



North South University

Department of Electrical & Computer Engineering

Lab Report

Experiment No: 02

Experiment Title: Universal Logic Gates

Course Code: CSE231L

Section: 10

Course Name: Digital Logic Design Lab

Lab Group #: 07

Written By: Tawhid Hasan (ID-2413054042) Md. Tayasshuk Imam
(2411600042)

Date of Experiment: 04/10/2025

Date of Submission: 11/10/2025

Group Members ID:	Group Members Name:
2413054042	Tawhid Hasan
2413692042	Tasvirul Hasan Riyad
2411600042	Md Tayasshuk Imam
2322723642	Md Mafiul Haque

Objectives :

1. Understand the concept of Universal Gates (NAND & NOR)
2. Implement the basic logic gates using universal gates.
3. Implement boolean functions using universal gates.
4. Understand gate level minimization.

Apparatus :

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

Theory :

Universal gates are those type of gates that can be used for implementing any Boolean function without need to use any other gates. A universal gate is capable of performing all the basic logical operations, such as AND, OR, NOT, and more complex operations like XOR. There are two main types of universal gates : NAND gate and NOR gate.

Any Boolean function can be implemented using these universal gates. The advantage is that NAND and NOR gates are easier and cost efficient to fabricate and the basic gates used in all IC digital logic families.

Circuit Diagram:

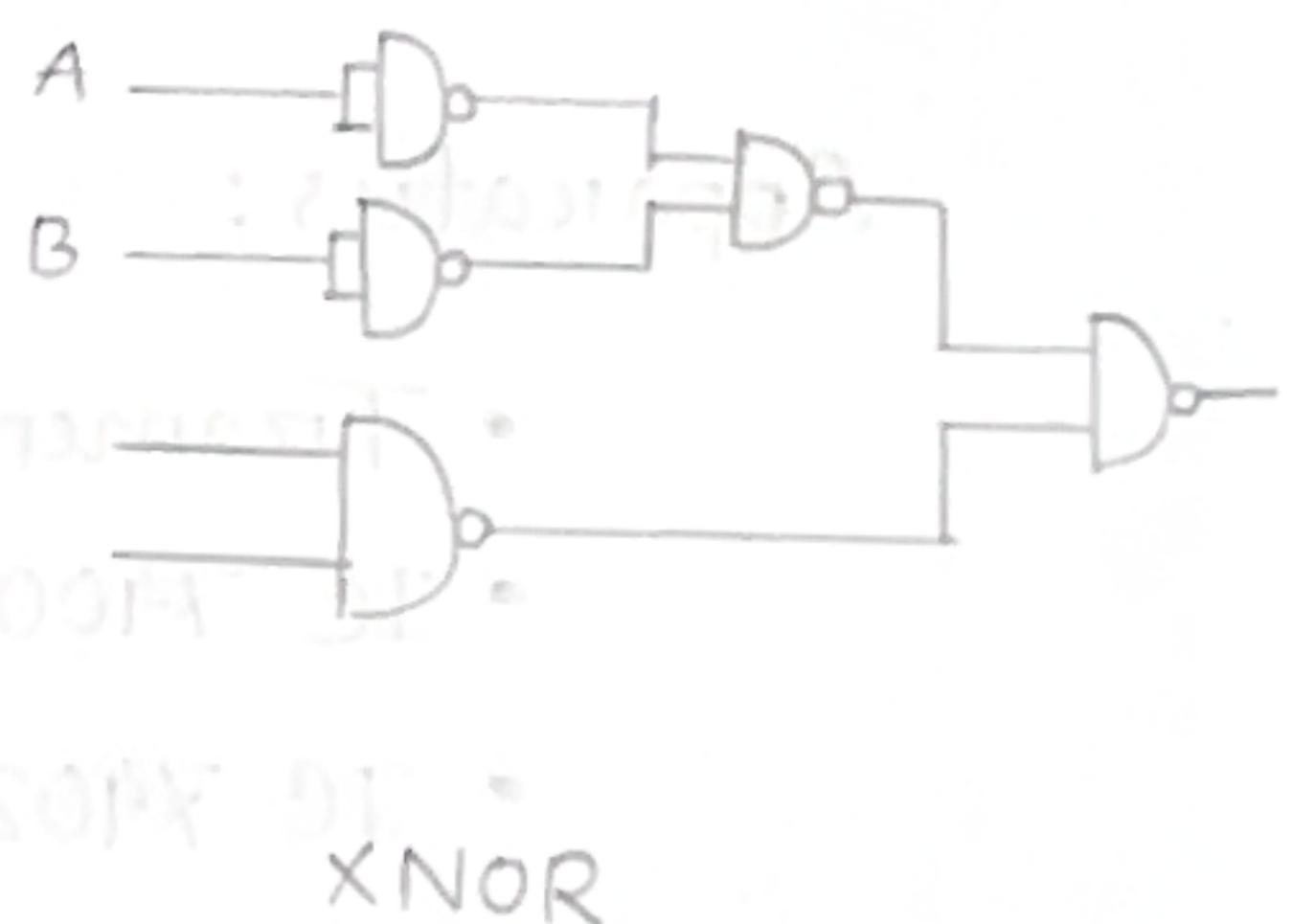
