

# STUDY OF LOGIC GATES

---

**AIM:-**To verify and interpret the logic and truth table for OR gate using Diode Resistance Logic (DRL), NOT gate using Transistor and AND, NAND, NOR, Ex-OR & Ex-NOR gates using Resistor Transistor Logic (RTL) in simulator 1 and verify the truth table for AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates in simulator 2.

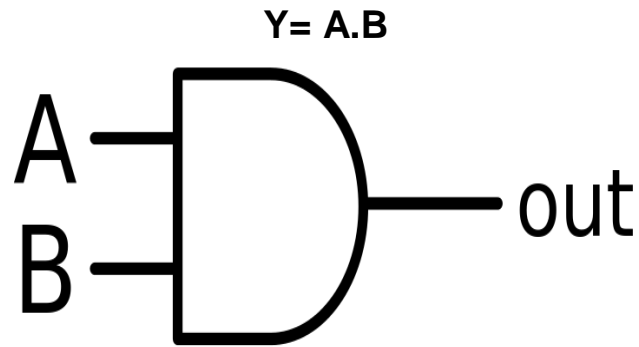
## Introduction

Logic gates are the basic building blocks of any digital system. Logic gates are electronic circuits having one or more than one input and only one output. The relationship between the input and the output is based on a certain logic. Based on this, logic gates are named as:

1. AND gate
2. OR gate
3. NOT gate
4. NAND gate
5. NOR gate
6. Ex-OR gate
7. Ex-NOR gate

## 1) AND gate

The AND gate is an electronic circuit that gives a high output (1) only if all its inputs are high. A dot (.) is used to show the AND operation i.e. A.B or can be written as AB

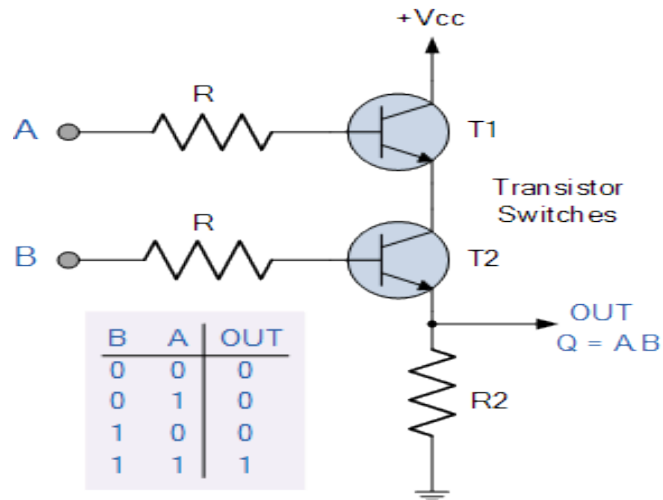


**Figure-1:Logic Symbol of AND Gate**

Input		Output
A	B	$Y=A.B$
0	0	0
0	1	0
1	0	0
1	1	1

**Figure-2:Truth Table of AND Gate**

A simple 2-input logic AND gate can be constructed using RTL (Resistor-Transistor-Logic) switches connected together as shown below with the inputs connected directly to the transistor bases. Both transistors must be saturated “ON” for an output at Q.

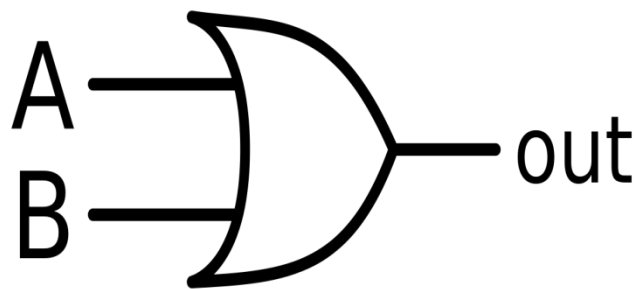


**Figure-3:AND Gate through RTL logic**

## 2) OR gate

The OR gate is an electronic circuit that gives a high output (1) if one or more of its inputs are high. A plus (+) is used to show the OR operation.

