

# Logic Gates

## Binary Logic

- Digital logic gates typically operate on binary logic.
  - Two distinct values in binary logic, denoted by 0 and 1.
- Why binary logic?
  - It is easy to design electronic circuits with two distinct states.
  - Examples:
    - An electronic switch is either open, or closed.
    - The voltage at a line is either low, or high.
    - Current in a line is either flowing, or not flowing.
    - Resistance value is either high, or low.
    - ... and so on

Logic gates (Basic)

NOT ✓

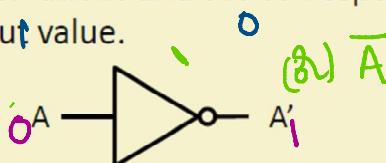
AND

OR

## Basic Logic Gates

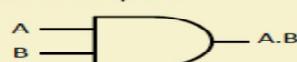
- NOT gate
  - A single input A, and an output A'.
- Behavior can be expressed by a truth table, which shows all possible input combinations and the corresponding output value.

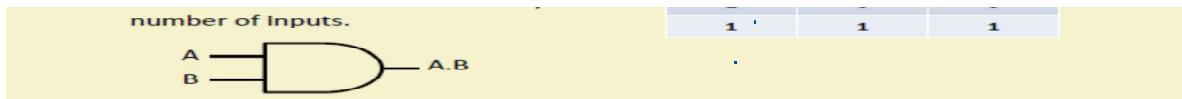
Truth Table	
A	B
0	1
1	0



- AND gate
  - For two inputs (say, A and B), the output will be 1 if both the inputs are at 1; will be 0 otherwise.
  - AND operation denoted as A.B
  - Definition can be extended to any number of inputs.

Truth Table		
A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1





3 i/p AND gate

$2^3$  possible combinations



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

- OR gate

- For two inputs (say, A and B), the output will be 1 if at least one of the inputs are at 1; will be 0 otherwise.
- OR operation denoted as  $A+B$
- Definition can be extended to any number of inputs.

Truth Table

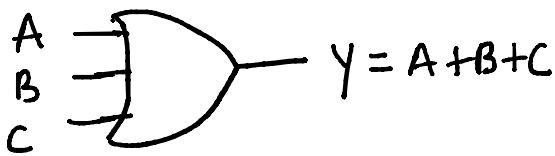
A	B	$A+B$
0	0	0
0	1	1
1	0	1
1	1	1



3 i/p OR gate

Truth table

## 3 i/p OR gate



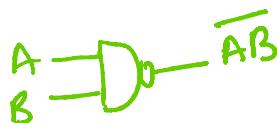
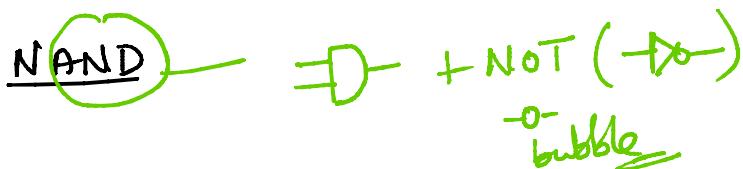
Truth table

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

## universal gates

NAND  $\rightarrow$  NOT + AND

NOR  $\rightarrow$  NOT + OR



- NAND gate

- For two inputs (say, A and B), the output will be 1 if at least one of the inputs are at 0; will be 0 otherwise.
- NAND operation denoted as  $(A \cdot B)'$
- Definition can be extended to any number of inputs.

Truth Table

A	B	$(A \cdot B)'$
0	0	1 ✓
0	1	1
1	0	1
1	1	0

$0 = A$     $0 = B$     $(A \cdot B)' = 1$   
 $AB = 0$ ,    $\overline{AB} = \overline{0} = 1$ .

