gate has only one output signal The output will depend upon the input signal/ signals and the type of gate

(iii) The operation of a logic gate may be described either by truth table or Boolean algebra.

* <I> OR Gate

The OR gate is one of the fundamental or basic gate. The OR gate has two or more inputs signals but only one output signal. The output voltage is high if any or all of the input voltages are high.

Fig. 1 shows two input OR gate by using diodes. The input ~~are~~ voltages are labelled A and B while the output voltage is Y.



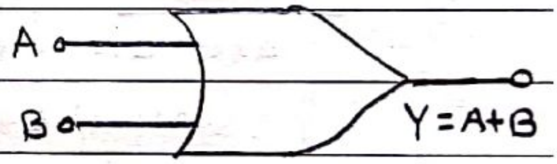


Fig. 1 OR Gate

Fig. 2 Logic Symbol

Let us assume the input voltages are either 0 V (low state) or +5 V (high state). There are only four possible cases.

Truth Table

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

Fig. 2 shows the logic symbol of two input OR gate.

→ case 1 - When A = 0 and B = 0, both diodes are non conducting, the output voltage is low.

→ case 2 - When A = 0 and B = 1 the high B input voltage and forward bias the lower diode, producing an output voltage. The upper diode is non conducting state so Y = 0 + 1 = 1

→ case 3 - When A = 1 and B = 0, the upper diode is on and the lower diode is OFF and Y = 1 + 0 = 1

→ case 4 - When A = 1 and B = 1, both diodes are ON i.e conducting state ∴ Y = 1.

All the above conditions are summarised in the truth table.

* <II> AND Gate

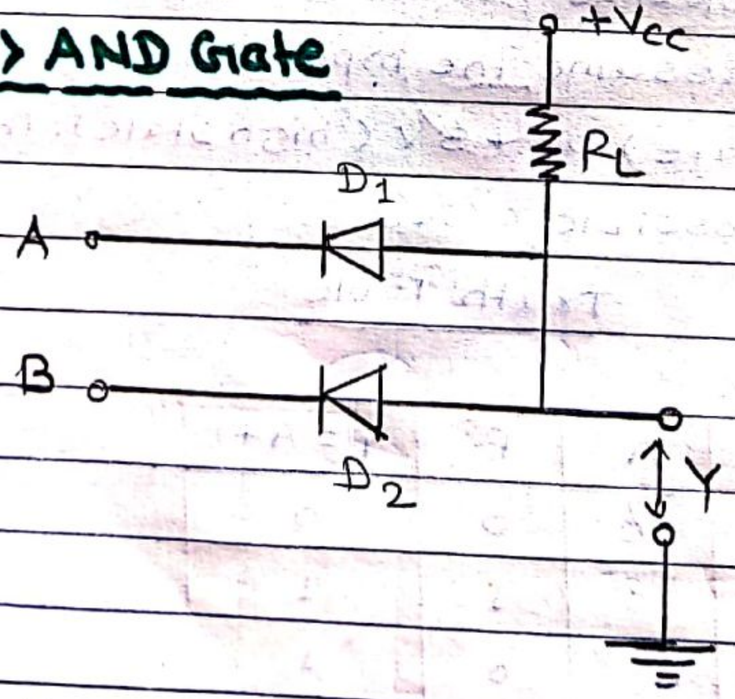
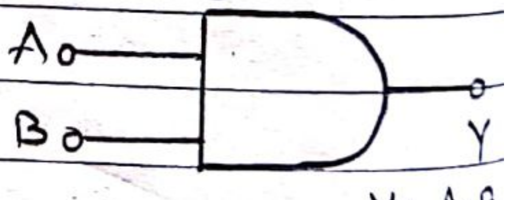


Fig 1 AND Gate



Y = A.B

Fig 2 Logic Symbol

Truth Table

A	B	Y = A·B
0	0	0
0	1	0
1	0	0
1	1	1

The AND gate is a logic gate that has two or more inputs but only one output. The output Y of AND gate is HIGH when all inputs are HIGH. However, the output Y of AND gate is LOW if any or all inputs are LOW.

The Fig. 1 shows two input AND gate using diodes. The input voltages are labelled A and B, while the output voltage is Y There are only four input-output possibilities.

→ **case 1 :-** When both A = 0 and B = 0, with both input voltages are low, both the diodes are grounded and output voltage is zero i.e

$Y = 0 \cdot 0 = 0$

→ **case 2 :-** When A = 0 and B = 1, since A is low the upper diode is forward biased, pulling the output down to a low voltage With the B input high, the lower diode goes into reverse bias. so

$Y = 0 \cdot 1 = 0$.

