**Mustansiriyah University**

**College of Engineering**

**Electrical Engineering Dept.**


# Fundamentals of Logic Circuits Laboratory

**First Year**

**Object:** To implement and verify the operation of logic gates.

**Theory:**

A logic gate is an electronic device that performs a Boolean operation on one or more inputs to produce an output. Binary logic deals with variables that take on two discrete values and with operations that assume logical meaning. The two values that a Boolean variable may take can be indicated as (True, False), (Yes, No), (ON, OFF) or (1,0).

**Logic Gates:**

1. AND gate: the AND gate is a circuit., which gives a high output (logic 1) if all inputs are high. A dot (.) is used to indicate the AND operation. In practice, however, the dot is usually omitted.
2. OR gate: the OR gate is a circuit., which gives a high output if one or more of its inputs are high. A plus sign (+) is used to indicate the OR operation.
3. NOT gate: the NOT gate is circuit., which produces at its output the negated (inverted) version of its input logic the circuit. Is known as an inverter. If the input is A, the inverted output is written as $\bar{A}$.
4. NAND gate: the NAND gate is a NOT- AND circuit, which is equivalent to an AND circuit followed by a NOT circuit The output of the NAND gate is high if any of its inputs is low.

5. NOR gate: the NOR gate is a NOT- OR circuit, which is equivalent to AND circuit Followed by a NOT circuit. The output of the NOR gate is low if any of its inputs is high.
6. EX-OR gate: the exclusive-OR gate is a circuit, which gives a high output if its two inputs are different. A circuited plus sign $\oplus$ is used to indicate the (EX-OR) operation.
7. EX-NOR gate: the exclusive-NOR gate is a circuit, which gives a high output if its two inputs are similar.

The seven gates that are the fundamental logic elements in digital system are illustrated in Fig. (1.1).

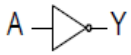
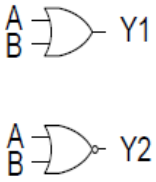
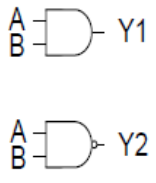
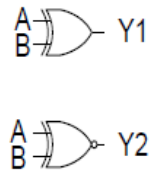
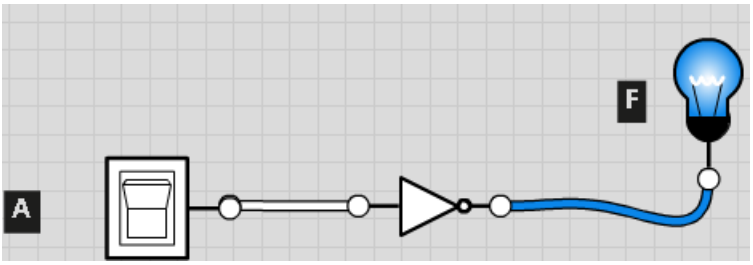
<table>
<tr><th>Logic Function</th><th>Logic Gate Symbol</th><th>Truth Table</th><th>Boolean Expression</th></tr>
<tr><td>NOT</td><td> 
</td><td><table><tr><th>Input (A)</th><th>Output (Y)</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></td><td>$Y=\overline{A}$</td></tr>
<tr><td>OR<br><br>NOR</td><td> 
</td><td><table><tr><th colspan="2">Input</th><th colspan="2">Output</th></tr><tr><th>A</th><th>B</th><th>Y1</th><th>Y2</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr></table></td><td>Y1= A+B<br><br>Y2= $\overline{A+B}$</td></tr>
<tr><td>AND<br><br>NAND</td><td> 
</td><td><table><tr><th colspan="2">Input</th><th colspan="2">Output</th></tr><tr><th>A</th><th>B</th><th>Y1</th><th>Y2</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr></table></td><td>Y1= A.B<br><br>Y2= $\overline{A.B}$</td></tr>
<tr><td>EX-OR<br><br>EX-NOR</td><td> 
</td><td><table><tr><th colspan="2">Input</th><th colspan="2">Output</th></tr><tr><th>A</th><th>B</th><th>Y1</th><th>Y2</th></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr></table></td><td>Y1= A⊕ B<br><br>Y1= $\overline{A\oplus B}$</td></tr>
</table>

