# CS & IT

## AND, OR, NAND, NOR GATE

**Logic Gate**

**Lecture No. 2**

By- CHANDAN SIR


01 AND , OR GATE

01 NAND GATE

02 NOR GATE

03 Discussion

# Guidelines to Attend Live Class

- ✓ Attend the class with positive attitude.
- ✓ Punctuality is necessary.
- ✓ Follow the day-wise study plan.
- ✓ Attempt DPP daily as per the schedule.
- ✓ Hold chat while attending the class. We will allow you to ask and put your questions in the comment box.

## AND

$0 \cdot 0 = 0$

$0 \cdot 1 = 0$

$1 \cdot 0 = 0$

$1 \cdot 1 = 1$

$\boxed{1 \cdot A = A}$

$\boxed{A \cdot A = A}$

$0 \cdot A = 0$ $\quad A \cdot 0 = 0$

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

## OR

$0 + 0 = 0$

$0 + 1 = 1$

$1 + 0 = 1$

$1 + 1 = 1$

$\boxed{A + A = A}$

$\boxed{A + 1 = 1}$

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

$A + 0 = A$

$A \cdot A = A$

$A + A = A$

1 + anything = 1

$1 + A = 1$

$1 + AB = 1$

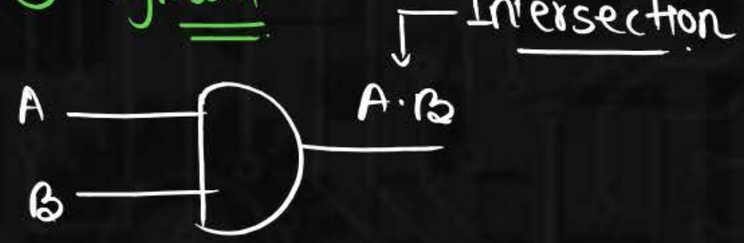
$1 + \bar{A}BCD = 1$

# AND GATE

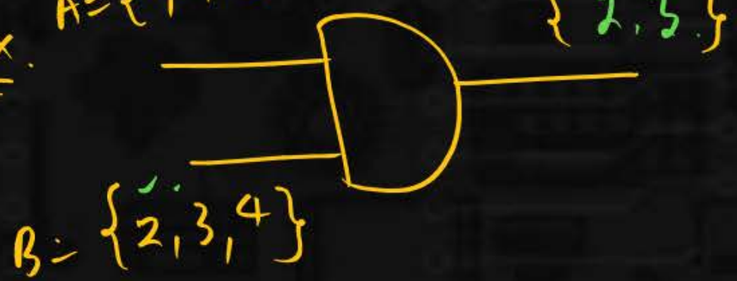
"n" → no. of inputs.

Total no. of combination = $2^n$

① Symbol




Ex. $A = \{1, 2, 3\}$




② Truth table.

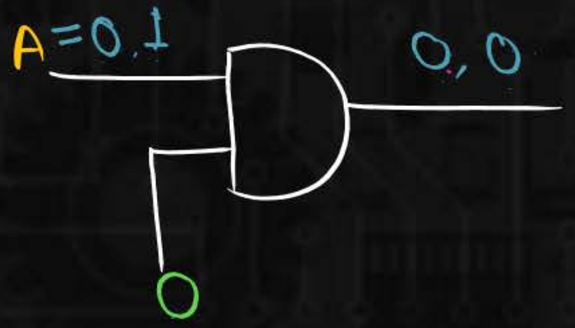
→ Relation between inputs & outputs.

| A | B | Y = A·B |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

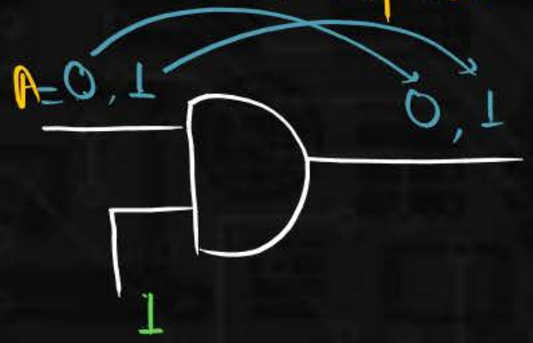
# AND GATE

floating terminal
Strobe
control input

③ Enable/Disabled :→




Control '0' Disabled




Control '1' Enabled

④ Commutative Law.

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

Associative Law :→

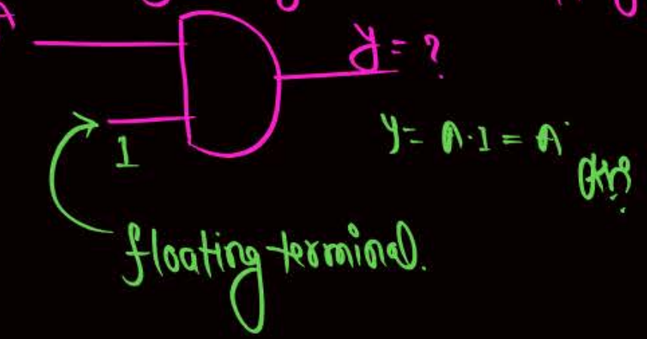
$$(A \cdot B) \cdot C = A \cdot (B \cdot C)$$

NOTE.