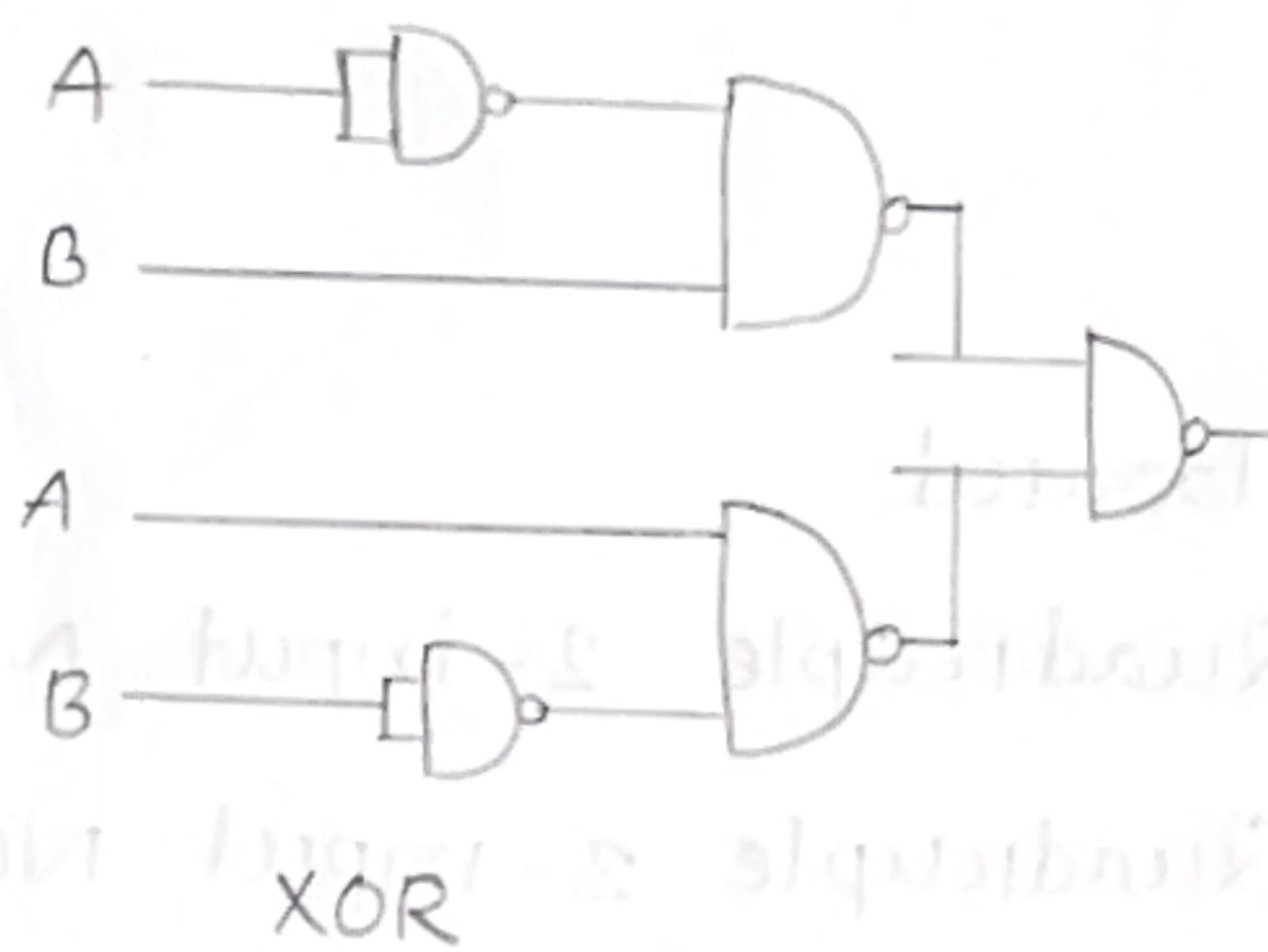