→ **Case 3:-** When A = 1 and B = 0, because of the similarity of the circuit, the action is similar to above case. The upper diode is off and the lower diode is on and Y is low i.e.

$$Y = 1 . 0 = 0$$

→ **Case 4:-** When A = 1 and B = 1, with both inputs at +5V, both the diodes are non-conducting. since the diodes are off there is no current through R_L and the output is pulled up the supply voltage. Therefore Y is high i.e. $Y = 1 . 1 = 1$

It is clear from the turth table that for AND gate, the output is high if all the inputs are high. Fig. 2 shows the logic symbol of AND gate.

* ⟨III⟩ ● NOT Gate

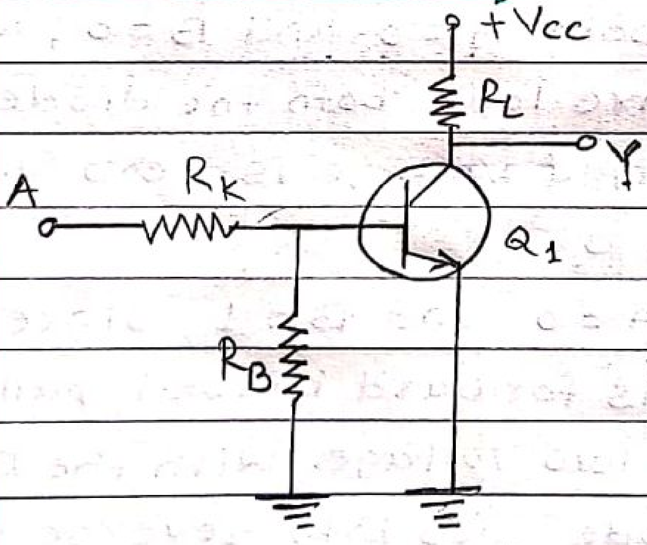


Fig 1. Not Gate

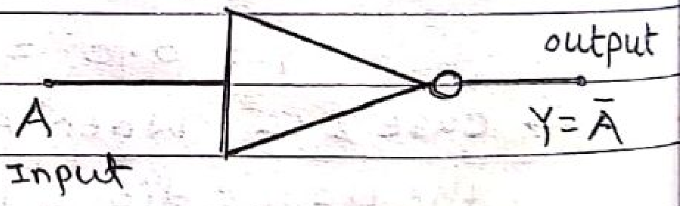


Fig 2. Logic Symbol

The NOT gate is one of the fundamental or basic logic gate. It is also called as inverter. It has only one input and one output, where the output is opposite of the input. The NOT gate is often called inverter because it inverts the input.

Fig. 1 shows a typical inverter circuit. When A is connected to ground, the base of transistor Q_1 will become negative. This negative potential causes the transistor to cut off and collector current to zero and output is +V volts. i.e when $A = 0$, $Y = 1$. similarly, when $A = 1$ then $Y = 0$. The truth table for NOT gate is as below

Truth Table

A	Y
0	1
1	0

Fig. 2 shows the logic symbol for NOT gate or inverter. Note that small bubble on the inverter symbol represents inversion.

The boolean expression for NOT function is

$$Y = \bar{A}$$

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

If $A = 1$, then $Y = \bar{1} = 0$.

* NAND Gate

NAND gate is combination of AND gate and NOT gate. In other words, output of AND gate is connected to the input of a NOT gate as in below Fig 1. Fig. 2 shows the logic symbol for a NAND gate.

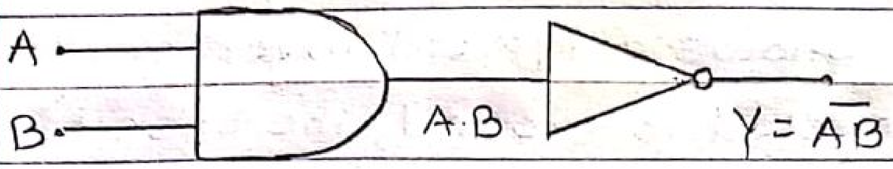


Fig 1 NAND Gate.

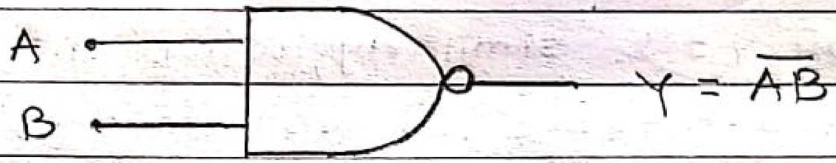


Fig 2 . Logic symbol

Truth Table

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

The NAND gate has two or more input signals but only one output signal. All input signals must be high to get a low output. If one or more inputs are low, the result of AND gate is low, therefore the final inverted output is high.