$$Y = A + B$$



**Figure-4:Logic Symbol of OR Gate**

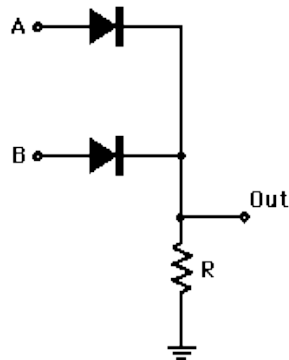
Input		Output
A	B	$Y=A+B$
0	0	0
0	1	1
1	0	1
1	1	1

**Figure-5: Truth Table of OR Gate**

OR gate can be realized by DRL (Diode-Resistance-Logic) or by TTL (Transistor-Transistor-Logic). Presently, we will learn how to implement the OR gate using DRL (Diode-Resistance-Logic). To realise OR gate, we will use a diode at every input of the OR gate. The anode part of diode is connected with input while the cathode part is joined together and a resistor, connected with the cathode is grounded. In this case, we have taken two inputs which can be seen in the circuit below.

When both the inputs are at logic 0 or low state then the diodes D1 and D2 become reverse biased. Since the anode terminal of diode is at lower voltage level than the cathode terminal, so diode will act as open circuit so there is no voltage across resistor and hence output voltage is same as ground. When either of the diodes is at logic 1 or high state then the diode corresponding to that input is forward bias. Since this time anode is at high voltage than cathode therefore current will flow through forward biased diode and this current then appears on resistor causing high voltage at output terminal also. Hence at output we get high or logic 1 or +5V. So, if any or both inputs are high, the output will be high or “1”.

Diode OR Gate

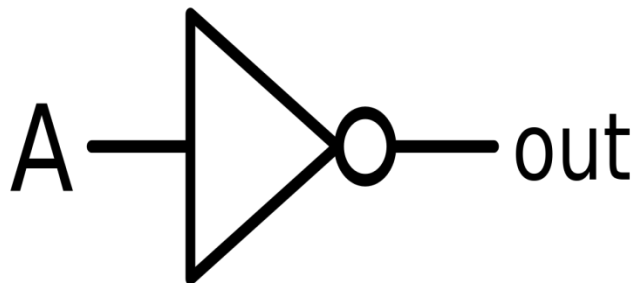


**Figure-6:OR Gate through DRL logic**

### 3) NOT gate

The NOT gate is an electronic circuit that produces an inverted version of the input at its output. It is also known as an inverter. If the input variable is A, the inverted output is known as NOT A. This is also shown as A' or A with a bar over the top, as shown at the outputs.

$$Y = A'$$



**Figure-7:Logic Symbol of NOT Gate**

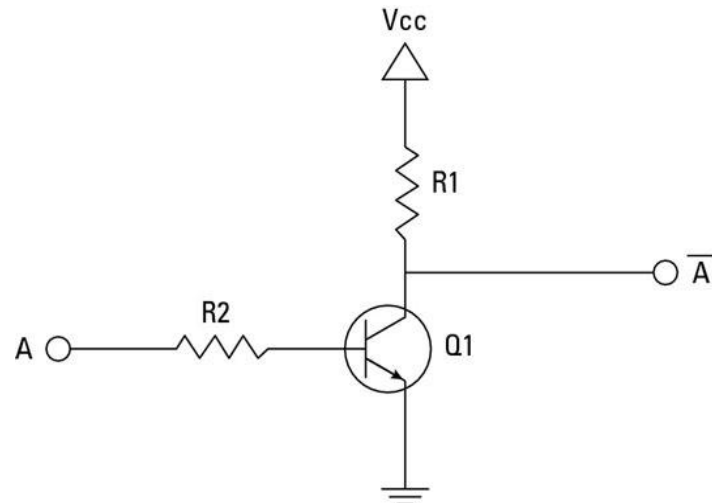
Input	Output
A	Y
0	1
1	0

**Figure-8: Truth Table of NOT Gate**

NOT gate can be realized through transistor. The input is connected through resistor R2 to the transistor's base. When no voltage is present on the input, the transistor turns off. When the transistor is off, no current flows through the collector-emitter path. Thus, current from the supply voltage ( $V_{cc}$ ) flows through resistor R1 to the output. In this way, the circuit's output is high when its input is low.