**Fig. 1.1. The fundamental of basic logic gates.**

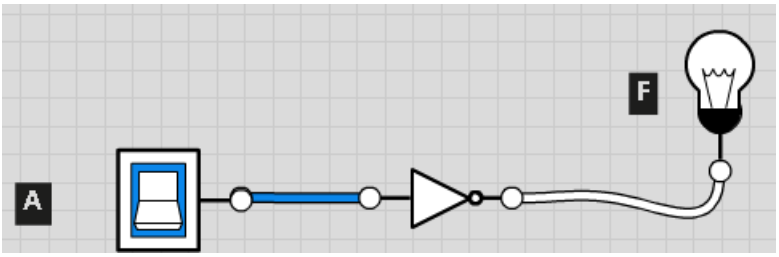
## Procedure:

By means of using logic gates, find the truth table of all possible gates shown in Fig. (1.1).

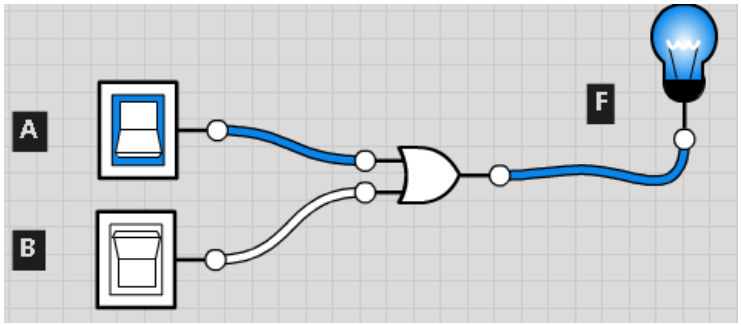
As presented below, use the logic gates that mentioned in Fig. 1.1. connect toggle switches with different logic gates on one side, the other side associated within the light bulb shown in Fig. 1.2.



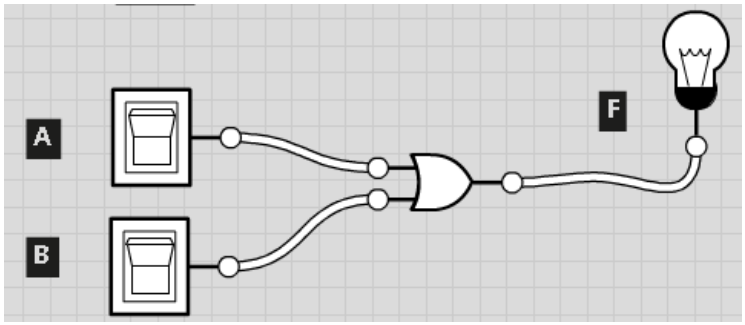

(a)



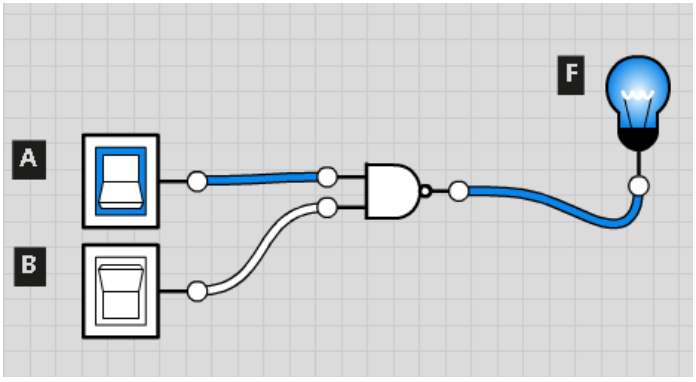

(b)



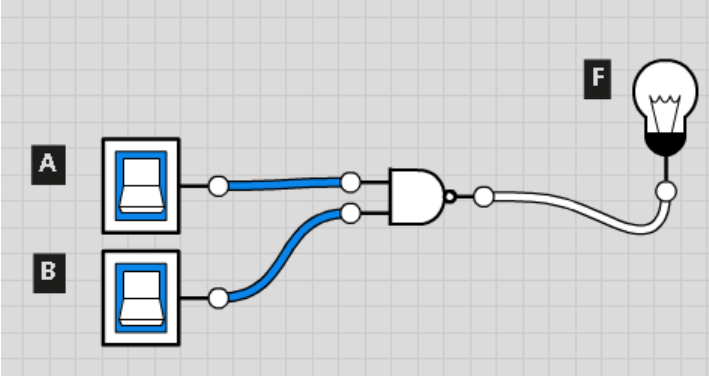

(c)