Page No.:
Date:

* NOR Gate

It is a combination of OR gate and NOT gate In other words, output of OR gate is connected to the input of a NOT gate as shown in Fig. 1. Fig. 2 shows the logic symbol of NOR gate Note that output of OR gate is inverted to form NOR gate. This is illustrated in the truth table foz NOR gate.

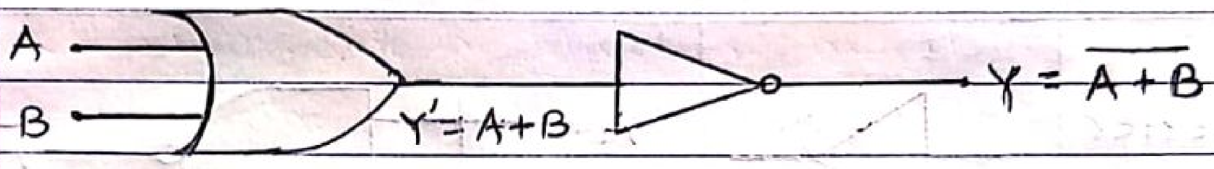


Fig 1 NOR Gate

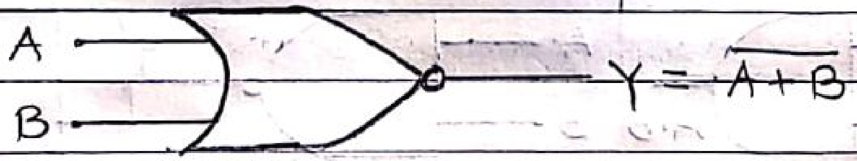


Fig 2 Logic Symbol

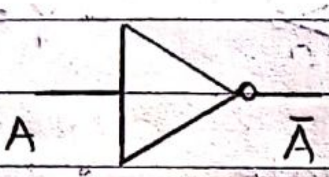
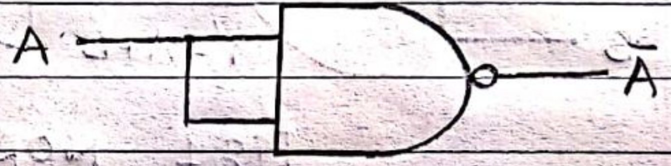
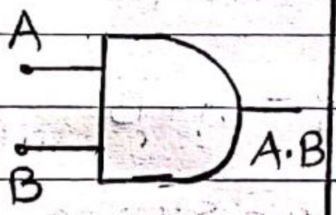
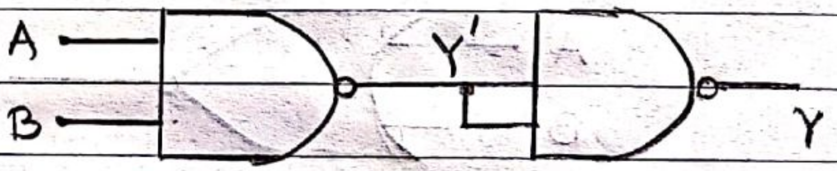
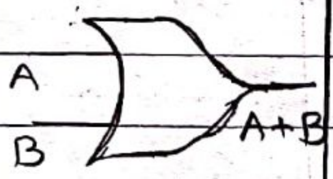
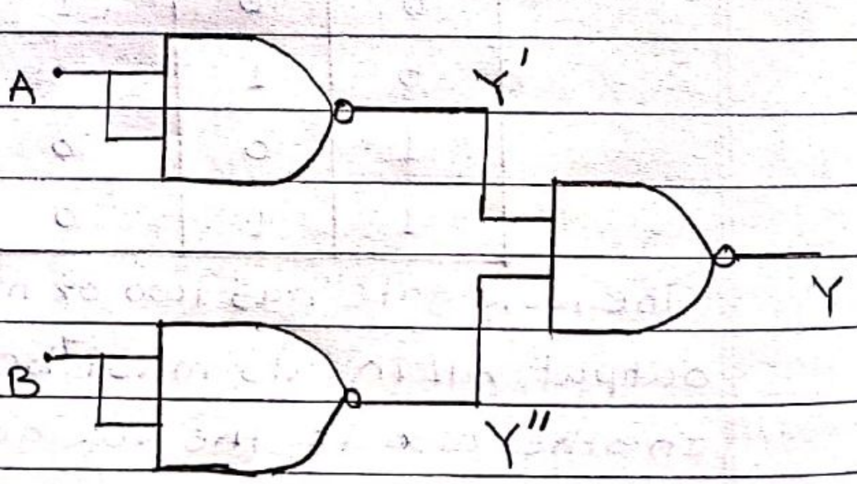
Truth Table

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

The NoR gate has two or more inputs but only one output. All inputs must be low to get high output. In other words, the NOR gate will recognize only the word whose bits are all 0s.