When voltage is present at the input, the transistor turns on, allowing current to flow through the collector-emitter circuit directly to ground. This ground path creates a shortcut that bypasses the output, which causes the output to go low.

In this way, the output is high when the input is low and low when the input is high.

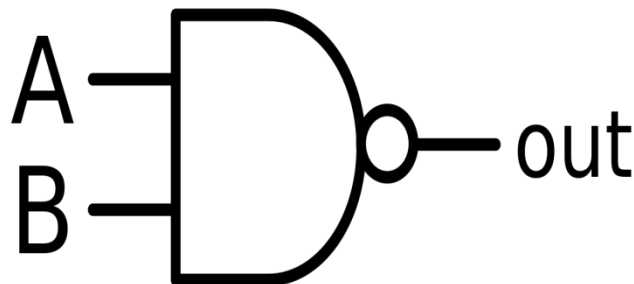


**Figure-9:NOT Gate through Transistor**

#### 4) NAND gate

This is a NOT-AND gate which is equal to an AND gate followed by a NOT gate. The outputs of all NAND gates are high if any of the inputs are low. The symbol is an AND gate with a small circle on the output. The small circle represents inversion.

$$Y = \overline{AB}$$

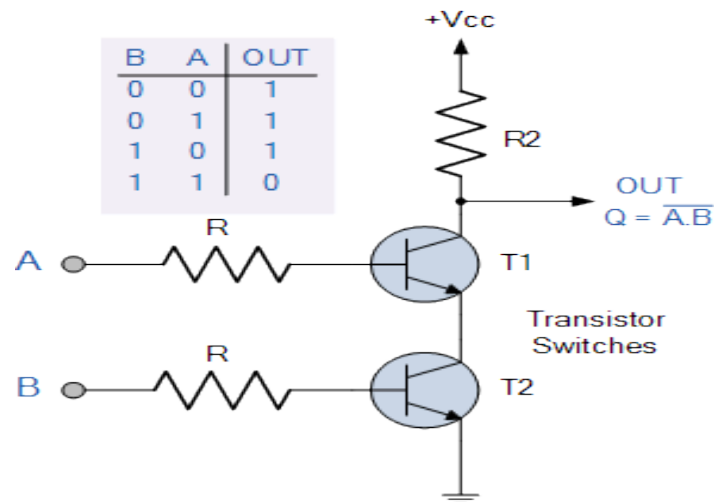


**Figure-10:Logic Symbol of NAND Gate**

Input	Input	Output
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

**Figure-11: Truth Table of NAND Gate**

A simple 2-input logic NAND gate can be constructed using RTL (Resistor-transistor-logic) switches connected together as shown below with the inputs connected directly to the transistor bases. Either transistor must be cut-off or “OFF” for an output at Q.



**Figure-12: NAND gate through RTL Logic.**

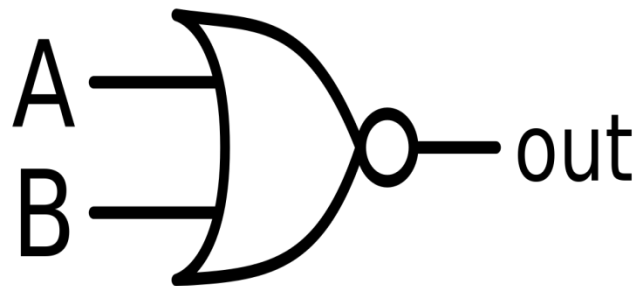
## 5) NOR gate

This is a NOT-OR gate which is equal to an OR gate followed by a NOT



gate. The outputs of all NOR gates are low if any of the inputs are high. The symbol is an OR gate with a small circle on the output. The small circle represents inversion.

$$Y = A + B$$



**Figure-13: Logic Symbol of NOR gate**

A	B	F
0	0	1
0	1	0
1	0	0
1	1	0

**Figure-14: Truth Table of NOR gate**

A simple 2-input logic NOR gate can be constructed using RTL (Resistor-transistor-logic) switches connected together as shown below with the inputs connected directly to the transistor bases. Both transistors must be cut-off or “OFF” for an output at Q.