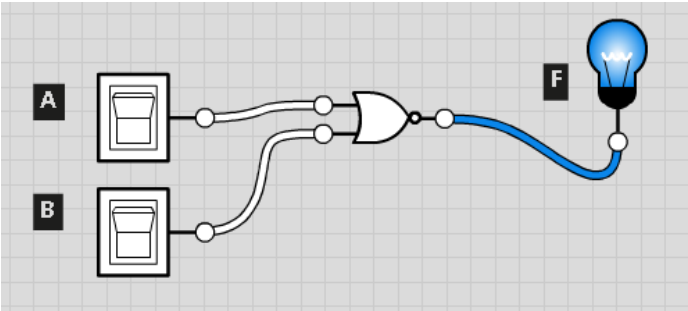

(d)



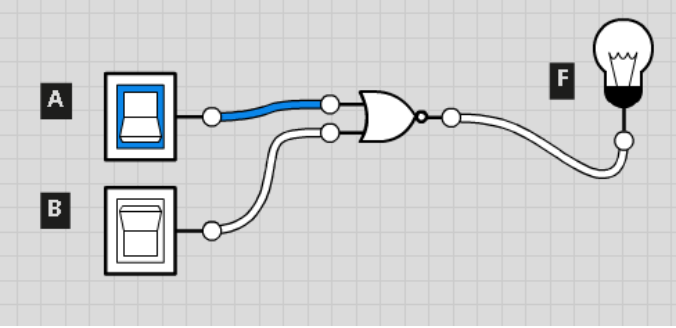

(e)



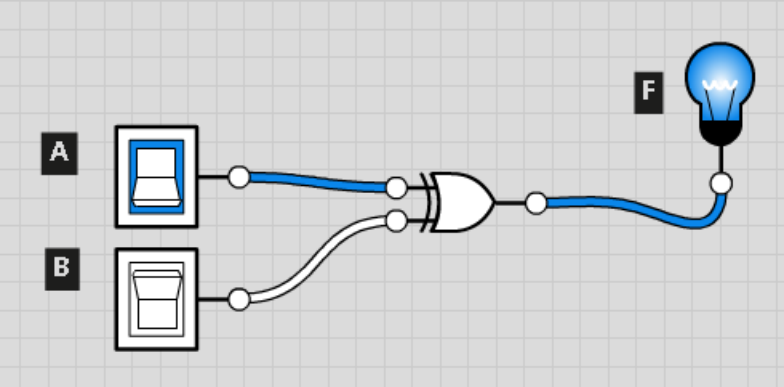

(f)



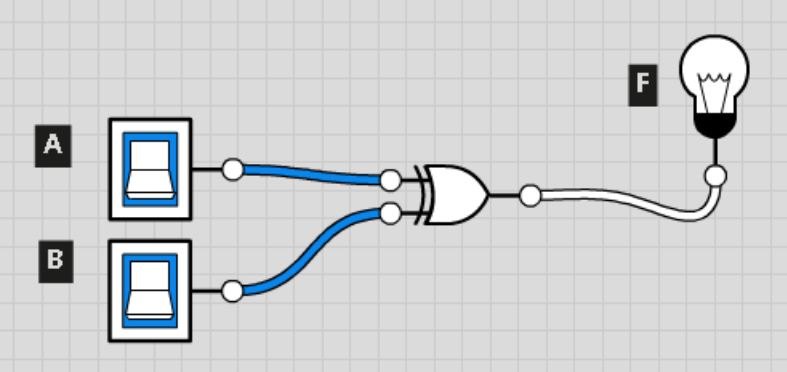

(g)




(h)




(i)




(j)

**Fig. 1.2. The execution of logic gates with different states.**

**Discussion:**

1- Implement the following functions and list their Truth Table using:

a. Mixing gates.

b. NAND gates only.

c. NOR gates only.

$$F1=A+\overline{B}$$

$$F2=(A+B)(A+C)$$

2- Implement the logic functions of the following gates using NOR gates only then list their Truth Table.

a. NOT gate.

b. OR gate.

c. AND gate.

d. NAND gate.