① TTL (Transistor - transistor Logic)

↳ floating terminal always acts as high.

Ex. Logic is designed by TTL, then o/p y will be—

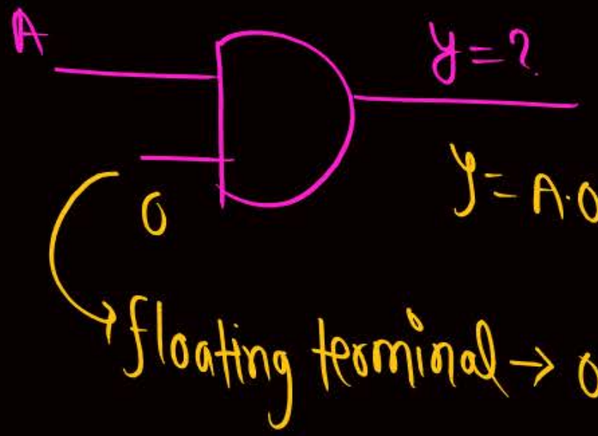



$y = A \cdot 1 = A$ Ans

② ECL (Emitter coupled Logic)

↳ floating terminal always acts as a Low.

Ex. Logic is designed by ECL, then output 'y' will be—




$y = A \cdot 0 = 0$ Ans

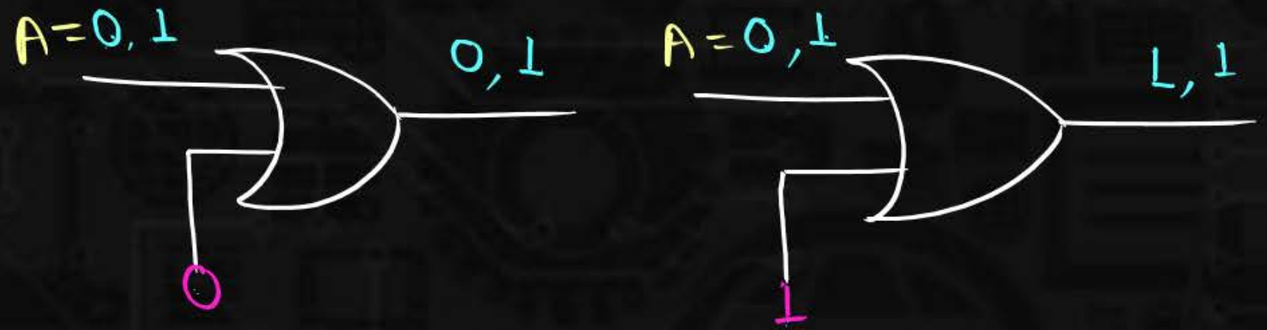
# OR GATE

## ① Symbol :→


## ② Truth Table.

| A | B | Y=A+B |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

## ③ Enable / Disable.




Control '0' Enable.