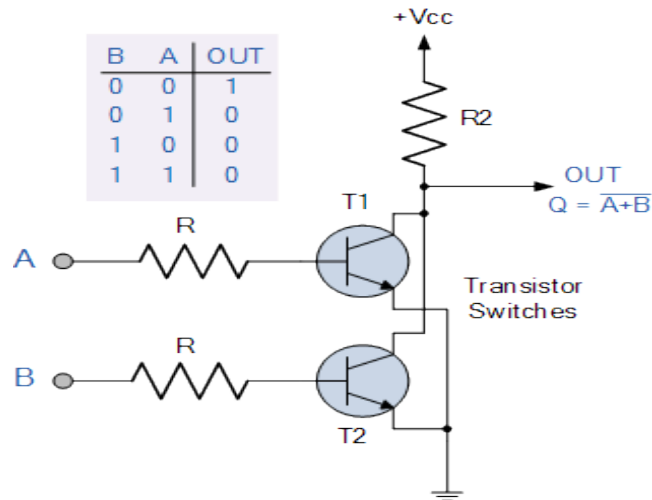


Figure-15:NOR gate through RTL Logic.

## 6) Ex-OR gate

The 'Exclusive-OR' gate is a circuit which will give a high output if either, but not both of its two inputs are high. An encircled plus sign ( $\oplus$ ) is used to show the Ex-OR operation.

$$Y = A \oplus B$$

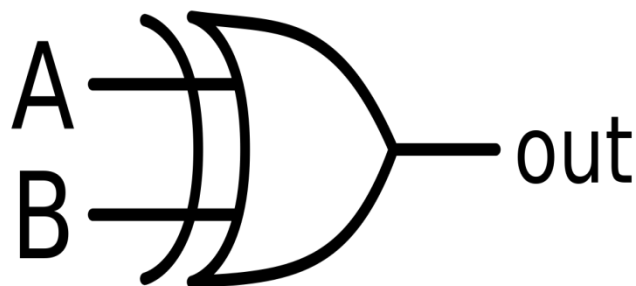
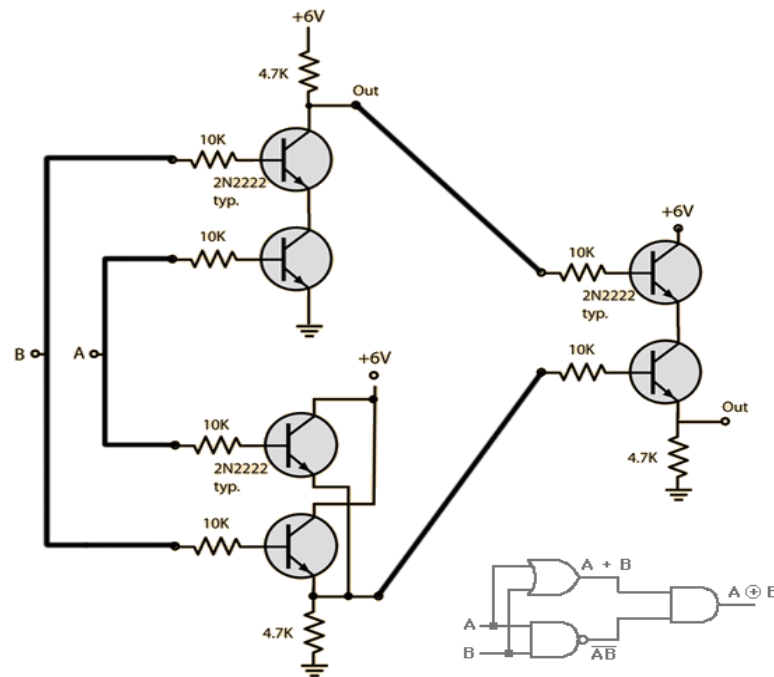


Figure-16:Logic Symbol of Ex-OR gate

A	B	A <b>XOR</b> B
0	0	0
0	1	1
1	0	1
1	1	0

**Figure-17:Truth Table of Ex-OR gate**

Ex-OR gate is created from AND, NAND and OR gates.The output is high only when both the inputs are different.