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

$$\begin{array}{c} 0 \\ | \\ A \\ | \\ B \end{array} \text{---} \text{NAND gate} \text{---} \overline{AB} = 1$$

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

$$\overline{AB} = \overline{0} = 1$$

$$\begin{array}{c} 1 \\ | \\ A \\ | \\ 0 \\ B \end{array} \text{---} \text{NAND gate} \text{---} \overline{AB} = 1$$

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

$$\overline{AB} = \overline{0} = 1$$

$$\begin{array}{c} 1 \\ | \\ A \\ | \\ 1 \\ B \end{array} \text{---} \text{NAND gate} \text{---} \overline{AB} = 0$$

$$AB = 1 \cdot 1 = 1$$

$$\overline{AB} = \overline{1} = 0$$

### 3 i/p NAND gate

$$\begin{array}{c} A \\ B \\ C \end{array} \text{---} \text{NAND gate} \text{---} y = \overline{\overline{ABC}}$$

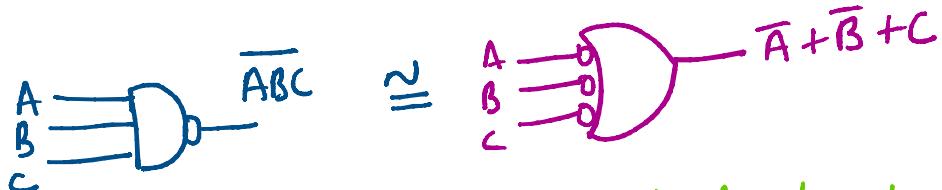
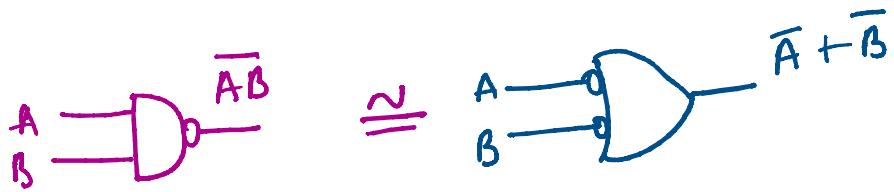
A	B	C	$y = \overline{\overline{ABC}}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Demorgan's theorem

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

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

$$-\overline{AB} - A \text{---} \overline{A} + \overline{B}$$



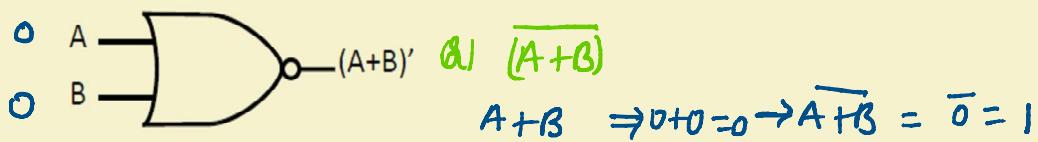
NAND gate is equivalent to bubble if OR gate

• NOR gate = OR + NOT

- For two inputs (say, A and B), the output will be 1 if both the inputs are at 0; will be 0 otherwise.
- NOR operation denoted as  $(A+B)'$
- Definition can be extended to any number of inputs.

Truth Table

A	B	$(A+B)'$
0	0	1 ✓
0	1	0 }
1	0	0 }
1	1	0 }



$$A+B = 0+1=1 ; \quad A+B = T=0$$



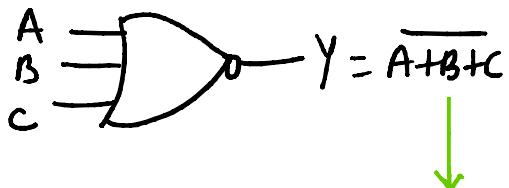
$$A+B = 1+0=1 ; \quad A+B = F=0$$



$$A \oplus B = \overline{A+B} = 0$$

$$A+B = 1+1 = 1 ; \quad \overline{A+B} = \overline{1} = 0$$

3 i/p NOR gate

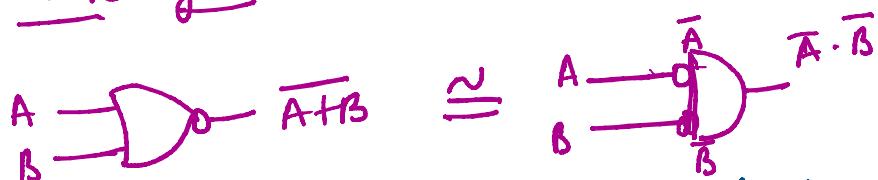


A B C	$y = \overline{A+B+C}$
0 0 0	1
0 0 1	0
0 1 0	0
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	0
1 1 1	0

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

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

NOR gate



NOR gate is equivalent to bubble i/p AND gate



- Exclusive OR (EXOR) gate
  - For two inputs (say, A and B), the output will be 1 if odd number of inputs are at 1; will be 0 otherwise.
  - EXOR operation denoted as  $A \oplus B$
  - Definition can be extended to any number of inputs.