Control '1' Disabled

## ④ Commutative Law.

$A+B=B+A$ ✓

## Associative Law. ✓

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

# OR GATE

₹5 = NOTE

AND, OR

→ Commutative
Associative
} ✓

# AND , OR GATE

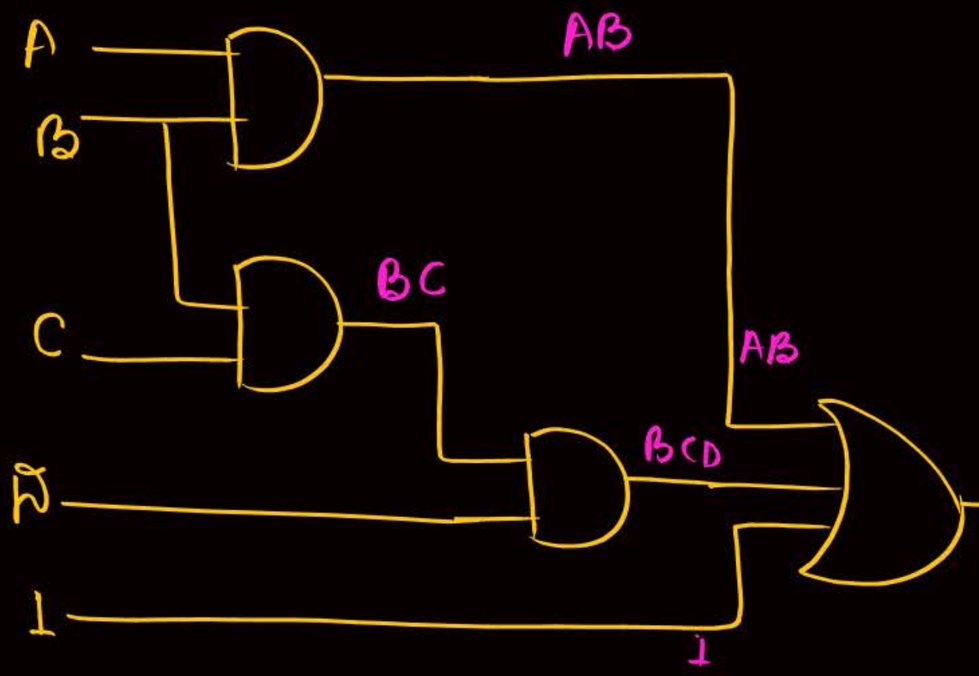
## Important Points

1) Whenever logic are designed by TTL (Transistor- Transistor logic) then floating terminal always works as a high.

2) Whenever logic are designed by ECL (Emitter coupled logic ) then floating terminal always works as a low.

MCQ

Q.




$y = ?$

$AB + BCD + 1$

$= 1$

$= AB$

(A) 0

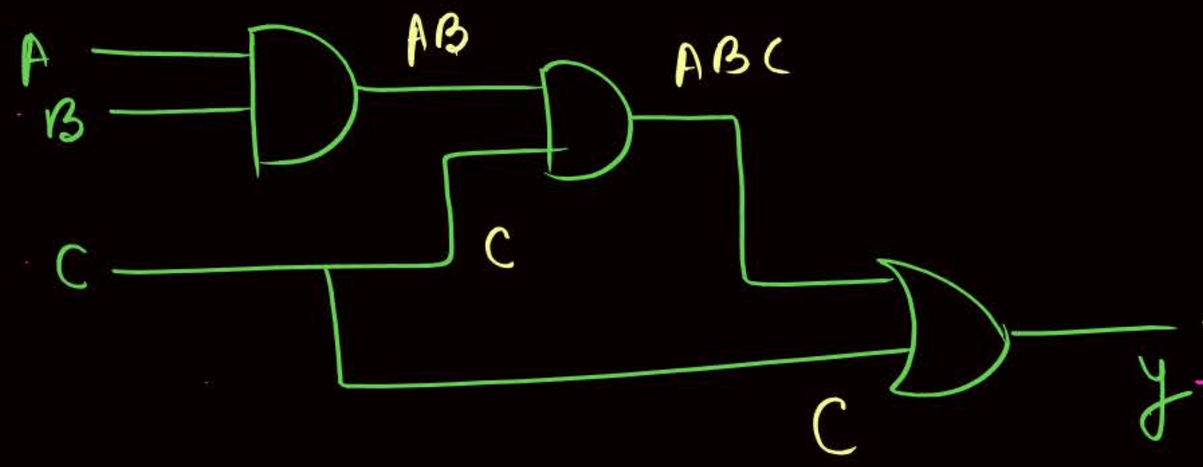
(B) 1

(C) AB

(D) Sir, Mujhe nahi ata hai, Mai Tare jameen par hu.

Q If the o/p is high then input will be—

(MSQ)




$y = ABC + C$

$= C[AB + 1]$

$= C \cdot 1$

$y = C$

$1 = C$ ✓

✗ (A) 0 0 0

(B) 0 0 1 ✓

✗ (C) 0 1 0

✗ (D) 1 1 0

(E) 1 1 1 ✓

(B)(E)

## D-Morgans Law.

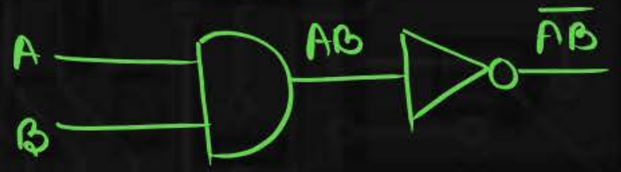
$$\overline{AB} = \bar{A} + \bar{B}$$

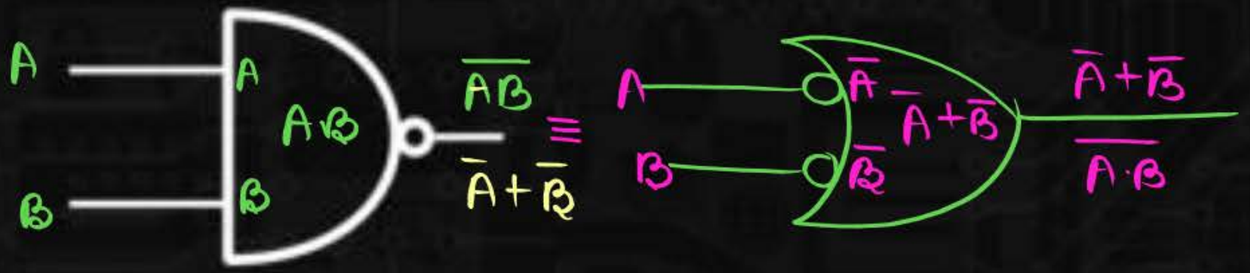
$$\overline{A+B} = \bar{A} \cdot \bar{B}$$