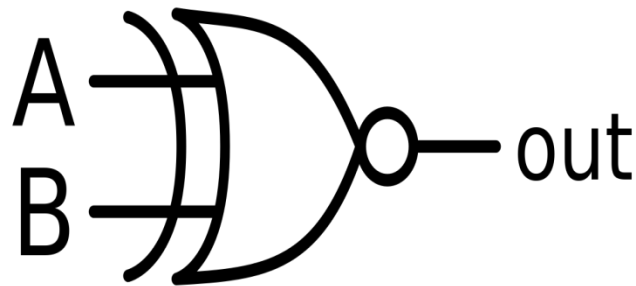


**Figure-18:Ex-OR gate through RTL Logic.**

## 7) Ex-NOR gate

The 'Exclusive-NOR' gate circuit does the opposite to the EX-OR gate. It will give a low output if either, but not both of its two inputs are high. The symbol is an EX-OR gate with a small circle on the output. The small circle represents inversion.

$$Y = A \oplus B$$



**Figure-19: Logic Symbol of Ex-NOR gate**

XNOR Truth Table		
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

**Figure-20: Truth Table of Ex-NOR gate**

Ex-NOR gate is created from AND, NOT and OR gates. The output is high only when both the inputs are same.

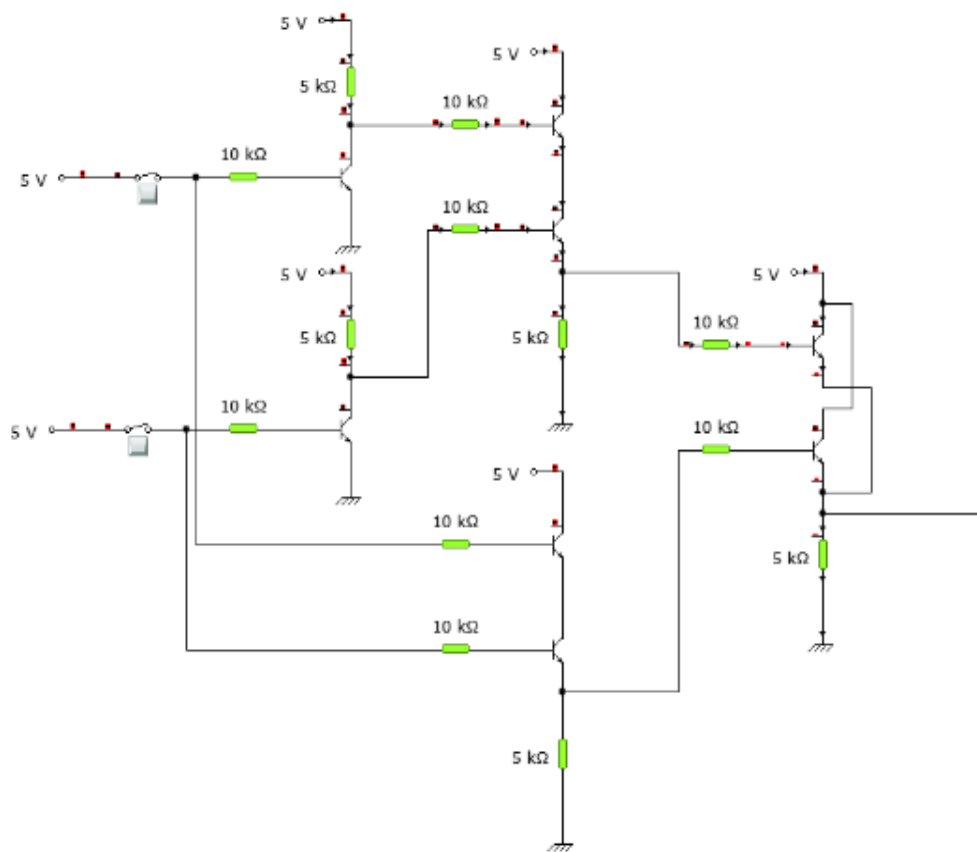


Figure-21:Ex-NOR gate through RTL Logic.

