Experiments Name: Universal Logic Gates and Boolean Functions

Objective:

- Understand the concept of Universal Gates (NAND & NOR)
- Implement the basic logic gates using universal gates.
- Implement Boolean functions using universal gates.
- Understand gate level minimization.

Apparatus:

- IC 7400 Quadruple 2-input NAND gates
- IC 7402 Quadruple 2-input NOR gates
- Trainer Board
- Wires

Theory:

NAND as Universal Gate:

The NAND gate is an essential digital logic gate that can be used to implement any logical function. This means that any logical function that can be implemented using a combination of AND, OR, and NOT gates can also be implemented using only NAND gates.

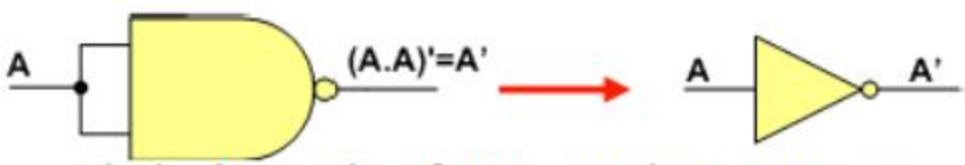


Fig: implementation of NOT gate using NAND gate

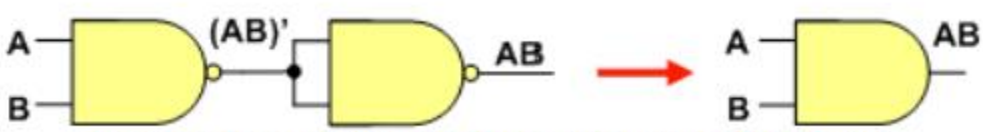


Fig: implementation of AND gate using NAND gate

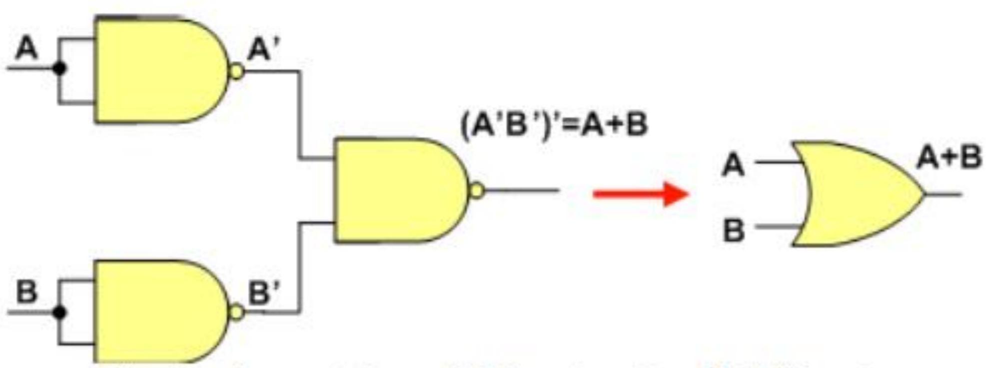
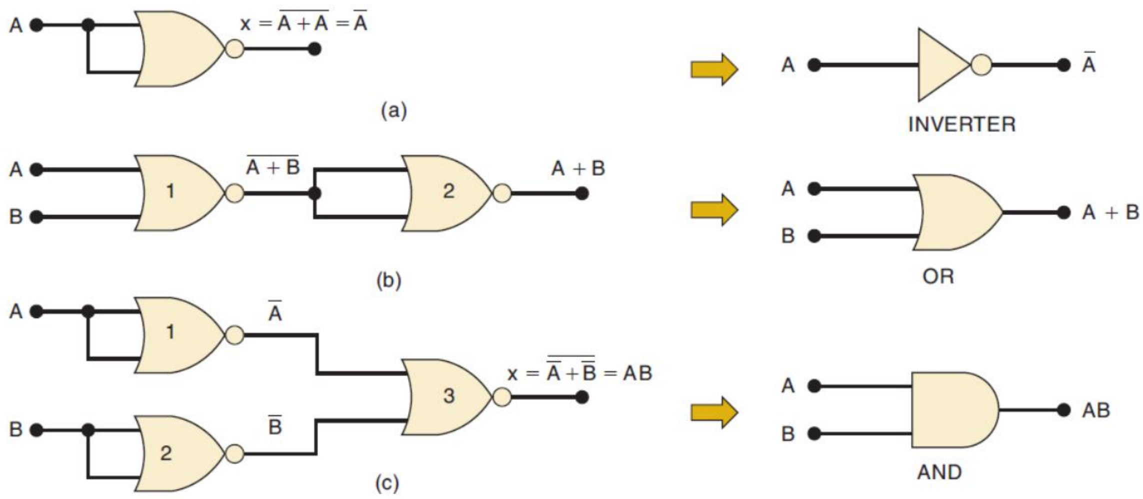
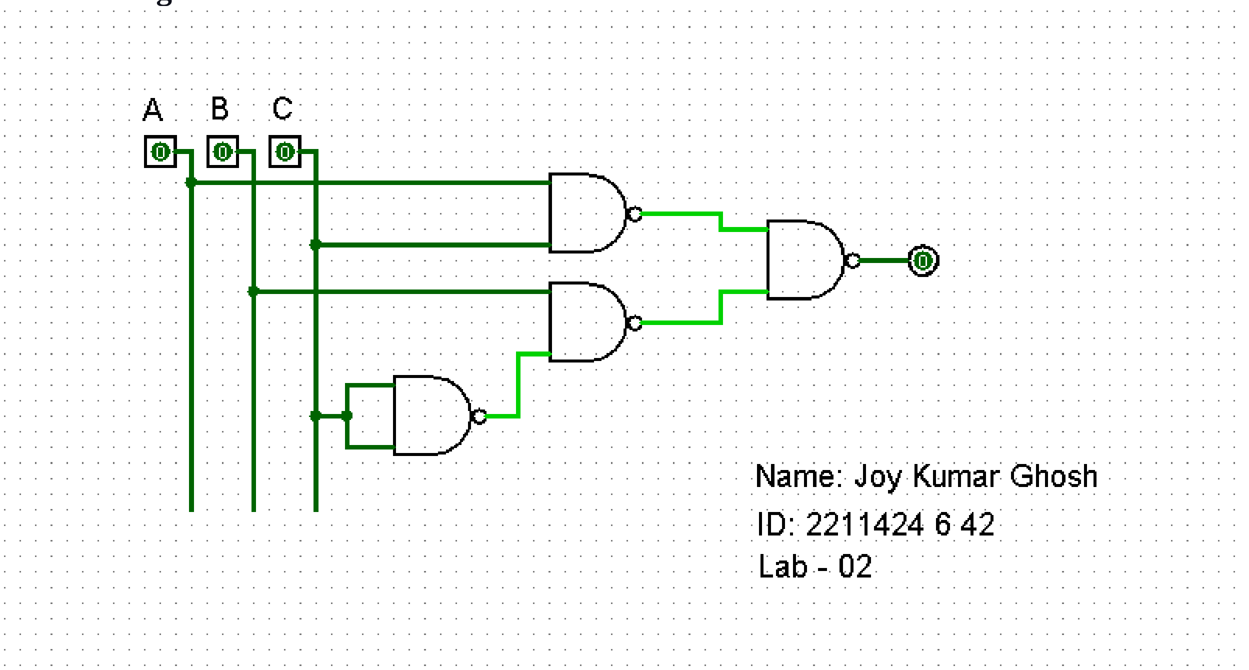


Fig: implementation of OR gate using NAND gate

NOR as Universal Gate:
The NOR gate is another digital logic gate that can also be used to implement any logical function. This means that any logical function that can be implemented using a combination of AND, OR, and NOT gates can also be implemented using only NOR gates.



