

# Digital Logic Gates:-

→ The logic gates accepts one (or) more inputs and produces only one output.

→ Basically logic gates are classified into 3 types.

1. Basic gates

2. Universal gates

3. Special gates.

## 1. Basic gates:-

The basic gates are

i) AND gate

ii) OR gate

iii) NOT gate

## i) AND gate:-

The AND gate accepts two (or) more inputs and produces only one output.

\* The output of an AND gate is product of the inputs.

Symbol :- Let us consider 2- i/p AND gate.



- The 2 i/p's AND gate produces the only one o/p i.e  $Y$ . ( $Y$  = product of input's).
- The 2 inputs are A and B.

Output  $\boxed{Y = A \cdot B}$

Truth Table :- 2 i/p AND gate provides  $2^2 = 4$  combinations i.e (0 to 3)

| Inputs | Output |                 |
|--------|--------|-----------------|
| A      | B      | $Y = A \cdot B$ |
| 0      | 0      | 0               |
| 0      | 1      | 0               |
| 1      | 0      | 0               |
| 1      | 1      | 1               |

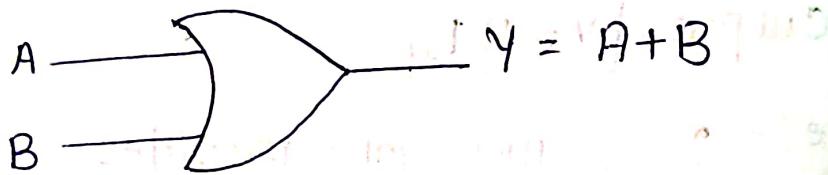
- From the above truth table, any one i/p is "0 (zero)", then the output is zero(0).
- If both i/p's are '1' then the o/p is '1'.
  - \* where as '1' is called as "High" and '0' is called as "Low".
  - \* Any one input is low, then the output is low, if both i/p's high then the o/p is high.

ii) OR gate:-

The OR gate accepts two or more inputs, and produces only one output.

\* The output of OR gate is sum of the inputs.

symbol:- Let us consider a 2 input OR gate.



- From the above symbol, A and B are the 2 inputs. Y is the output.
- The output  $Y = \text{sum of inputs}$ .

$$Y = A + B$$

Truth Table:- a 2 input OR gate provides  $2^2 = 4$  combinations.

Inputs      Output

$$Y = A + B$$

0 0 → 0

1 0 → 1

2 1 → 1

3 1 → 1

→ From the truth table any one ilp is '1' then the olp is '1'. If both ilp's are zero (0) then the olp is '0'.

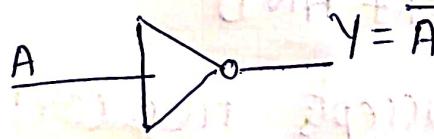
→ If any one input is high then the olp is high, if both ilp's are low then the olp is low.

iii) NOT gate:-

The NOT gate accepts only one input and produces only one output.

\* the output of NOT gate is complement of input.

Symbol :-



→ From the above symbol 'A' is the input and 'Y' is the output.

→ The olp  $Y = \text{complement of } A$

$$Y = \bar{A}$$

Truth Table :-

| Input<br>A | Output<br>$Y = \bar{A}$ |
|------------|-------------------------|
| 0          | 1                       |
| 1          | 0                       |

- If the input is '0' then the output is High
- if the input is '1' then the output is Low
- Input is High - o/p is Low
- Input is Low - o/p is High.

## 2. Universal Gates:-

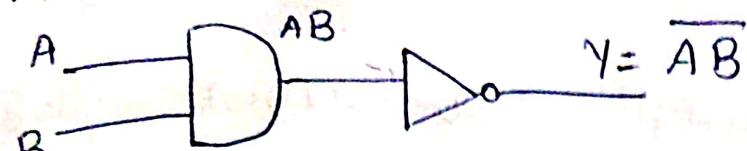
By using universal gates, implement + any logic gate. The universal gates are

- i) NAND gate
- ii) NOR gate.