Truth Table

A	B	$(A \oplus B)$
0	0	0
0	1	1
1	0	1
1	1	0

- EXOR operation denoted as  $A \oplus B$
- Definition can be extended to any number of inputs.

0	1	1
1	0	1
1	1	0



## 2 i/p EX-OR gate

$$\begin{array}{l}
 0 = A \\
 0 = B
 \end{array}
 \quad
 \begin{array}{l}
 \text{A} \oplus B \\
 = 0
 \end{array}
 \quad
 \begin{array}{l}
 = \bar{A}B + A\bar{B} \\
 = 1 \cdot 0 + 0 \cdot 1 \\
 = 0 + 0 \\
 = 0 \checkmark
 \end{array}$$

$$\begin{array}{l}
 0 = A \\
 1 = B
 \end{array}
 \quad
 \begin{array}{l}
 \text{A} \oplus B \\
 = 1 \checkmark
 \end{array}
 \quad
 \begin{array}{l}
 = \bar{A}B + A\bar{B} \\
 = 1 \cdot 1 + 0 \cdot 0 \\
 = 1 + 0 = 1
 \end{array}$$

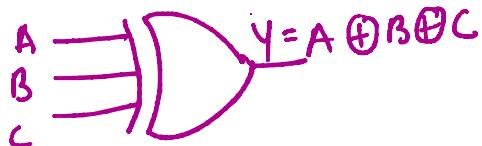
$$\begin{array}{l}
 1 = A \\
 0 = B
 \end{array}
 \quad
 \begin{array}{l}
 \text{A} \oplus B \\
 = 1
 \end{array}
 \quad
 \begin{array}{l}
 = \bar{A}B + A\bar{B} \\
 = 0 \cdot 0 + 1 \cdot 1 \Rightarrow 0 + 1 = 1
 \end{array}$$

$$\begin{array}{l}
 1 = A \\
 1 = B
 \end{array}
 \quad
 \begin{array}{l}
 \text{A} \oplus B \\
 = 0 \checkmark
 \end{array}
 \quad
 \begin{array}{l}
 = \bar{A}B + A\bar{B} \\
 = 0 \cdot 1 + 1 \cdot 0
 \end{array}$$

$$\begin{array}{l} \overline{A} = 0 \\ \overline{B} = 0 \end{array}$$

$$= 0$$

3 ifp EX-OR gate



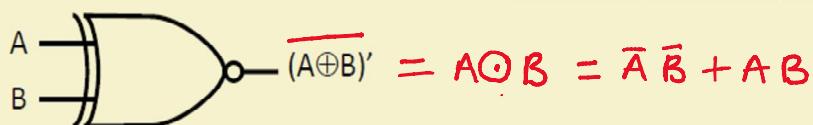
A	B	C	$Y = A \oplus B \oplus C$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

- Exclusive NOR (EXNOR) gate (Equivalence Gate)

- For two inputs (say, A and B), the output will be 1 if even number of inputs are at 1; will be 0 otherwise.
- EXNOR operation denoted as  $(A \oplus B)'$
- Definition can be extended to any number of inputs.

Truth Table  $A \oplus B = \overline{(A \oplus B)'} = \overline{(A \oplus B)'}$

A	B	$(A \oplus B)'$
0	0	1 ✓
0	1	0
1	0	0
1	1	1 ✓



$$\overline{(A \oplus B)} = A \odot B$$

$$\begin{aligned}
 0 &= A \\
 0 &= B
 \end{aligned}
 \quad
 \begin{aligned}
 Y &= 1 \\
 A \odot B &
 \end{aligned}
 \quad
 \begin{aligned}
 &= \overline{A} \overline{B} + A B \\
 &= 1 \cdot 1 + 0 \cdot 0 \\
 &= 1 + 0 = 1
 \end{aligned}$$