# NAND, NOR GATE

## NAND GATE

1) Symbol




Bubbled OR ≡ NAND

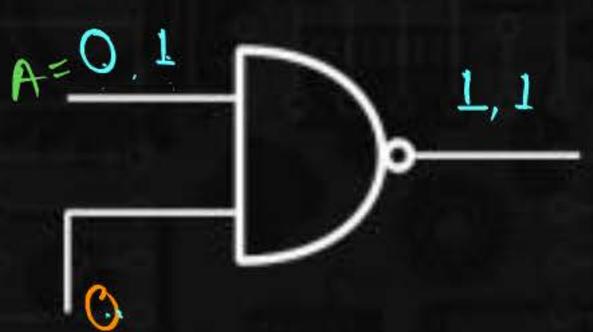
2) Truth Table

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

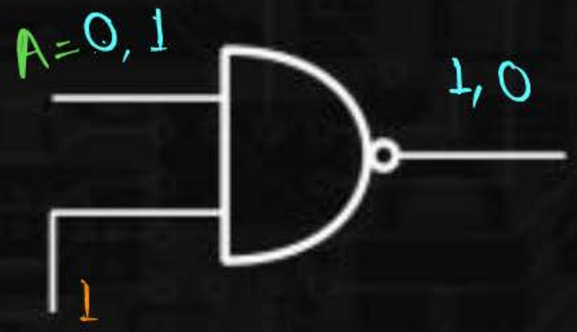
# NAND, NOR GATE

## NAND GATE

3) Enable/Disable




Control '0' Disabled.

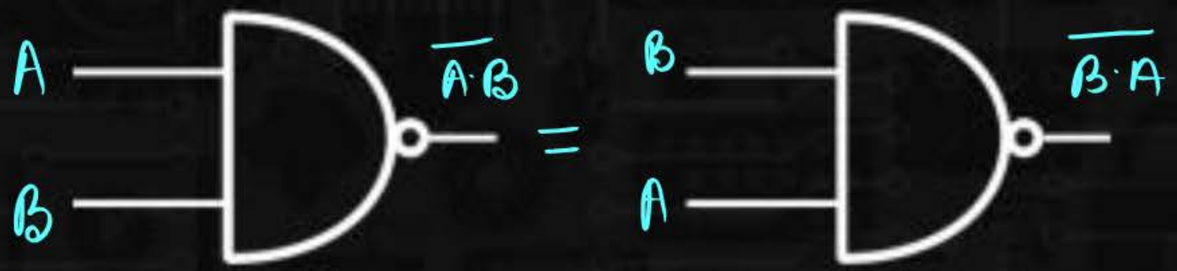



Control '1' Enabled

# NAND, NOR GATE

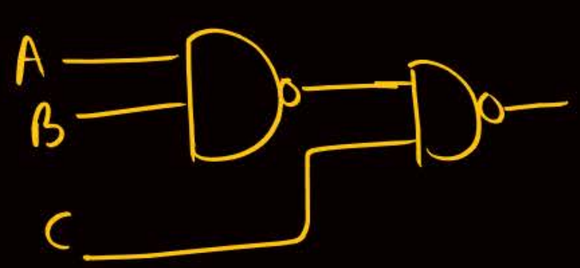
## NAND GATE

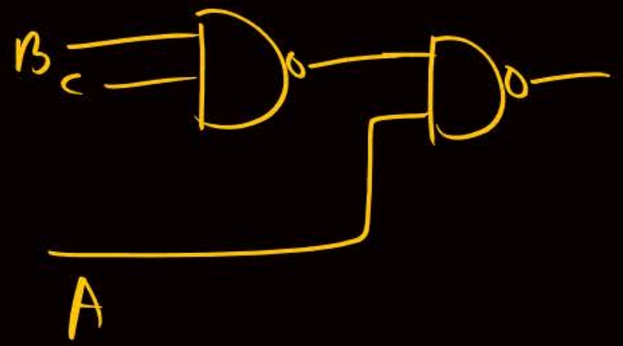
4) Commutative Law



## Associative Law: ✗

$$\overline{(\overline{A \cdot B}) \cdot C} \neq \overline{A \cdot (\overline{B \cdot C})}$$

