

Studies On Logic Gates

Name :	Keerti P. Charantimath
Roll Number :	19MA20059

1. AIM OF THE EXPERIMENT:-

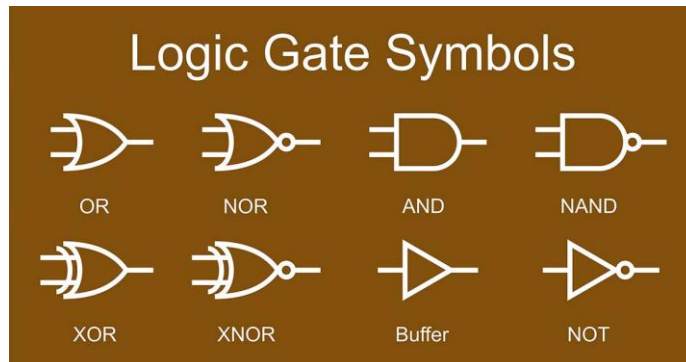
- First Experiment –
 - Verify - $AB + A'C + BC = AB + A'C$
- Second Experiment
 - Verify - $AB + A'C = (A + C)(A' + B)$
- Third Experiment -
 - Verify - $AB + CD = ((AB)'(CD)')'$
- Fourth Experiment -
 - Verify - $(A + B)(C + D) = ((A + B)'(C + D)')'$

2. TOOLS USED:-

- 7400(quad 2-input NAND gates)
- 7402(quad 2-input NOR gates)
- 7404(Hex inverter)
- 7408(quad 2-input AND gates)
- 7432(quad 2-input OR gates)

3. BACKGROUND KNOWLEDGE:-

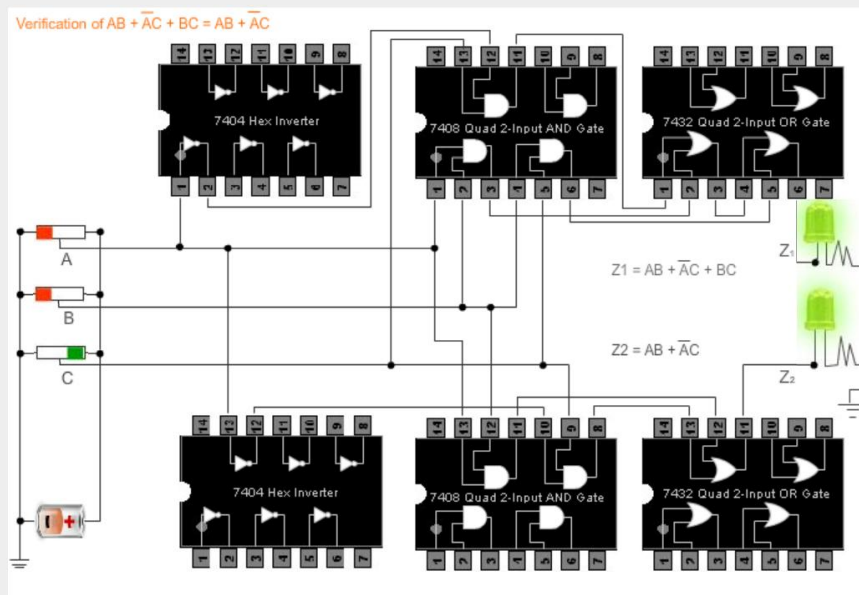
- Logic gates are the basic building blocks of any digital system. It is an electronic circuit having one or more than one input and only one output. The **relationship between the input and the output is based on certain logic.**
- Based on this, logic gates are named as AND gate, OR gate, NOT gate, NAND gate, NOR gate.