$$b = A \rightarrow \text{NOR} \rightarrow A \oplus B$$

$$l = B \rightarrow \text{NOR} \rightarrow A \oplus B = 0$$

$$\begin{aligned} & \bar{A} \bar{B} + AB \\ &= l \cdot 0 + 0 \cdot l \\ &= 0 + 0 \\ &= 0 \end{aligned}$$

$$l = A \rightarrow \text{NOR} \rightarrow A \oplus B = 0$$

$$\begin{aligned} & \bar{A} \bar{B} + AB \\ &= 0 \cdot l + l \cdot 0 \\ &= 0 + 0 \\ &= 0 \end{aligned}$$

$$l = A \rightarrow \text{NOR} \rightarrow A \oplus B = \overline{A \oplus B} = 1$$

$$\begin{aligned} & \bar{A} \bar{B} + AB \\ &= 0 \cdot 0 + 1 \cdot 1 \\ &= 0 + 1 \\ &= 1 \end{aligned}$$

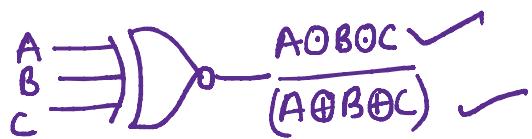
$$\underline{\text{EX-OR}} \rightarrow A \oplus B = \bar{A}B + A\bar{B}$$

$$\underline{\text{EX-NOR}} \rightarrow A \oplus B = \bar{A}\bar{B} + AB = \overline{(A \oplus B)}$$

$$\begin{aligned} \frac{A \oplus B}{\overline{A \oplus B}} &= \frac{\bar{A}B + A\bar{B}}{(\bar{A}B + A\bar{B})} \rightarrow \because (\overline{A+B}) = \bar{A} \cdot \bar{B} \\ &= (\bar{A}B) \cdot (\bar{A}\bar{B}) \rightarrow \because \overline{AB} = \bar{A} + \bar{B} \\ &= (\bar{A} + \bar{B}) \cdot (\bar{A} + \bar{B}) \quad \because \bar{\bar{A}} = A \\ &\quad \swarrow \pi \cdot \nearrow \bar{A} + B \quad \because \bar{\bar{B}} = B \\ &\therefore \pi \cdot \bar{A} + B = 0 \end{aligned}$$

$$\begin{aligned}
 &= (A + B) - \\
 &= (A + \bar{B}) \cdot (\bar{A} + B) \quad \therefore B = B \\
 &= (\underbrace{A \cdot \bar{A}}_0) + (\bar{A} \cdot \bar{B}) + AB + \underbrace{B \cdot \bar{B}}_0 \quad \therefore B \cdot \bar{B} = 0 \\
 &= \boxed{\bar{A} \bar{B} + AB} = A \oplus B = \overline{A \oplus B} = (A \oplus B)'
 \end{aligned}$$

3 inputs X-NOR



3 inputs  $\rightarrow$

A	B	C	$A \oplus B \oplus C$
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

# How to Construct these Gates?

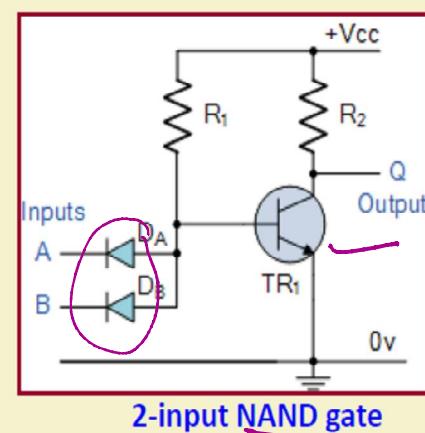
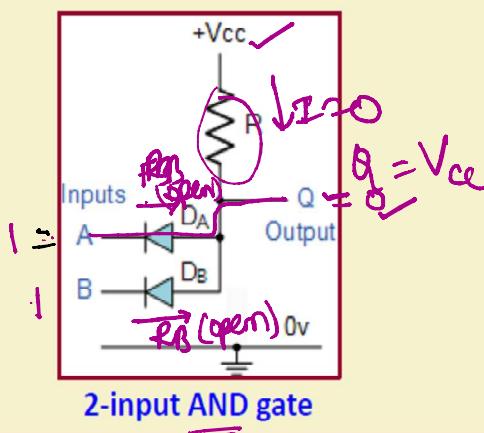
- Various logic families exist:
  - Diode transistor logic (DTL)
  - Transistor transistor logic (TTL)
  - Emitter Coupled Logic (ECL)
  - Complementary Metal Oxide Semiconductor (CMOS) Logic
- CMOS is almost universally used today.
- Some emerging technologies also exist:
  - All-optical implementation of logic
  - Memristor based logic
  - Quantum dot logic, and many more