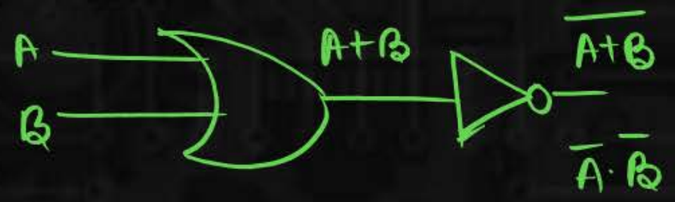


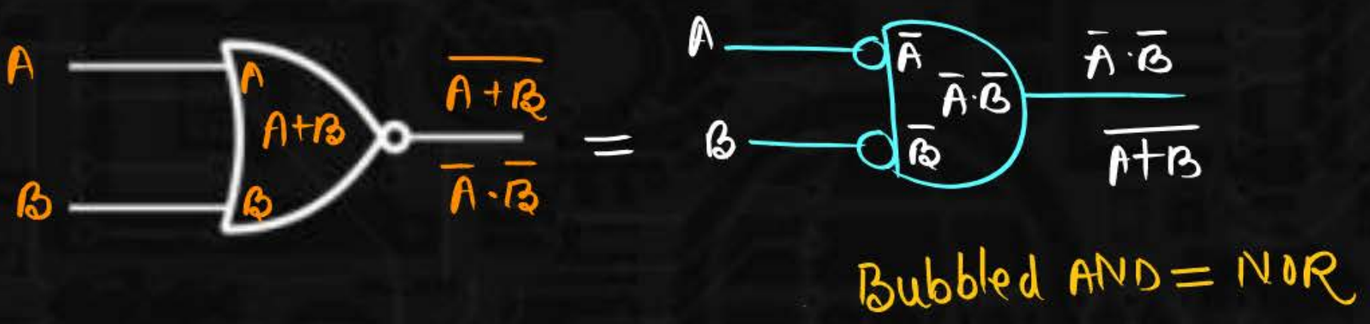



# OR GATE, NAND, NOR GATE

## NOR GATE

1) Symbol




Bubbled AND = NOR

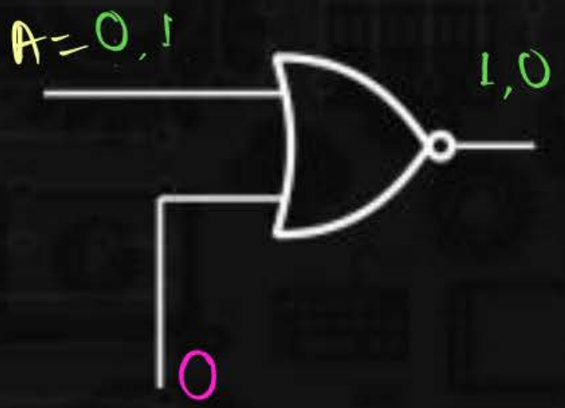
2) Truth Table

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

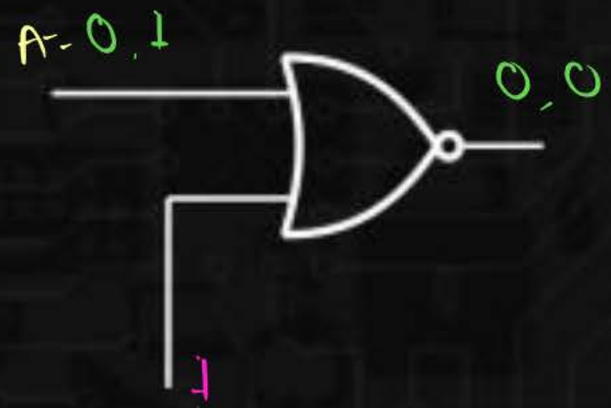
# NAND, NOR GATE

## NOR GATE

3) Enable/Disable




Control '0' Enabled

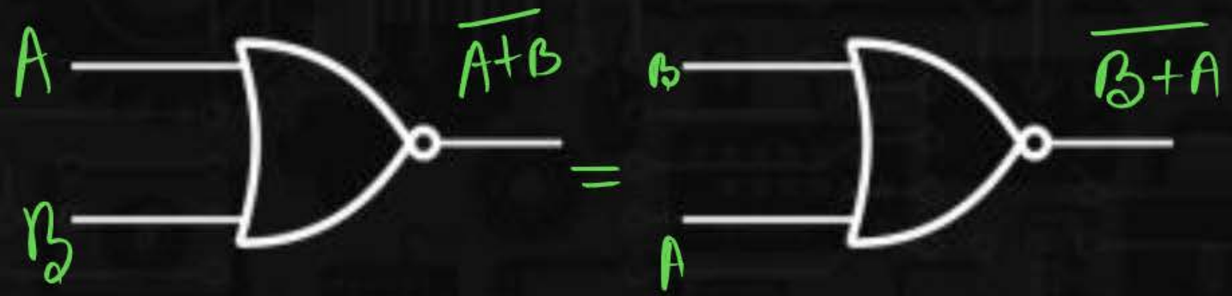



Control '1' Disabled

# NAND, NOR GATE

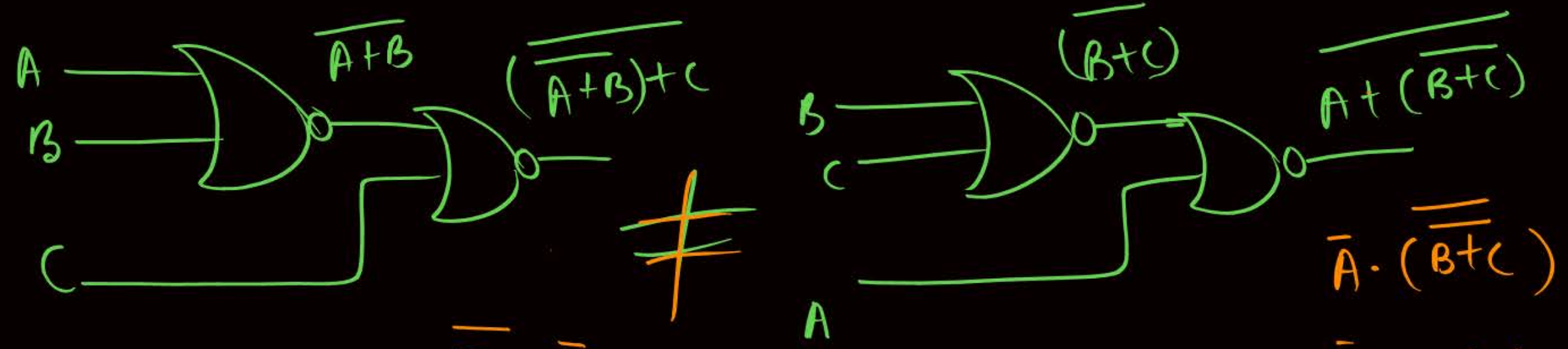
## NOR GATE

4) Commutative Law



## Associative Law:

↳ does not follow.




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

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

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

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