* NAND Gate as a Universal Gate

The NAND gate is universal gate because its repeated use can produce other logic gates. The table below show how NAND gates can be connected to produce inverter (NOT gate), AND gate and OR gate.

Logic Function	Symbol	Circuit using NAND gates only
Inverter		
AND		
OR		

* XOR Gate (Exclusive OR gate)

The name exclusive OR gate is usually shortened to XOR gate. The XOR gate can be obtained by using OR, AND and NOT gates as shown in below Fig.1.

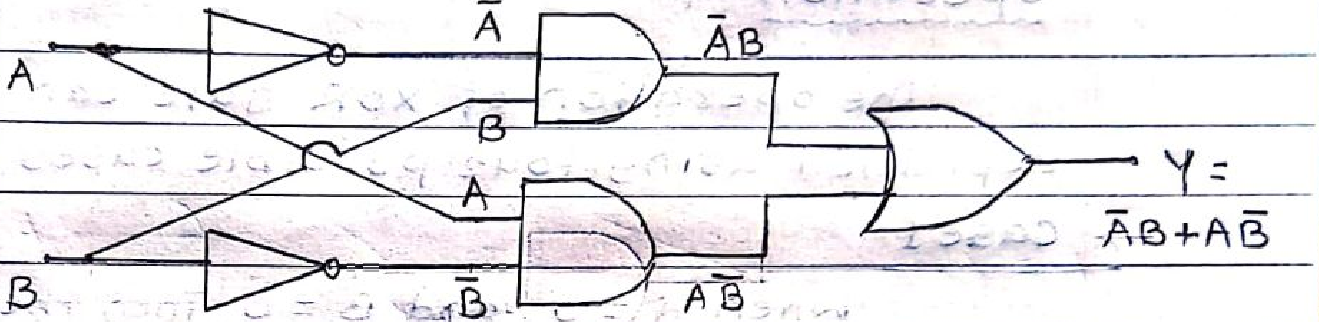


Fig 1 XOR Gate

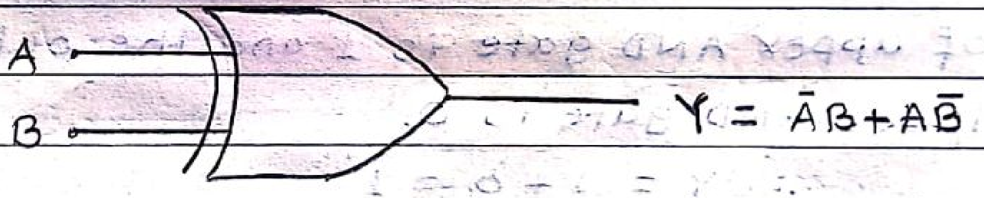


Fig. 2 Logic symbol

Truth Table

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

The logic gate which output is high when an odd no. of inputs is high is called as XOR gate.

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

Operation

The operation of XOR gate can be explained using four possible cases

→ **Case 1**

When $A = 0$ and $B = 0$, Then the output of both AND are 0. $\therefore Y = 0$.

→ **Case 2**

When $A = 0$ and $B = 1$, then output of upper AND gate is 1 and the output of lower AND gate is 0.

$$\therefore Y = 1 + 0 = 1$$

→ **Case 3**

When $A = 1$ and $B = 0$, then the output of upper AND gate is 0 and the output of lower AND gate is 1.

$$\therefore Y = 0 + 1 = 1$$

→ **Case 4**

When $A = 1$ and $B = 1$, then the output of both AND gates are 0.

$$\therefore Y = 0 + 0 = 0$$

* Ex-NOR Gate (Exclusive NOR gate)