# How to Construct Logic Gates?

- Various logic families exist:
  - Diode transistor logic (DTL)
  - Transistor transistor logic (TTL)
  - Emitter Coupled Logic (ECL)
  - Complementary Metal Oxide Semiconductor (CMOS) Logic
- CMOS is almost universally used today.

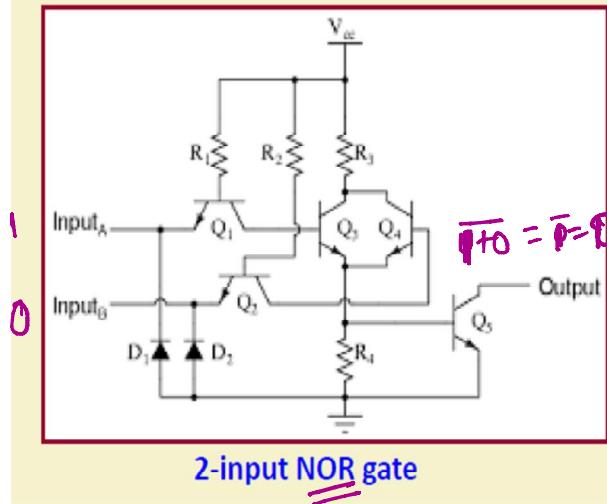
## Diode Transistor Logic

- Uses semiconductor diodes and bipolar transistors, along with resistances.

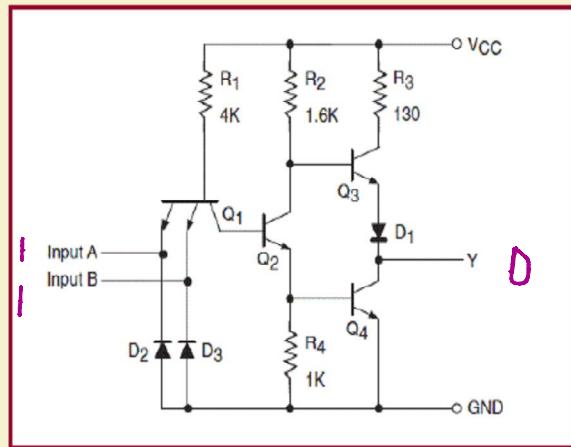


A	B	AB
0	0	0 ✓
0	1	0 ✓
1	0	0 ✓
1	1	1 ✓

# Transistor Transistor Logic



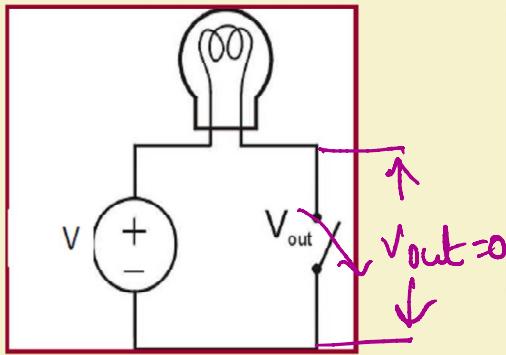
2-input NOR gate



2-input NAND gate

## Basic Concepts of Switch Based Circuits

- They rely on the operation of tiny switches, which can be in one of two states.
  - open or closed, ON or OFF, voltage or no voltage, etc.



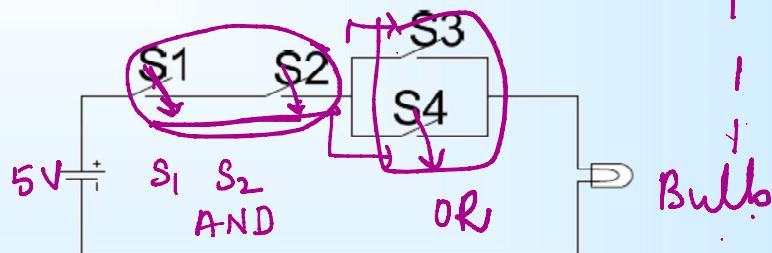
*Switch open:*

- No current flows.
- Light is OFF.