## PRETEST:

Electronic circuits that operate on one or more input signals to produce standard output \_\_\_\_\_

- ☐ a: Series circuits ☐ b: Parallel circuits ☐ c: Logic signals ☐ d: Logic gates

A \_\_\_\_\_ gate gives the output as 1 only if all the inputs signals are 1.

- ☐ a: AND ☐ b: OR ☐ c: NOR ☐ d: Ex-OR

The boolean expression of an OR gate is \_\_\_\_\_

- ☐ a:  $A.B$  ☐ b:  $A'B+AB'$  ☐ c:  $A+B$  ☐ d:  $A'B'$

The gate which is used to reverse the output obtained is \_\_\_\_\_

- ☐ a: NOR ☐ b: NAND ☐ c: EX-OR ☐ d: NOT

Which of the following gate will returns a 1 only if both the inputs are 0?

- ☐ a: AND ☐ b: OR ☐ c: NAND ☐ d: EX-OR


Submit Quiz

5 out of 5

## PROCEDURE:



### 1)AND Gate

#### Simulator 1:

Step-1) Connect the supply(+5V)  to the circuit.  
Step-2) Press the switches for inputs "A" and "B".

The switch in ON state is  and the switch in OFF state is .

Step-3) The bulb does not glow if any one or both the switches (2 and 3) are OFF and glows only if both the switches (2 and 3) are ON.

The bulb in OFF state is  and the bulb in ON state is .


Step-4) Repeat step-2 and step-3 for all state of inputs.

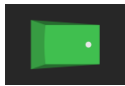

#### Simulator 2:

Step-1) Enter the Boolean input "A" and "B".  
 Step-2) Enter the Boolean output for your corresponding inputs.  
 Step-3) Click on "Check" Button to verify your output.  
 Step-4) Click "Print" if you want to get print out of Truth Table.



## 2)OR Gate

### Simulator 1:

Step-1) Connect the supply(+5V)  to the circuit.  
 Step-2) Press the switches for inputs "A" and "B".

The switch in ON state is  and the switch in OFF state is .

Step-3) The bulb glows if any one or both the switches are ON else it won't glow.

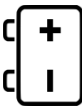
The bulb in OFF state is  and the bulb in ON state is .  
 Step-4) Repeat step-2 and step-3 for all state of inputs.



### Simulator 2:

Step-1) Enter the Boolean input "A" and "B".  
 Step-2) Enter the Boolean output for your corresponding inputs.  
 Step-3) Click on "Check" Button to verify your output.  
 Step-4) Click "Print" if you want to get print out of Truth Table.



## 3)NOT gate

### Simulator 1:

Step-1) Connect the supply(+5V)  to the circuit.  
Step-2) Press the switch 1 to connect battery to the circuit.  
Step-3) Press the switch 2 for input "A".

The switch in ON state is  and the switch in OFF state is .

Step-4) The bulb glows if switch 2 is OFF else it won't glow.

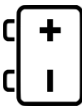
The bulb in OFF state is  and the bulb in ON state is .



### Simulator 2:

Step-1) Enter the Boolean input "A".  
Step-2) Enter the Boolean output for your corresponding input.  
Step-3) Click on "Check" Button to verify your output.  
Step-4) Click "Print" if you want to get print out of Truth Table.

### 4)NAND gate

### Simulator 1:

Step-1) Connect the supply(+5V)  to the circuit.  
Step-2) Press the switch 1 to connect battery to the circuit.  
Step-3) Press the switches 2 and 3 for inputs "A" and "B".

The switch in ON state is  and the switch in OFF state is .



Step-4) The bulb glows if any one or both the switches are OFF else it won't glow.


The bulb in OFF state is  and the bulb in ON state is .