The output of XNOR gate is opposite (compliment) that of the XOR gate

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

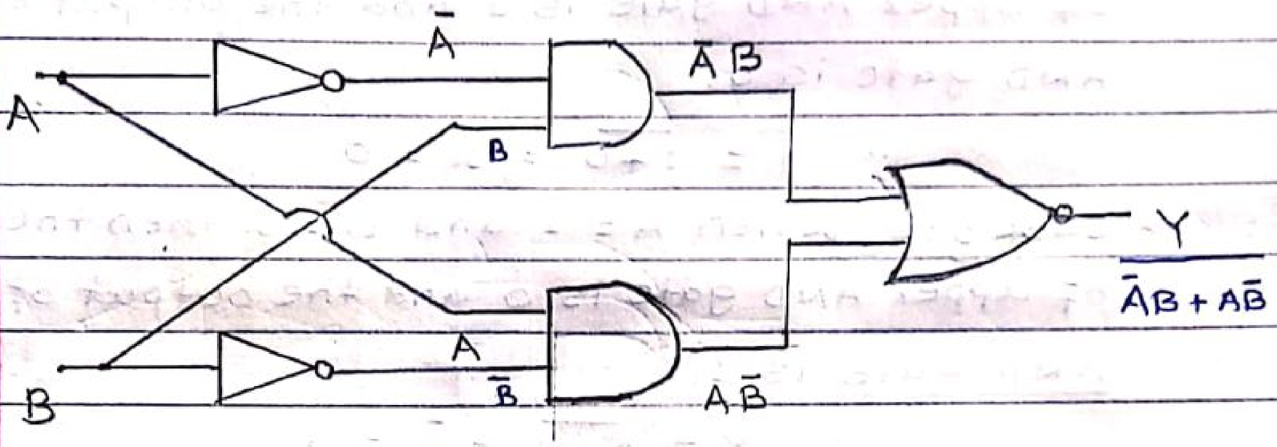


Fig 1 XNOR Gate

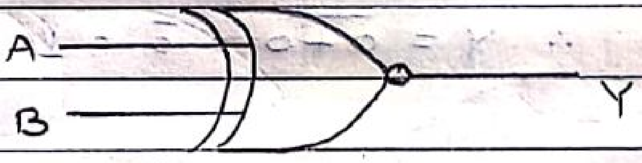


Fig 2. Logic symbol

Truth Table

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

The operation of XNOR gate can be explained using four possible cases:

→ **case1:-** When A=0 and B=0 then the output of both AND gate is 0

$$\therefore \quad Y = \overline{0+0} = \bar{0} = 1 .$$

→ **case2:-** When A=0 and B = 1 then the output of upper AND gate is 1 and the output of lower AND gate is 0.

$$\therefore \quad Y = \overline{1+0} = \bar{1} = 0$$

→ **case3:-** When A = 1 and B = 0 then the output of upper AND gate is 0 and the output of lower AND gate is 1.

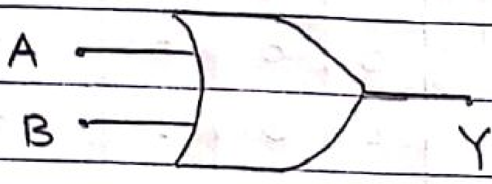
$$\therefore \quad Y = \overline{0+1} = \bar{1} = 0$$

→ **case4:-** When A = 1 and B = 1, then the output of both AND gates are 0.

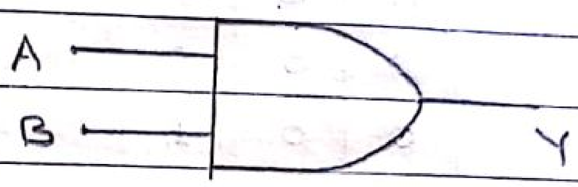
$$\therefore \quad Y = \overline{0+0} = \bar{0} = 1.$$

* Logic Expression for 2,3 and 4 inputs

⟨I⟩ For 2 Inputs



OR Gate



AND Gate

Y = A+B

Y = A.B

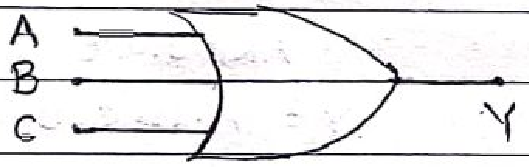
Truth Table For OR gate

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

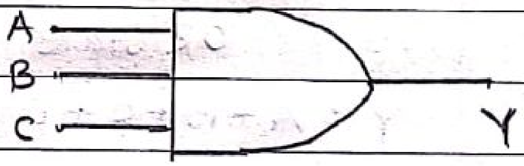
Truth Table for AND gate

A	B	Y = A.B
0	0	0
0	1	0
1	0	0
1	1	1

⟨II⟩ For 3 Inputs



OR Gate



AND Gate

Y = A+B+C

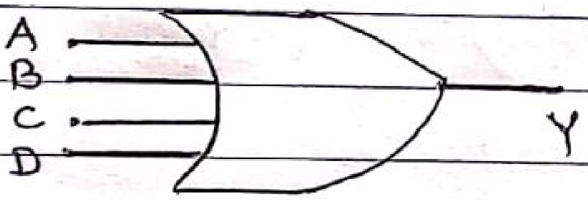
Y = A.B.C

Truth Table

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

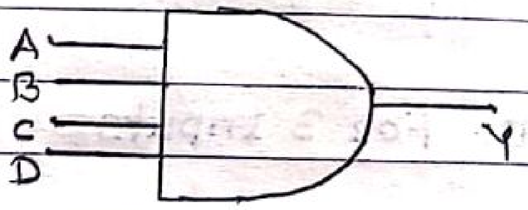
A	B	C	Y = A·B·C
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