Circuit Diagram:



Experimental Procedure:

01. First, we implemented basic gates using NAND gate as showed in theory part of the NAND as Universal Gates. And verify it working or not.

02. Then we implemented XOR circuit using only NAND gates as showed in Figure F1 with the labeling of PIN numbers.
03. After that we just add another NAND IC and make a NOT gate for converting the XOR to XNOR as showed in Figure F1.
04. Then we simplified the circuit in Figure D2, reduce the inversion operation and implemented it using the NAND gate only as showed in the Figure F3(Part-2), and verify it with the truth table of table-1.
05. In the next day, we implemented all the basic gates using NOR gate as showed in Figure F2(with PIN Labels).
06. Then we also implemented the circuit of XNOR gate using the NOR gate only.
07. After that we just add a NOT gate using NOR then the circuit converted into XOR gate. We connect the PIN as showed in the Figure F2.

Simulation:
Attached.

Experimental Data Table:

E. Experimental Data

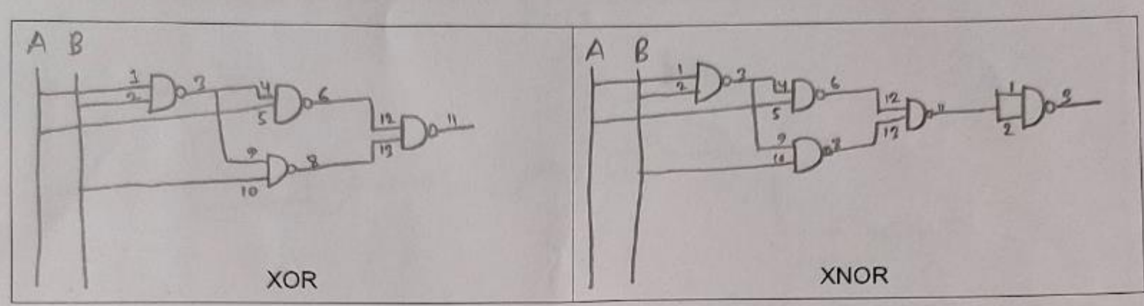


Figure F1: Implementation of XOR and XNOR using NAND gates

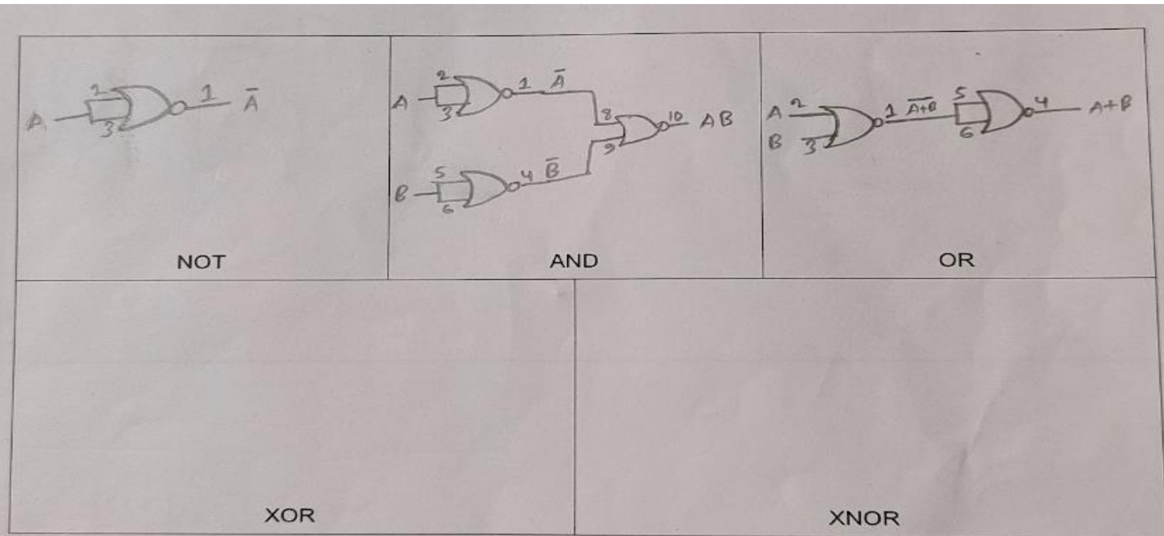
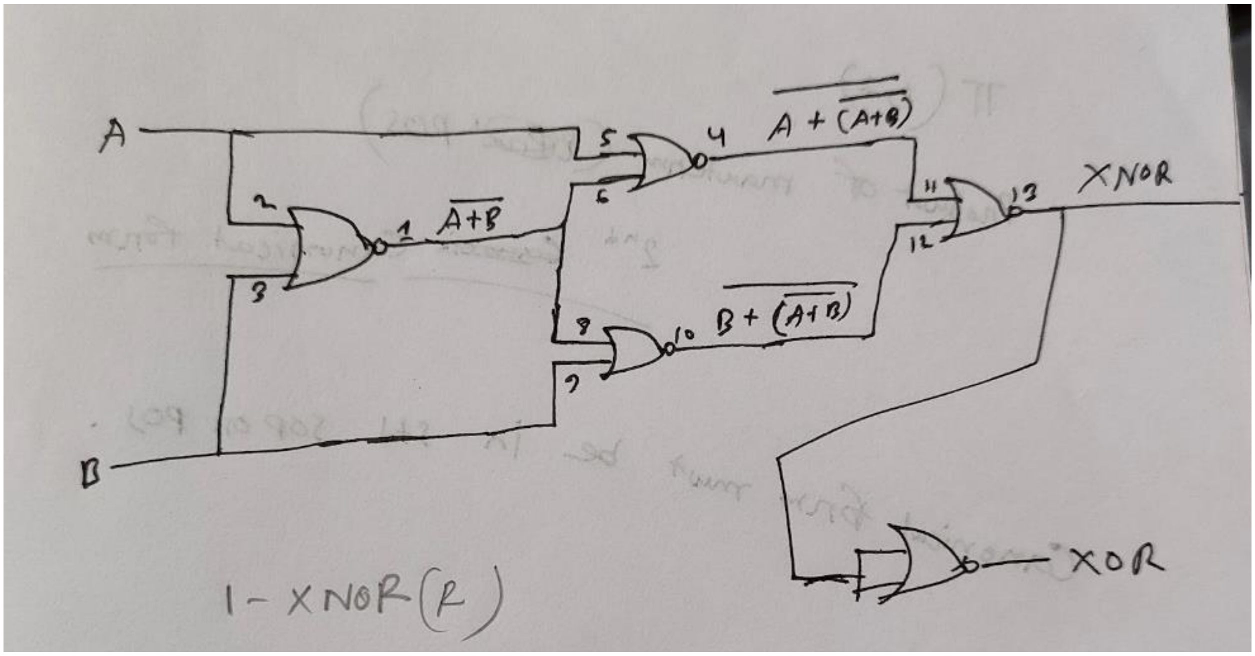