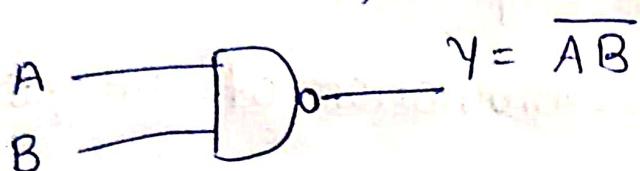
### i) NAND gate:- NOT + AND

- The NAND gate accepts two (or) more inputs and produces only one output.
- NAND is the combination of AND gate and NOT gate.
- The output of NAND gate is complement of AND gate output.
- The o/p of NAND gate is complement of product of inputs.

Symbol :- Let us consider a 2 input NAND gate.



(or)



Truth Table :-  $2^2 = 4$  combinations.

| Inputs |   | 0: | 0lp                 |
|--------|---|----|---------------------|
| A      | B | AB | $Y = \overline{AB}$ |
| 0      | 0 | 0  | 1                   |
| 0      | 1 | 0  | 1                   |
| 1      | 0 | 0  | 1                   |
| 1      | 1 | 1  | 0                   |

→ From the truth table any one input is low (0), then the output is high (1). If both inputs are high (1), then the output is low (0).

ii) NOR gate :-

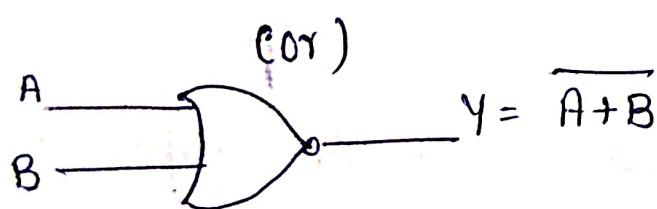
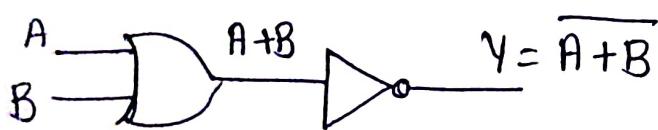
The NOR gate accepts two (or) more inputs and produces only one output.

→ The NOR gate is combination of OR gate and NOT gate.

→ It is the complement of OR gate.

→ The output of NOR gate is complement of OR'Sum of inputs.

Symbol:- Let us consider 2 i/p NOR gate.



Truth Table:-  $2^2 = 4$  combinations.

| Inputs |   | $A+B$ | output<br>$Y = \overline{A+B}$ |
|--------|---|-------|--------------------------------|
| A      | B |       |                                |
| 0      | 0 | 0     | 1                              |
| 0      | 1 | 1     | 0                              |
| 1      | 0 | 1     | 0                              |
| 1      | 1 | 1     | 0                              |

\* If both i/p's are low(0) - o/p is high(1).  
if any i/p is high(1) - o/p is low(0).

3. Special gates:— There are two special gates

i) XOR [Exclusive OR] gate (or) Ex-OR

ii) XNOR [Exclusive NOR] gate

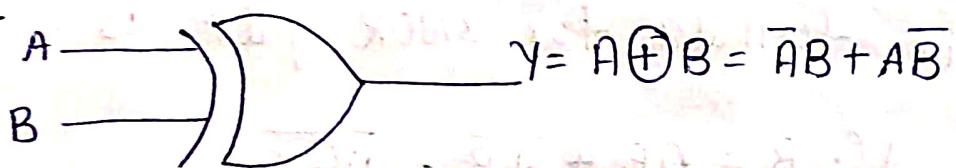
i) XOR (Exclusive OR) gate (or) Ex-OR:-

The XOR gate accepts two (or) more inputs and produces only one output.