⟨III⟩ For 4 Inputs



OR Gate

$Y = A+B+C+D$



AND Gate

$Y = A \cdot B \cdot C \cdot D$

Truth Table

A	B	C	D	Y = A+B+C+D
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1

A	B	C	D	Y = A·B·C·D
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0

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

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

* Laws of Boolean Algebra

Boolean algebra is a system of mathematics based on logic. It has its own set of fundamental laws which are necessary for manupulating different Boolen expressions.

<1> OR laws:

$A + 0 = A$ ------------ (1)

$A + 1 = 1$ ---------- (2)

$A + A = A$ ---------- (3)

$A + \bar{A} = 1$ ---------- (4)

<2> AND laws:

$A \cdot 0 = 0$ ------------- (5)

$A \cdot 1 = A$ ------------ (6)

$A \cdot A = A$ ----------- (7)

$A \cdot \bar{A} = 0$ ----------- (8)

when $A = 0$ then $\bar{A} = 1$ $\therefore A \cdot \bar{A} = 0 \cdot 1 = 0$

when $A = 1$ then $\bar{A} = 0$ $\therefore A \cdot \bar{A} = 1 \cdot 0 = 0$

<3> Laws of complementation:

$\bar{0} = 1$ -------------- (9)

$\bar{1} = 0$ -------------- (10)

If $A = 0$, then $\bar{A} = 1$ ---- (11)

If $A = 1$, then $\bar{A} = 0$ ---- (12)

$\bar{\bar{A}} = A$ ------------ (13)

<4> Commutative laws:

These laws allow change in position of variables in OR and AND expressions

$A + B = B + A$ --------- (14)

$A \cdot B = B \cdot A$ ----------- (15)

<5> Associative laws:

These laws allow removal of brackets from logical expression and regrouping of variables

$A + (B + C) = (A + B) + C$ ------ (16)

$(A + B) + (C + D) = A + B + C + D$ ------ (17)

$A(BC) = (AB)C$ -------- (18)

Also associative laws permit factoring or multiplying out of an expression.

$A(B + C) = AB + AC$ -------- (19)

$A + BC = (A + B)(A + C)$ ------- (20)

$A + \bar{A}B = A + B$ ------- (21)

⟨6⟩ Absorptive laws:

$A + AB = A$ ------------ (22)

$A(A+B) = A$ ------------ (23)

$A(\bar{A}+B) = AB$ ------------ (24)

* De Morgan's Theorems

⟨I⟩ First Theorem

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

The first theorem says that " complement of a sum equals the product of complements."

Let us prepare the table for all values of variables A and B.

1	2	3	4	5	6	7
A	B	$A+B$	$\overline{A+B}$	$\bar{A}$	$\bar{B}$	$\bar{A}\cdot\bar{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

The column 4 and 7 are identical and hence the theorem is proved

When three inputs are involved, De Morgan's first theorem is written as

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

For Four inputs - $\overline{A+B+C+D} = \bar{A}\cdot\bar{B}\cdot\bar{C}\cdot\bar{D}$

and so on.

<II>. Second Theorem

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

The second theorem says that " The complement of a product equals the sum of complements".

Let us prepare the table for all values of variables A and B.

1	2	3	4	5	6	7
A	B	AB	$\overline{AB}$	$\bar{A}$	$\bar{B}$	$\bar{A}+\bar{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

The column 4 and 7 are identical hence the theorem is proved.

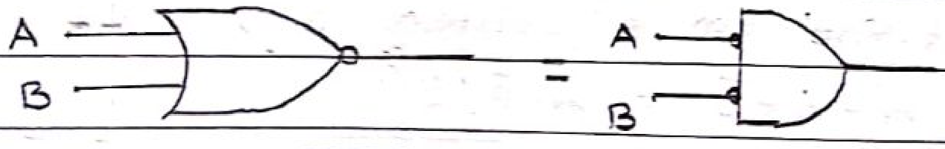
When three inputs are involved, De Morgan's second theorem is written as

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

If four inputs are used

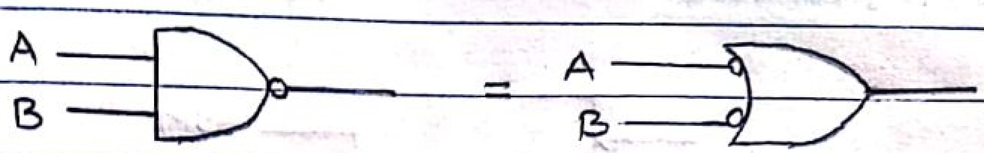
$\overline{ABCD} = \bar{A} + \bar{B} + \bar{C} + \bar{D}$ and so on.