*Switch closed:*

- Current flows.
- Light is ON.

A control circuit for an electric bulb



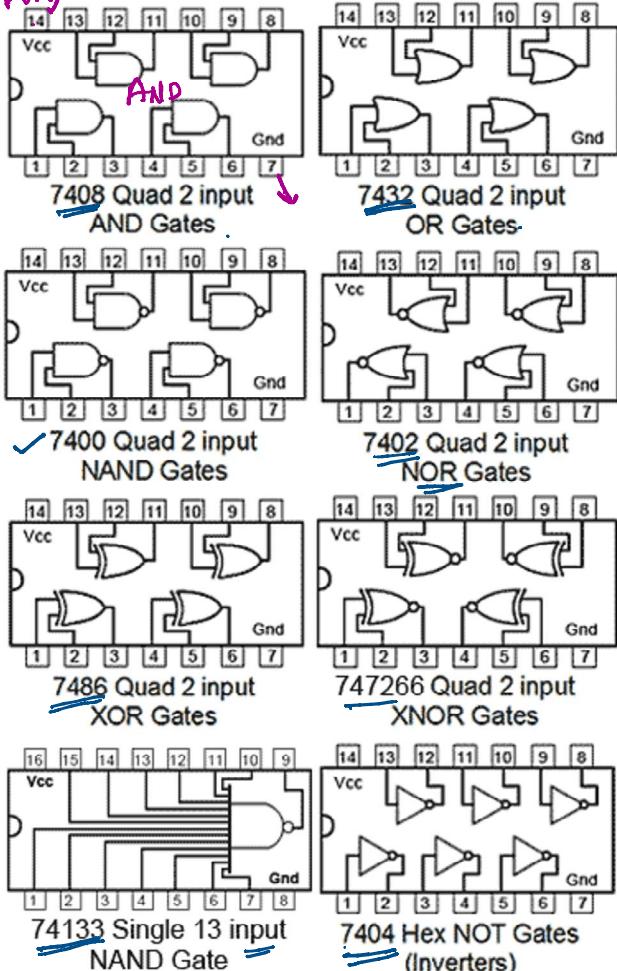
	S <sub>1</sub>	S <sub>2</sub>	S <sub>3</sub>	S <sub>4</sub>	Y(bulb)
1	1	1	1	0	1
1	1	1	0	1	1
1	1	1	1	1	1

Four switches control the operation of the bulb

'the bulb is switched on if the switches S<sub>1</sub> and S<sub>2</sub> are closed, and S<sub>3</sub> or S<sub>4</sub> is also closed, otherwise the bulb will not be switched on'