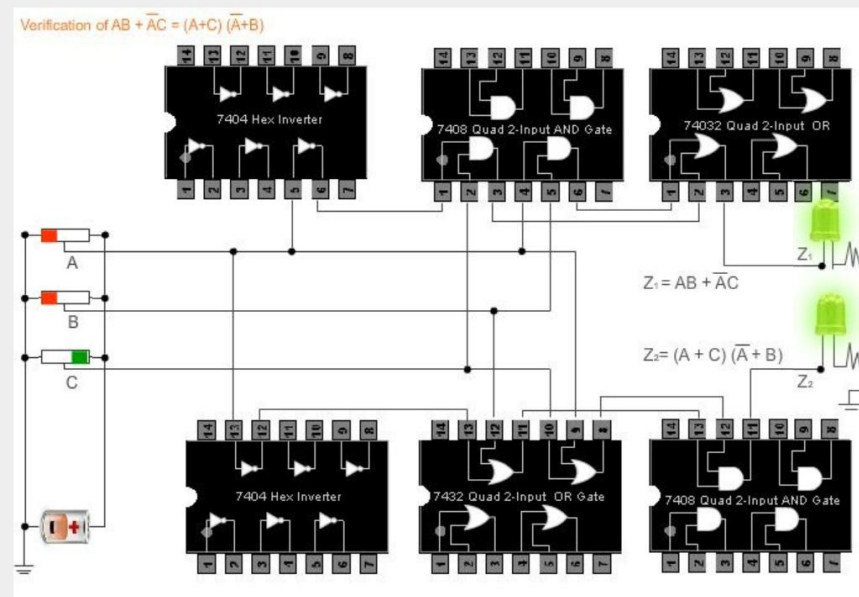
- The **AND gate** is a basic digital logic gate that implements logical conjunction. A HIGH output (1) results only if all the inputs to the AND gate are HIGH (1).
- The **OR gate** is a digital logic gate that implements logical OR. A HIGH output (1) results if one or both the inputs to the gate are HIGH (1). If neither input is high, a LOW output (0) results.
- The **NOT gate**, often called an inverter, is a basic digital logic gate to start with because it has only a single input with simple behavior. A NOT gate performs logical negation on its input. In other words, if the input is true, then the output will be false. Similarly, a false input results in a true output.
- The **NAND gate** (NOT-AND) is a logic gate which produces an output which is false only if all its inputs are true; thus its output is complement to that of an AND gate. A LOW (0) output results only if all the inputs to the gate are HIGH (1).
- The **NOR gate** is a digital logic gate that implements logical NOR - it behaves according to the truth table to the right. A HIGH output (1) results if both the inputs to the gate are LOW (0). NOR is the result of the negation of the OR operator.

4. CIRCUIT DIAGRAM :-

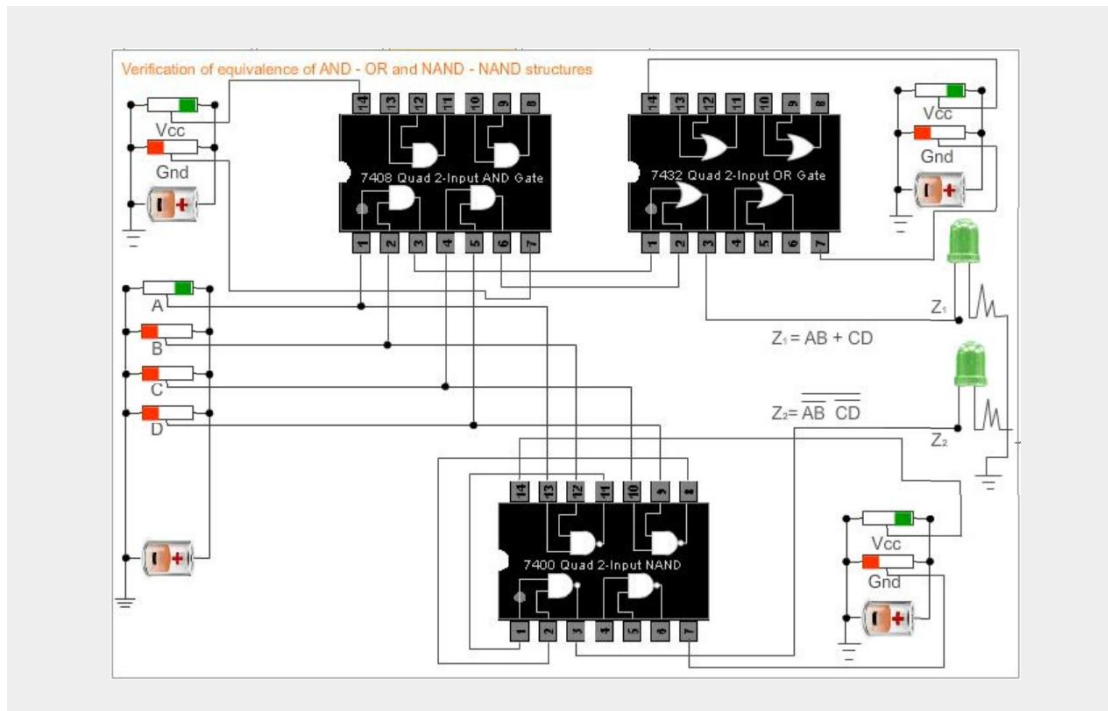
First Experiment - Verify - $AB + A'C + BC = AB + A'C$