The formula for XOR gate is

$$A \oplus B = \bar{A}B + A\bar{B} \rightarrow \text{OP of XOR}$$

Symbol:-



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

Truth Table:-  $2^2 = 4$  combinations

Inputs

Output

$$\begin{array}{ccccccccc} A & B & \bar{A} & \bar{A}B & \bar{B} & A\bar{B} & Y = \bar{A}B + A\bar{B} \\ \hline 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{array}$$

$$\begin{array}{ccccccccc} 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}$$

$$\begin{array}{ccccccccc} 1 & 0 & 0 & 0 & 1 & 1 & 1 \end{array}$$

$$\begin{array}{ccccccccc} 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{array}$$

\* If the both inputs are same (either 0 or 1) then the output is low (0).

\* If the both inputs are different then the output is High (1).

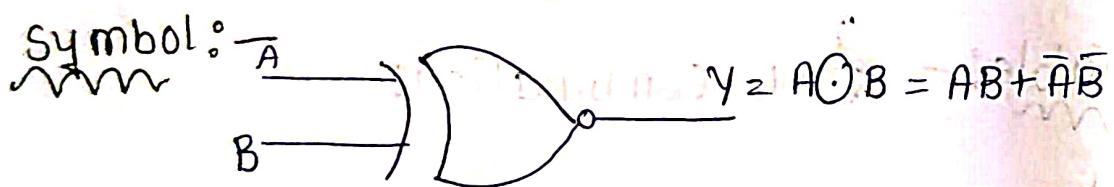
ii) XNOR (Exclusive NOR) gate (or) Ex-NOR:

The XNOR gate accepts two (or) more inputs and produces only one output.

\* The output of XNOR gate is the complement of XOR gate.

The formula for XNOR gate is

$$A \odot B = AB + \bar{A}\bar{B} = \overline{A \oplus B}$$



Truth Table:-

Inputs

| A | B | AB | $\bar{A}$ | $\bar{B}$ | $Y = AB + \bar{A}\bar{B}$ | Output |
|---|---|----|-----------|-----------|---------------------------|--------|
| 0 | 0 | 0  | 1         | 1         | 1                         | 1      |
| 0 | 1 | 0  | 1         | 0         | 0                         | 0      |
| 1 | 0 | 0  | 0         | 1         | 0                         | 0      |
| 1 | 1 | 1  | 0         | 0         | 0                         | 1      |

\* If the both inputs are same (either 0 or 1), then the output is High (1).  
 If the both inputs are different - then the output is Low (0).

### Multilevel NAND/NOR Realizations:-

Implementation of all logic gates using NAND and NOR gates:

→ The Universal gates (NAND & NOR) gates are used to implement the all logic gates. i.e AND gate, OR gate, NOT gate, XOR Ex-OR gate and XNOR gate.

|               | NAND | NOR |
|---------------|------|-----|
| Basic gates   | NOT  | 1   |
|               | AND  | 2   |
|               | OR   | 3   |
| Special gates | XOR  | 4   |
|               | XNOR | 5   |

- \* To implement the NOT gate, we require one NAND gate.
  - \* To implement the AND gate, we require two NAND gates.
  - \* To implement the OR gate, we require three NAND gates.
  - \* To implement the XOR gate, we require four NAND gates.
  - \* To implement the XNOR gate, we require five NAND gates.
- 
- \* To implement the NOT gate, we require one NOR gate.
  - \* To implement the AND gate, we require three NOR gates.
  - \* To implement the OR gate, we require two NOR gates.
  - \* To implement the XOR gate, we require five NOR gates.
  - \* To implement the XNOR gate, we require four NOR gates.

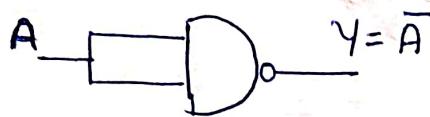
## NAND gate

### 1. NOT gate:

→ We know that, NAND gate accepts two inputs but NOT gate accepts only one input.

So we have to join the two inputs.