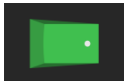

### **Simulator 2:**

- Step-1) Enter the Boolean input "A" and "B".
- Step-2) Enter the Boolean output for your corresponding inputs.
- Step-3) Click on "Check" Button to verify your output.
- Step-4) Click "Print" if you want to get print out of Truth Table.

### **5)NOR gate**

### **Simulator 1:**

- Step-1) Connect the supply(+5V)  to the circuit.
- Step-2) Press the switch 1 to connect battery to the circuit.
- Step-3) Press the switches 2 and 3 for inputs "A" and "B".

The switch in ON state is  and the switch in OFF state is .

Step-4) The bulb glows if both the switches are OFF else it won't glow.

The bulb in OFF state is  and the bulb in ON state is .


### **Simulator 2:**

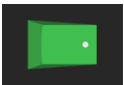

- Step-1) Enter the Boolean input "A" and "B".
- Step-2) Enter the Boolean output for your corresponding inputs.

Step-3) Click on "Check" Button to verify your output.  
Step-4) Click "Print" if you want to get print out of Truth Table.

## 6)Ex-OR gate

### Simulator 1:

Step-1) Connect the supply(+5V)  to the circuit .  
Step-2) Press the switches for inputs "A" and "B".

The switch in ON state is  and the switch in OFF state is .

Step-3) The bulb glows if one of the switches is ON and one of the switches is OFF else it won't glow.


The bulb in OFF state is  and the bulb in ON state is .



### Simulator 2:

Step-1) Enter the Boolean input "A" and "B".  
Step-2) Enter the Boolean output for your corresponding inputs.  
Step-3) Click on "Check" Button to verify your output.  
Step-4) Click "Print" if you want to get print out of Truth Table.

## 7)Ex-NOR gate

### Simulator 1:

Step-1) Connect the supply(+5V)  to the circuit .  
Step-2) Press the switches for inputs "A" and "B".

The switch in ON state is  and the switch in OFF state is .

Step-3) The bulb glows if both the switches are ON or if both the switches are OFF else it won't glow.

The bulb in OFF state is  and the bulb in ON state is .

## Simulator 2:

Step-1) Enter the Boolean input "A" and "B".  
Step-2) Enter the Boolean output for your corresponding inputs.  
Step-3) Click on "Check" Button to verify your output.  
Step-4) Click "Print" if you want to get print out of Truth Table.

## POSTTEST:

The expression of an Ex-OR gate is \_\_\_\_\_

- ☒ a:  $A'B+AB'$  ☐ b: 3 bits ☐ c:  $A+A.B$  ☐ d:  $A'+B'$

When logic gates are connected to form a gating/logic network it is called as a \_\_\_\_\_ logic circuit.

- ☒ a: Combinational ☐ b: Sequential ☐ c: Systematic ☐ d: Hardwired

The NOR gate is OR gate followed by \_\_\_\_\_

- ☐ a: AND gate ☐ b: NAND gate ☒ c: NOT gate ☐ d: None of the above

The NAND gate is AND gate followed by \_\_\_\_\_

- ☒ a: NOT gate ☐ b: OR gate ☐ c: AND gate ☐ d: None of the above

In boolean algebra, the bar sign (-) indicates \_\_\_\_\_

- ☐ a: OR operation ☐ b: AND operation ☒ c: NOT operation ☐ d: None of the above

## CONCLUSION:

1. In AND GATE both the inputs (all if more than 2) are high voltage (logic 1) which gives output as high voltage i.e. logic 1.
2. For OR GATE if any of the inputs is 1 (high) then output is 1 (high).
3. In NOT GATE, we get output as complement of input, i.e. for high (1) output is low (0) and for low (0) output is high (1).
4. NAND GATE is exactly opposite of AND GATE where we get output low if and only if all the inputs are high.
5. NOR gate is opposite of OR GATE, where we get high output when all the inputs are low.
6. EX-OR GATE is used to get low output as a result of high inputs only.
7. EX-NOR GATE is used to get high voltage as a output when both (or all) are either high or low i.e. both 0 or both 1.

SIGNATURE