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

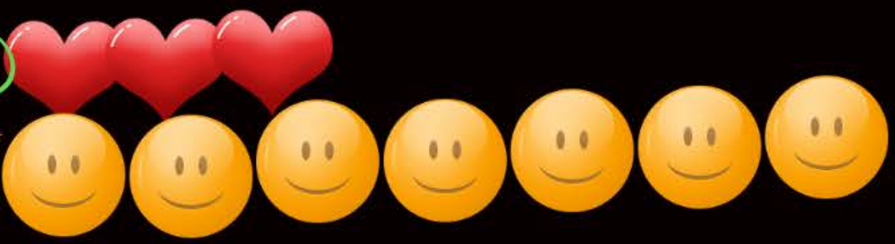
$\bar{A}B+\bar{A}C$

Z 10

## NOTE

① NAND, NOR follows commutative Law but does not follow associative Law.

② NAND, NOR are called universal logic



Q.1 Which of the following option is called universal logic?

(MCQ)

A. NAND

B. NOR

C. Both A & B

D. None

**Q.2** Which of the following option is called universal logic?

{ MCQ }

A. NAND

B. NOR → DCTL

Direct coupled transistor logic X

Current Hogging problem

C. AND

D. OR

**Q.3** Which of the following option(s) is/are called universal logic?

{MSQ}

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

B. $\overline{(A \cdot B)}$

C. $(A + \overline{B})$

D. $A \cdot \overline{B}$

universal!