Symbol:-



$$y = \bar{A}$$

Basically

1ip's olp

Inputs

$$0 \quad 0$$

Output

$$1 \quad 1$$

A    B    Y

$$0 \quad 0 \quad 1$$

|   |   |   |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 0 | 1 |

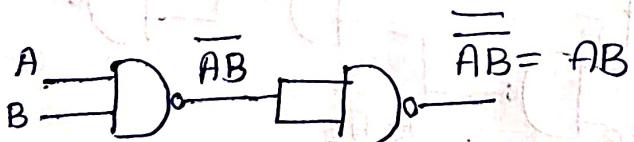
$$1 \quad 1 \quad 0$$

→ If the inputs are '0' then

the olp is '1' i.e complement of 0 = 1.

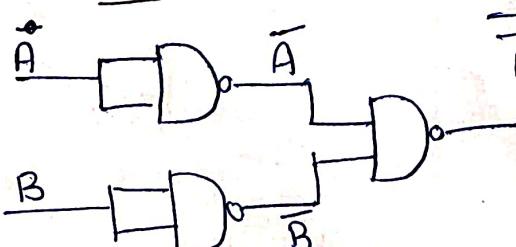
→ If the inputs are '1' then the olp is '0' i.e complement of 1 = 0.

### 2. AND gate:-



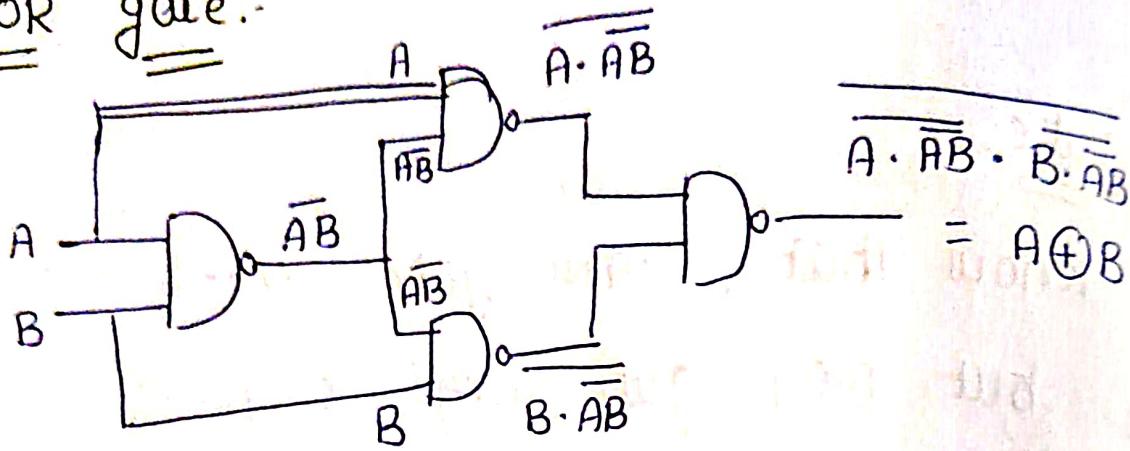
$$\overline{\overline{AB}} = AB$$

### 3. OR gate:-



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

#### 4. XOR gate:-



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

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

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

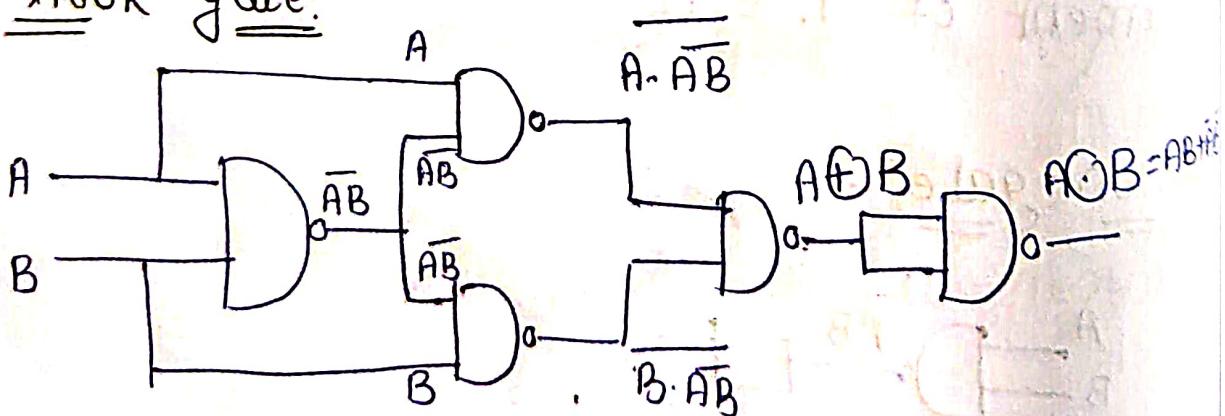
$$= A(\bar{A}B + A\bar{B}) + B(\bar{A}B + A\bar{B}) \quad (\because A \cdot \bar{A}B = A(\bar{A}B + A\bar{B}))$$