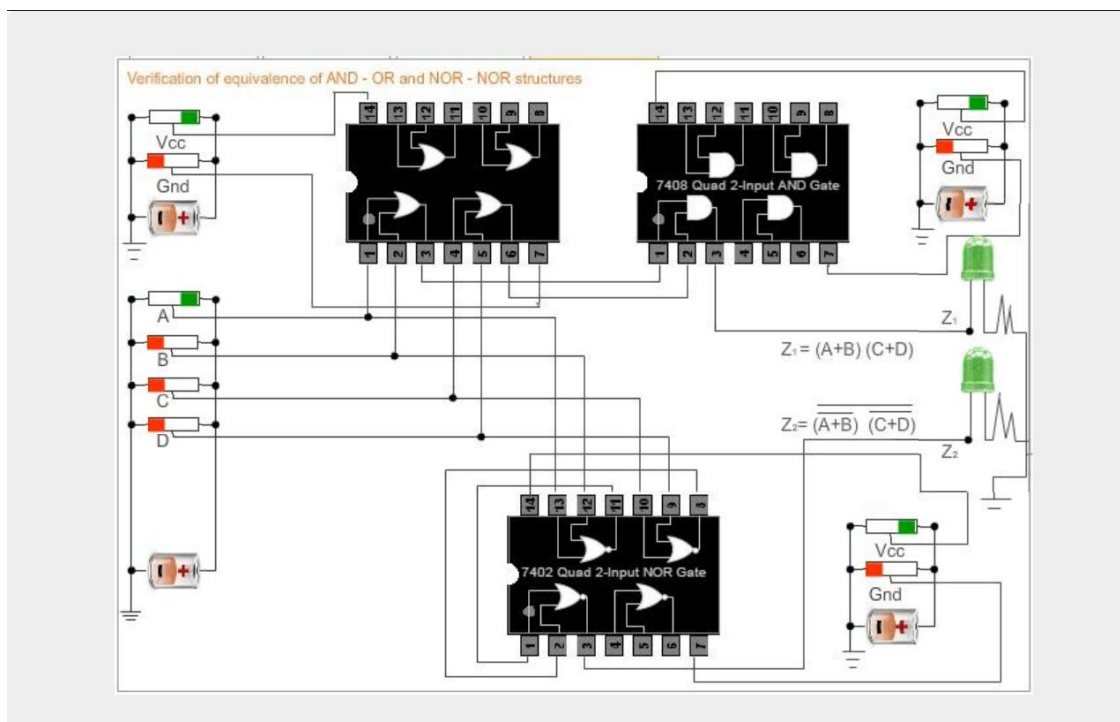
Second Experiment - Verify - $AB + A'C = (A + C)(\bar{A} + B)$



Third Experiment -
Verify - $AB + CD = ((AB)'(CD)')'$



Fourth Experiment -
Verify - $(A + B)(C + D) = ((A + B)'(C + D)')$



5. TRUTH TABLES :-

- First Experiment -
Verify - $AB + A'C + BC = AB + A'C$

A	B	C	$AB + A'C + BC$	$AB + A'C$
1	1	1	1	1
1	1	0	1	1
1	0	1	0	0
1	0	0	0	0
0	1	1	1	1
0	1	0	0	0
0	0	1	1	1
0	0	0	0	0

- Second Experiment -
Verify - $AB + A'C = (A + C)(A' + B)$

A	B	C	$AB + A'C$	$(A + C)(A' + B)$
1	1	1	1	1
1	1	0	1	1
1	0	1	0	0
1	0	0	0	0
0	1	1	1	1
0	1	0	0	0
0	0	1	1	1
0	0	0	0	0

- Third Experiment -
Verify - $AB + CD = ((AB)'(CD)')'$

A	B	C	D	$AB+CD$	$((AB)'(CD)')'$
1	1	1	1	1	1
1	1	1	0	1	1
1	1	0	1	1	1
1	1	0	0	1	1
1	0	1	1	1	1
1	0	1	0	0	0
1	0	0	1	0	0
1	0	0	0	0	0
0	1	1	1	1	1
0	1	1	0	0	0
0	1	0	1	0	0

0	1	0	0	0	0
0	0	1	1	1	1
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	0	0

- Fourth Experiment -**

Verify - $(A + B)(C + D) = ((A + B)'(C + D)')'$

A	B	C	D	$(A+B)(C+D)$	$((A+B)'(C+D)')$
1	1	1	1	1	1
1	1	1	0	1	1
1	1	0	1	1	1
1	1	0	0	0	0
1	0	1	1	1	1
1	0	1	0	1	1
1	0	0	1	1	1
1	0	0	0	0	0
0	1	1	1	1	1
0	1	1	0	1	1
0	1	0	1	1	1
0	1	0	0	0	0
0	0	1	1	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	0	0

6. CONCLUSION

- First Experiment -**

- The values measured for $(AB + A'C + BC)$ and $(AB + A'C)$ from the truth table and the experiment are the same.

- Second Experiment -**

- The values measured for $(AB + A'C)$ and $(A + C)(A' + B)$ from the truth table and the experiment are the same.

- Third Experiment -**

- The values measured for $(AB + CD)$ and $((AB)'(CD)')'$ from the truth table and the experiment are the same.