{Sat, Sunday} {4:00PM - 6:00PM, 6:30PM - 8:30PM}

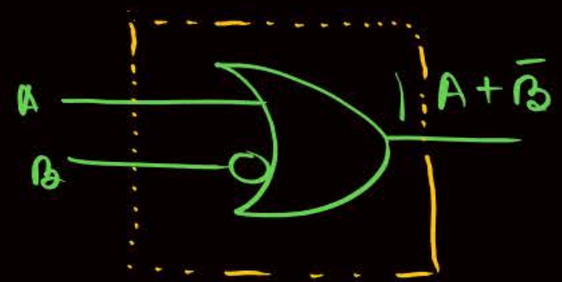
function → universal

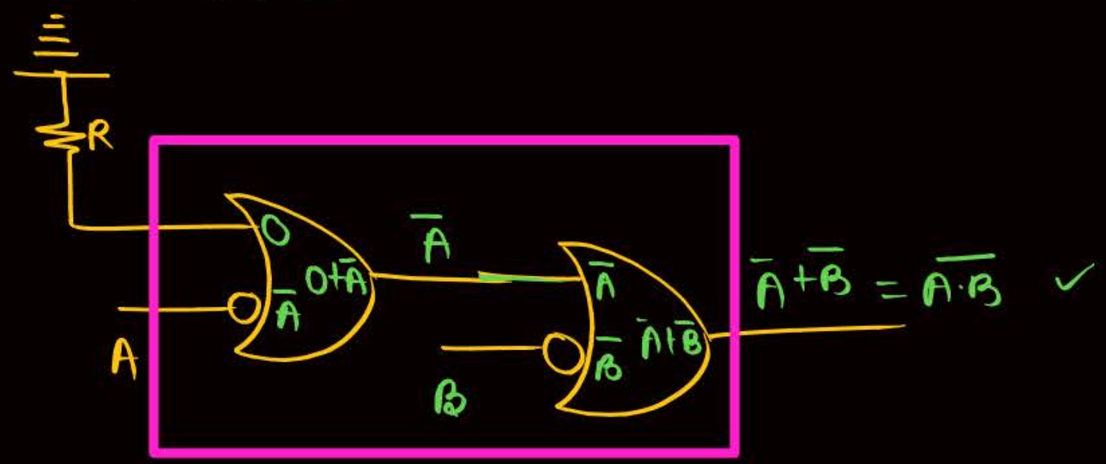
$\overline{A \cdot B}$ ✓

or

$\overline{A + B}$ ✓

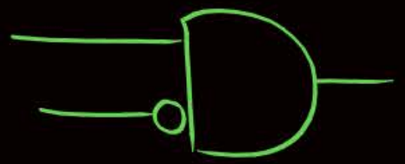
$A + \bar{B}$

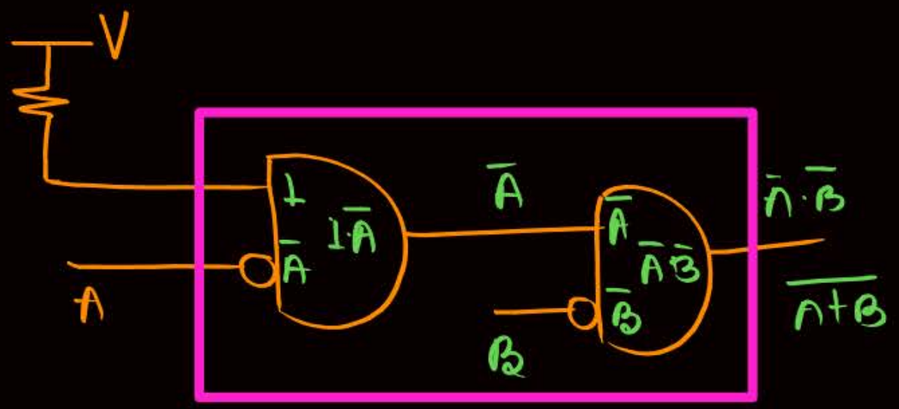



NAND

$A \cdot \bar{B}$






NOR

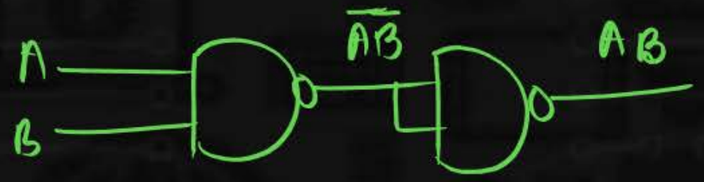
$$\left\{ \begin{array}{l} \overline{A \cdot B} \\ \overline{A+B} \\ \bar{A} \cdot B \\ A \cdot \bar{B} \\ \bar{A}+B \\ A+\bar{B} \\ \bar{A} \cdot \bar{B} \\ \bar{A}+\bar{B} \end{array} \right\} \text{ universal}$$