The graphic summary of De morgan's first and second theorem is as follows.



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

and

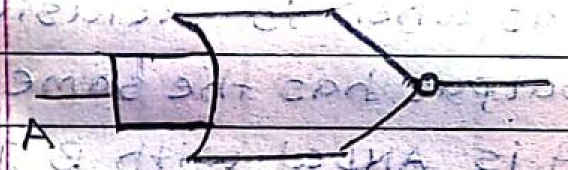


$\overline{AB} = \overline{A} + \overline{B}$

* NOR Gate as Universal Gate

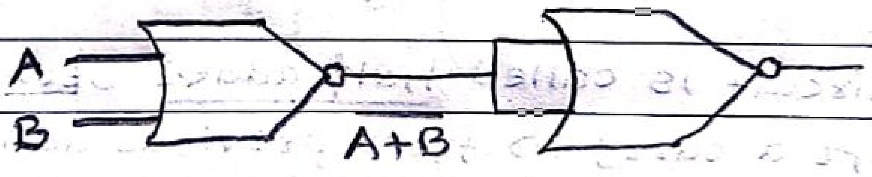
The NOR gate can be used to realize the basic logic functions OR, AND and NOT. That is why it is often referred to as universal gate.

<I> NOT gate using NOR gate



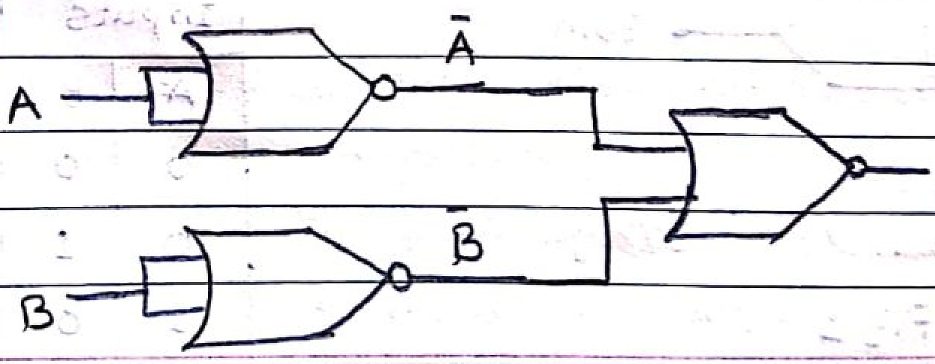
$\overline{A+A} = \overline{A}$

<II> OR gate using NOR gate



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

<III> AND gate using NOR gate



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

* Half Adder

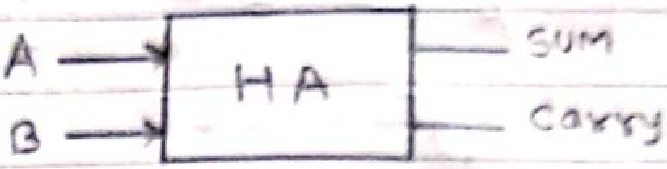


Fig.1

Half adder add 2 binary digits at a time and produce a 2-bit data i.e sum and carry according to binary addition rules. Fig.1. Shows symbol of half adder, consisting of two inputs A and B. The two outputs are sum and carry. Truth table of half adder shows that the sum output has the same logic pattern as when is exclusive OR with B. The carry output has the same logic pattern as when A is ANDed with B. Therefore a half adder can be formed from a combination of one XOR gate and one AND gate as shown in Fig.2

The circuit is called half adder because it can not accept a carry in from previous addition. For that we need a 3-input adder called full adder.

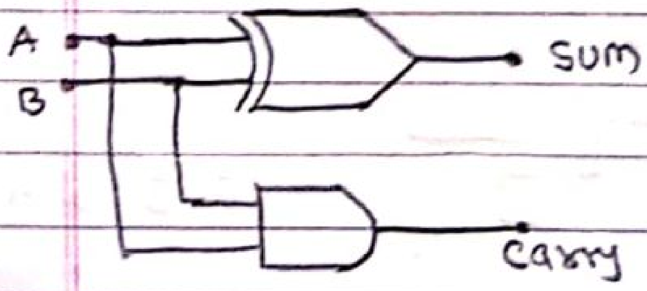


Fig.2

Truth Table

Inputs		outputs	
A	B	sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

* Full Adder

The Full Adder can be constructed from two half adders and an OR gate as shown in Fig.1.