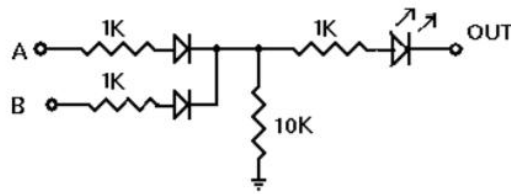
- Fourth Experiment -**

- The values measured for $(A + B)(C + D)$ and $((A + B)'(C + D)')$ from the truth table and the experiment are the same.

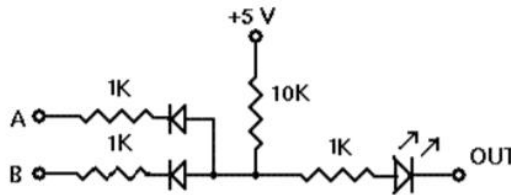
7. DISCUSSION:-

- Simple digital logic gates can be made by combining transistors, diodes and resistors as discrete components
- Diode logic gates use diodes to perform OR and AND logic functions as shown in the circuit diagram. Diodes have the property of easily passing an electrical current in one direction, but

not the other. Thus, diodes can act as a logical switch. Diode logic gates are very simple and inexpensive, and can be used effectively in limited space. They cannot perform a NOT function, so their usefulness is quite limited.

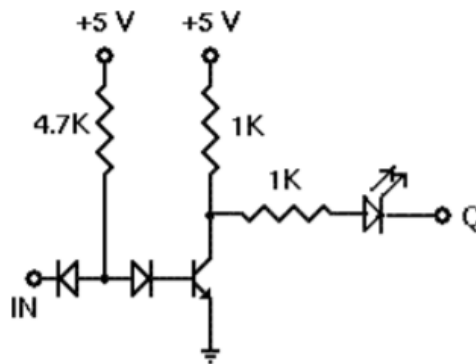


DRL OR gate

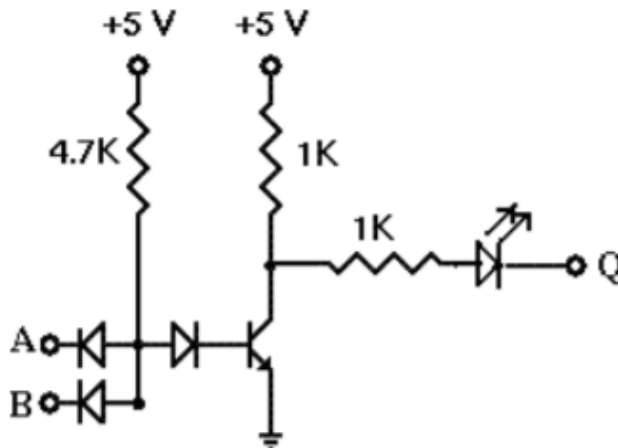


DRL AND Gate

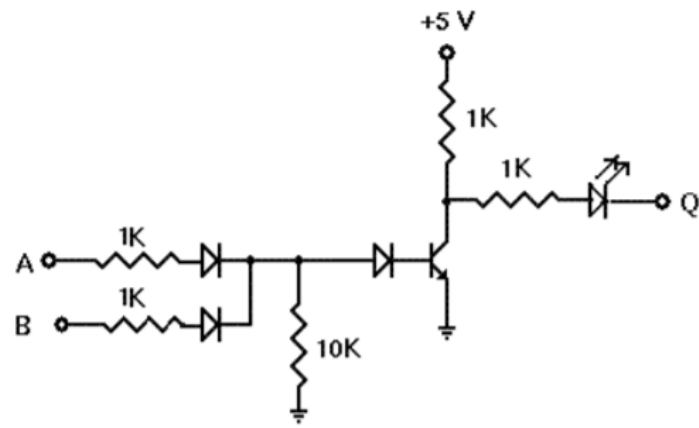
- The NOT gate uses a transistor and a collector load resistor as shown in the circuit diagram. The input is connected through a pair of diodes in series with the base of the transistor. The diode connected directly to the transistor base serves to raise the input voltage required to turn the transistor on to about 1.3 to 1.4 volts. Any input voltage below this threshold will hold the transistor off.



- NAND gate combines NOT gate with a simple Diode-Resistor Logic AND gate as shown in its circuit diagram. Thus, any number of inputs can be added simply by adding input diodes to the circuit.



- Similar to NAND circuit one can construct the NOR gate by using a OR gate followed by a transistor inverter (NOT gate), as shown in circuit diagram.



- The applications of Logic Gates are:
 - NAND Gates are used in Burglar alarms and buzzers.
 - They are basically used in circuits involving computation and processing.
 - They are also used in push button switches. E.g. Door Bell.
 - They are used in the functioning of street lights.
 - AND Gates are used to enable/inhibit the data transfer function.
- The limitations of logic gates are
 - Operating Voltage is limited.
 - Time delay occurs between input and output.