# NAND, NOR GATE

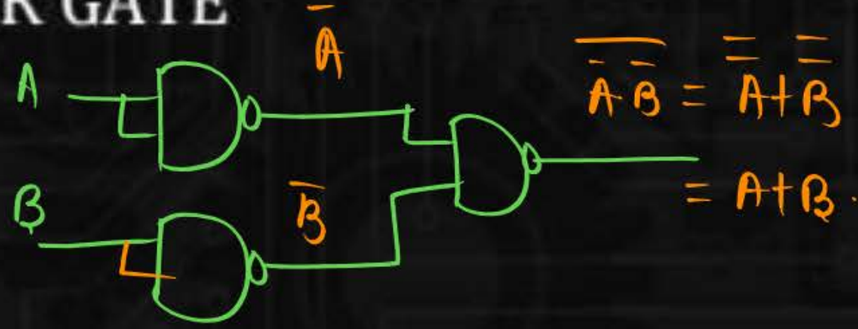
## NAND AS UNIVERSAL LOGIC

1) NOT GATE

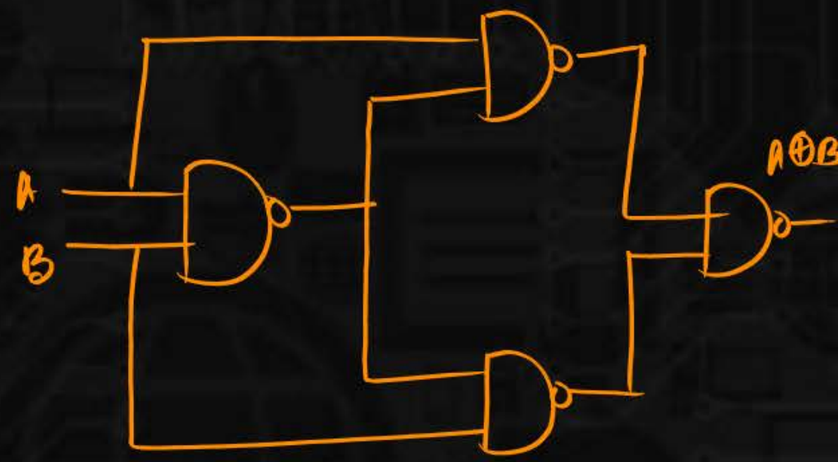

2) AND GATE



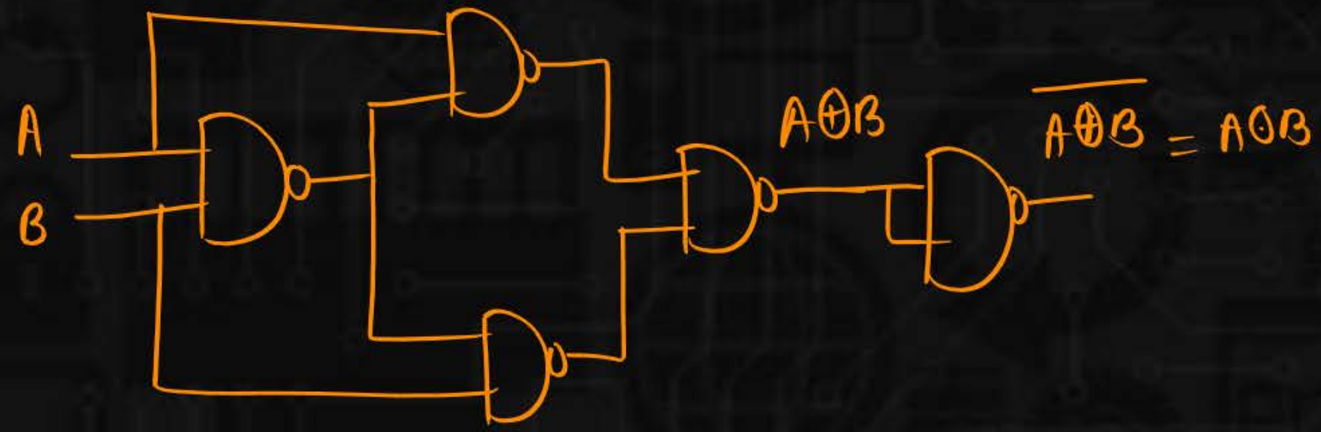

3) OR GATE




4) XOR GATE




5) XNOR GATE



# NAND, NOR GATE

## Alternate Symbol

**Q.4** Find the minimum number of two input NAND GATE required to implement the function-

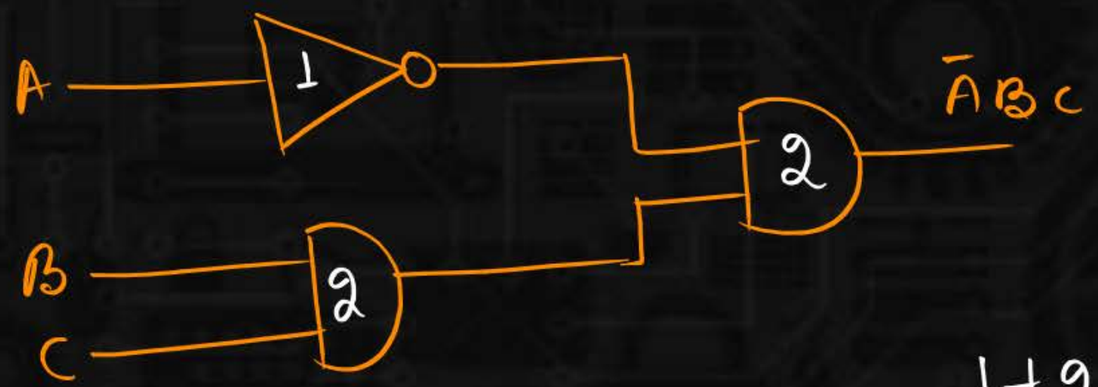
$f(A, B, C) = \overline{A}BC$

A. 3

B. 5

C. 6

D. 4




$1+2+2=5$.


THANK YOU GW
SOLDIERS !