Relay operations in telephone exchanges is another example

14 Pin



7408

2  $\overline{1}$

2  $\overline{2}$

2  $\overline{3}$

2  $\overline{4}$

4  $\overline{0/p}$

$$8+9 = 12 \stackrel{p}{=} 0/p \& 0/p$$

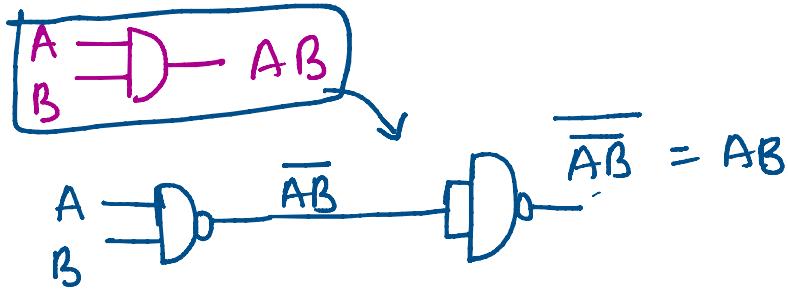
$\rightarrow V_{ce}$

$\checkmark 1 - \text{Ground}$

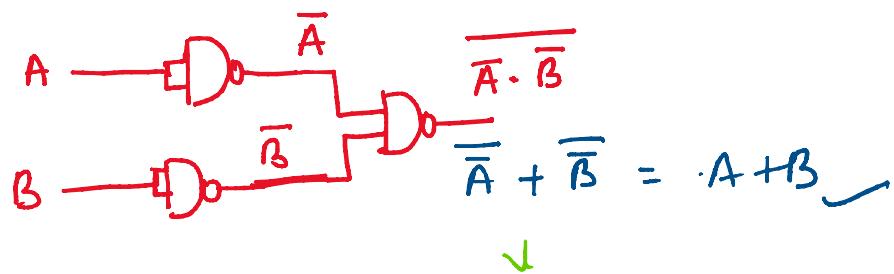
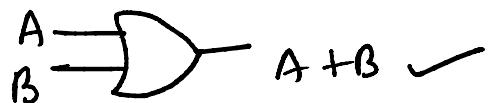
Universal gates (NAND and NOR)  
Any logic gate can be constructed using  
NAND and NOR

Realization of basic gates (AND, OR & NOT)  
using NAND gate

AND gate using NAND



OR gate using NAND



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

NOT gate using NAND

$$A \overline{D} \overline{A}$$

$$A \overline{D} \overline{A}$$

NAND gate is equivalent to bubble if OR gate

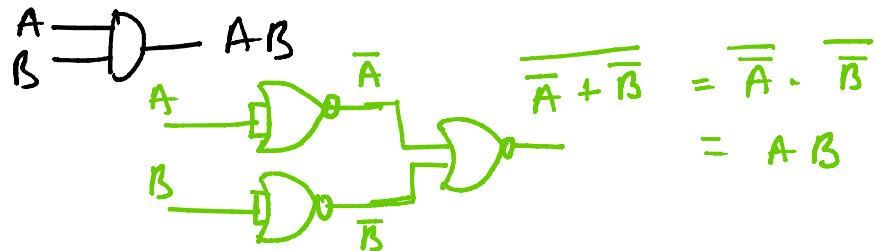
$$\overline{AB} = \overline{\overline{A} + \overline{B}}$$

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

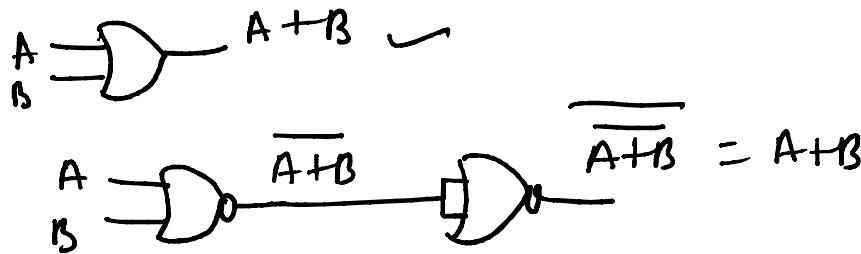
Realization of basic gates using NOR gate

Realization of basic gates using NOR gate

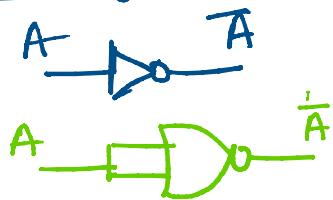
AND gate using NOR gate



OR gate using NOR gate

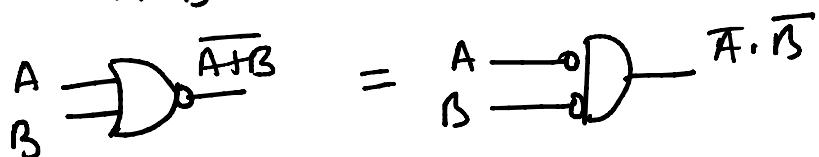


NOT gate using NOR gate



NOR gate is equivalent to bubble i/p AND gate

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



### NAND Realization

1. Implement given function using basic gates.
2. Convert AND to NAND gates with AND invert symbol
3. Convert OR to NAND gates with bubbled OR symbol
4. Where ever received an Inverter (step 2 and step 3) followed

by inverter use another inverter

### NOR Realization

1. Implement given function using basic gates.
2. Convert AND to NOR gates with bubbled input AND symbol
3. Convert OR to NOR gates with OR followed by invert symbol
4. Where ever received an Inverter (step2 and step 3)followed by inverter use another inverter