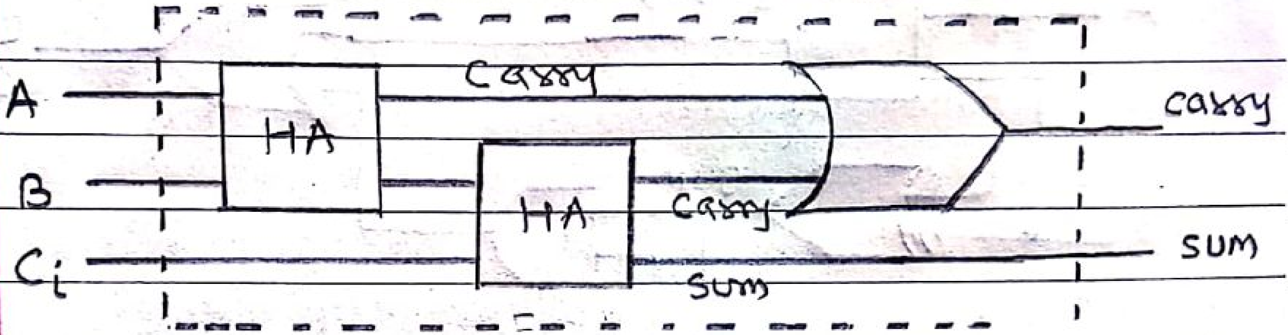


Fig 1

Truth Table

Inputs			outputs	
A	B	C	sum	carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

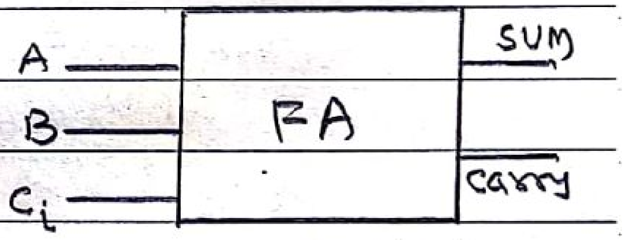


Fig. 2 Symbol

The truth table gives all possible inputs and output relationship for the full adder. A and B are the inputs from the respective digits of the registers to be added and C_i is the input from carry genezated by the previous stage

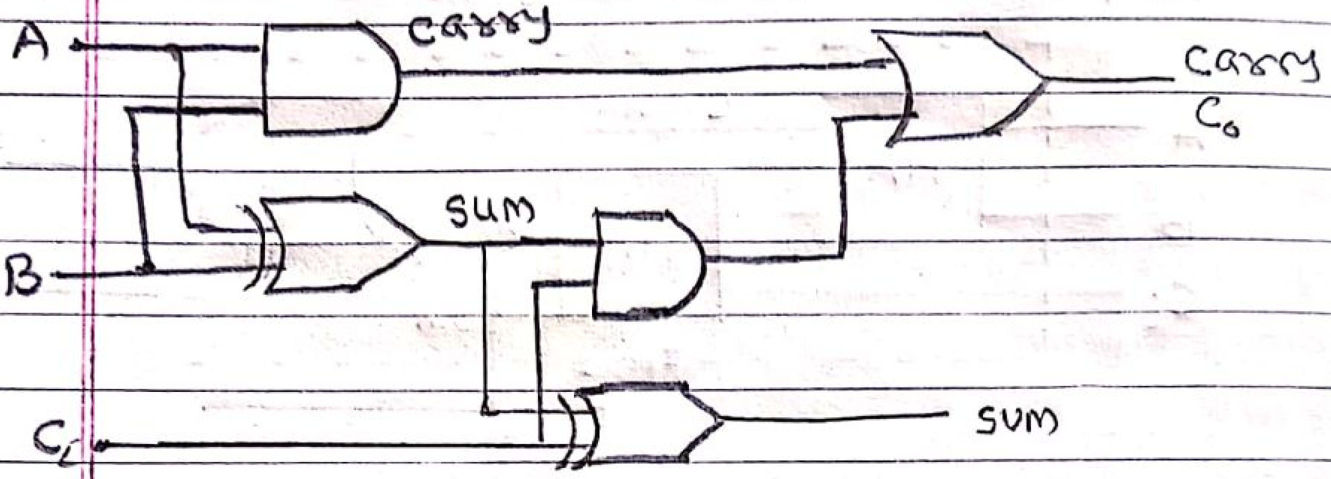


Fig 3