$$= \cancel{\frac{A \cdot \bar{A}B}{A}} + \cancel{\frac{A \cdot \bar{A}B}{B}} + \cancel{\frac{\bar{A} \cdot B \cdot B}{B}} + \cancel{\frac{A \cdot B \cdot \bar{B}}{B}}$$

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

$$= A \oplus B$$

#### 5. XNOR gate:



NOR gate

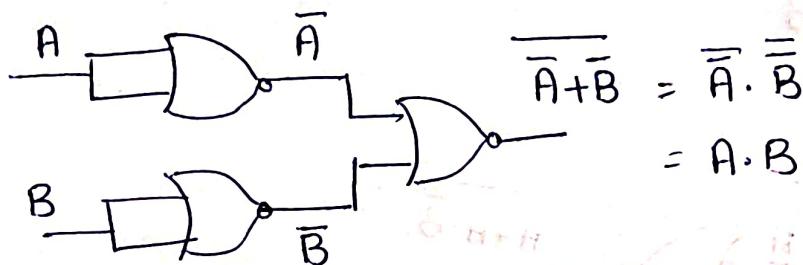
1. NOT gate:



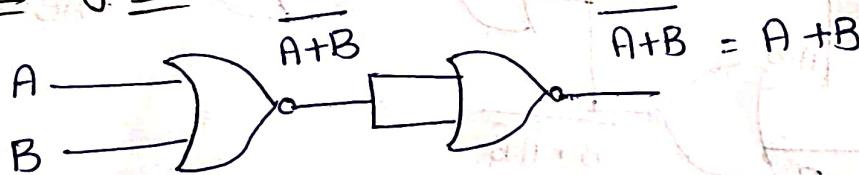
A B NOR

|   |   |   |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |

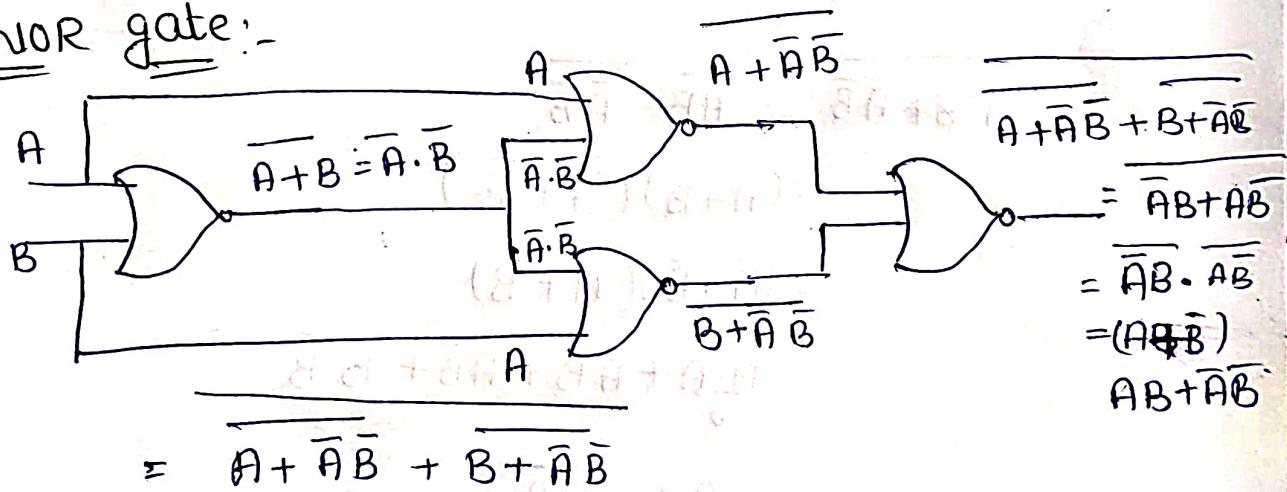
2. AND gate:



3. OR gate:



4. xNOR gate:-



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

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

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

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

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

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

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

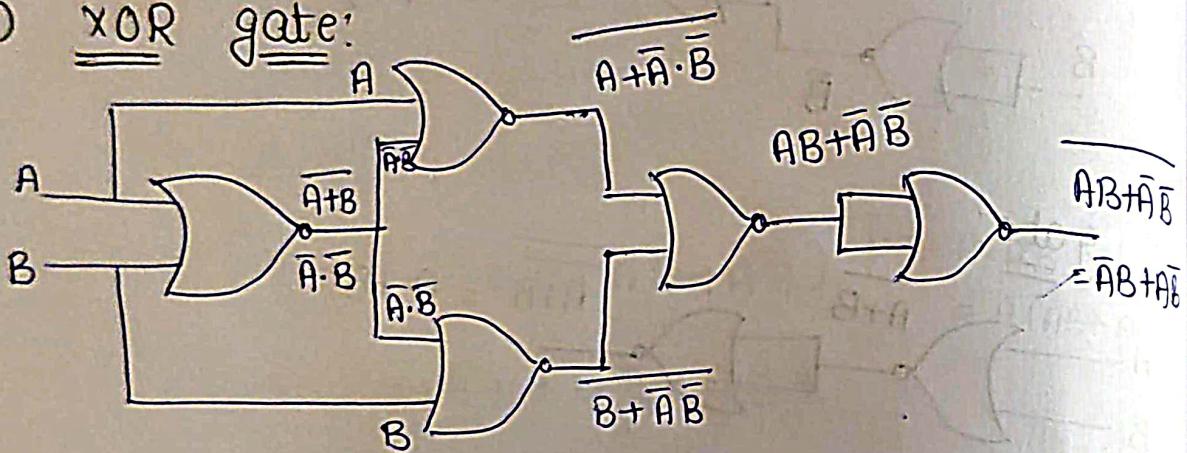
$$= \bar{B} \cdot (A + B)$$

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

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

$$\begin{aligned}
 & \overline{\bar{A}B + A\bar{B}} \quad [\because \text{from DeMorgan's theorem} \\
 & \quad \overline{A+B} = \bar{A} \cdot \bar{B}] \\
 & = \overline{\bar{A}B} \cdot \overline{A\bar{B}} \\
 & = (\bar{A}+\bar{B})(\bar{A}+B) \\
 & = \underbrace{\bar{A} \cdot \bar{A}}_0 + AB + \overline{A}\bar{B} + \underbrace{\bar{B}\bar{B}}_0 \\
 & = AB + \bar{A}\bar{B}
 \end{aligned}$$

5) XOR gate:



$$\begin{aligned}
 \overline{AB + A'\bar{B}} &= \overline{AB} \cdot \overline{A'\bar{B}} \\
 &= (\bar{A} + \bar{B})(\bar{\bar{A}} + \bar{\bar{B}}) \\
 &= (\bar{A} + \bar{B})(A + B) \\
 &= \underbrace{\bar{A} \cdot \bar{A}}_0 + \bar{A}B + A\bar{B} + \underbrace{B \cdot \bar{B}}_0 \\
 &= \bar{A}B + A\bar{B}
 \end{aligned}$$

\* The NAND and NOR gates are used to implement the all other gates. But we prefer only NAND gate realization.

Because i) The time is very less to implement  
the other gates.

ii) Cost is very less