Figure F2: Implementation of NOT, AND, OR, XOR and XNOR using NOR gates

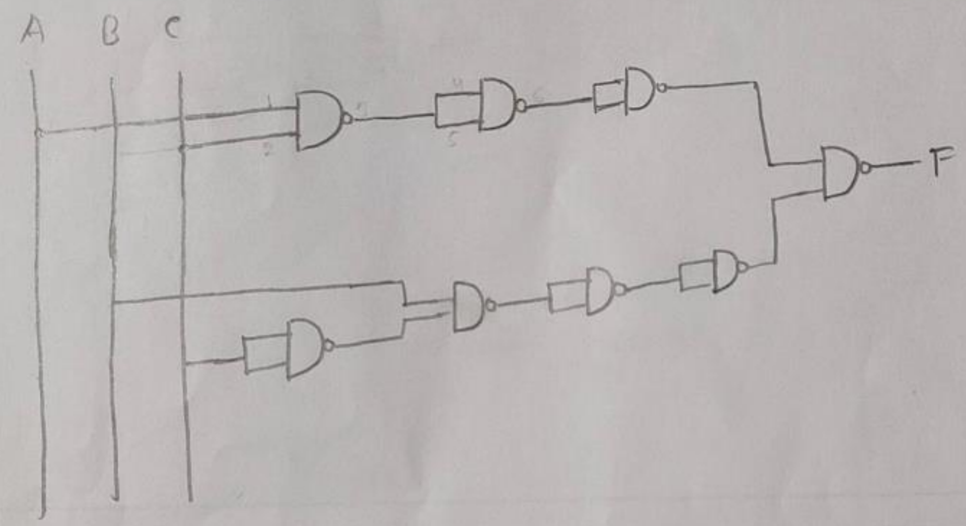


A B C	I_1 = AC	I_2 = BC'	F = I_1 + I_2
0 0 0	0	0	0
0 0 1	0	0	0
0 1 0	0	1	1
0 1 1	0	0	0
1 0 0	0	0	0
1 0 1	1	0	1
1 1 0	0	1	1
1 1 1	1	0	1

Table F1: Truth table of combinational circuit in Figure B2

Table F1: Truth table of combinational circuit in Figure B2

Part 1



Part 2

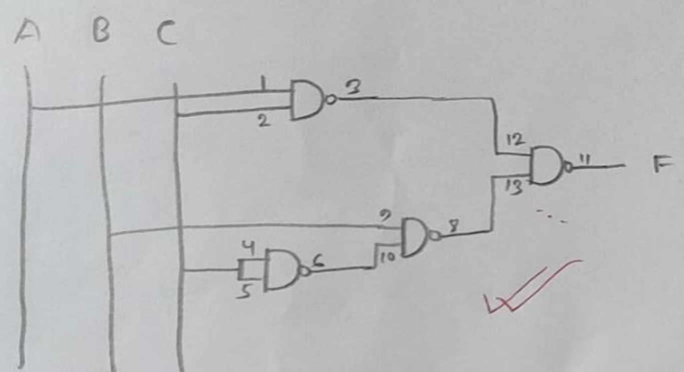


Figure F3: Universal (NAND) gate implementation of the circuit of Figure D2

Results:

After making the circuit in figure D2 using only NAND Gate, we get the exact output as the truth table of table F1. Hence, we implement the circuit without any errors.

Questions and Answers (Q/A):

01. NAND and NOR gates are called universal gates because they can be used to implement any digital logic function. This means that any complex digital system can be constructed using only NAND or NOR gates without any other types of gates.
    It is economical to use only one type of gate to produce digital logic ICs because it simplifies the manufacturing process and reduces costs. Instead of producing and stocking different types of gates, the manufacturer can focus on producing a single type of gate in large quantities. This reduces the cost of production and enables the manufacturer to offer the gates at a lower price.
    In short, using a single type of gate, such as NAND or NOR gates, to produce digital logic ICs is cost-effective and efficient and can simplify manufacturing and design processes.
02. Drawing Pending for Sazid
03. Drawing Pending for Sazid
04. Drawing IC Diagram for the circuit in Figure F3 – Part 2.
05. Simulation Attached.

Discussion:

This experiment taught us about two universal gates, NAND, and NOR. We learn about the implementation of basic gates using NAND and NOR. We also implement some Boolean functions using these Gates. And learn about identifying and minimizing any inversions in a circuit. We don't face any difficulty in this experiment. Everything works perfectly.