Figure F1: Implementation of XOR and XNOR using NAND gates.

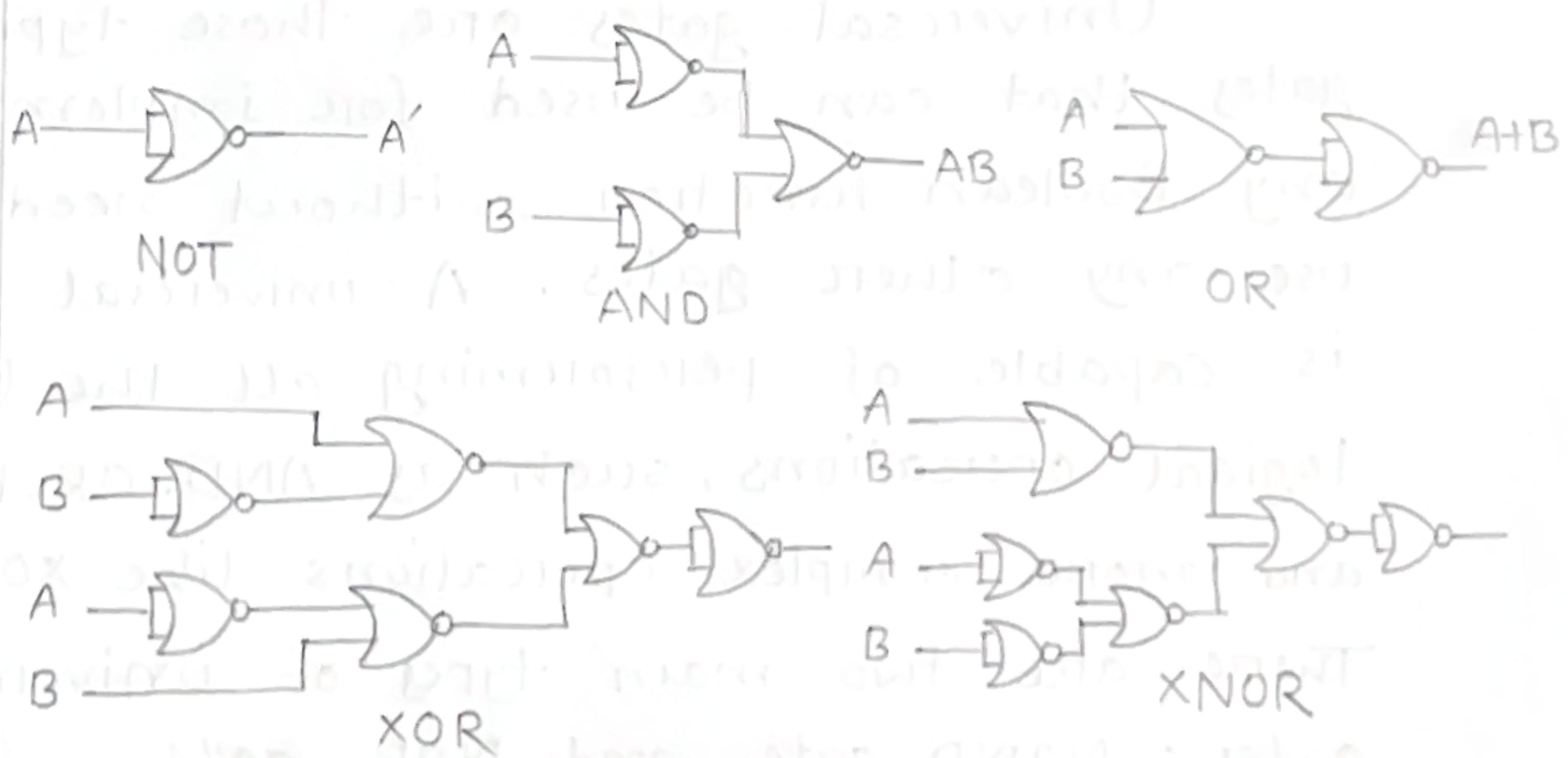


Figure F2 : Implementation of NOT, AND, OR, XOR and XNOR using NOR gates.

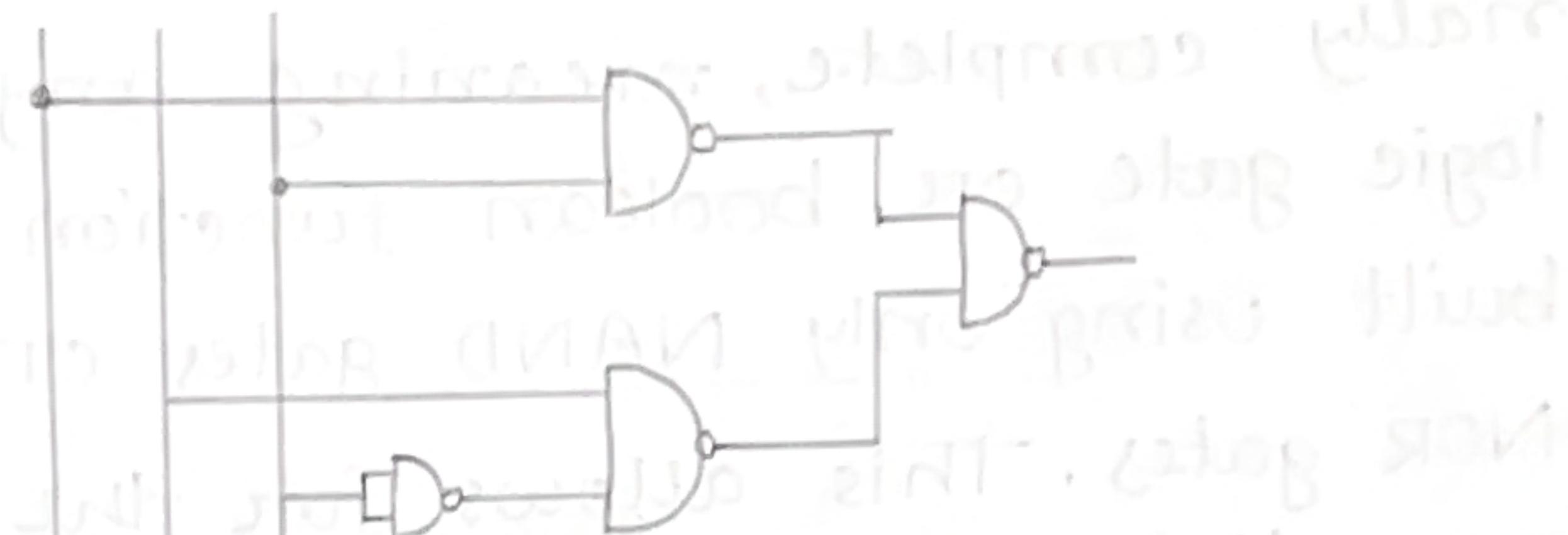


Figure F3 : Implementation of a combinational gate circuit using universal NAND gate.

Data/Truth table :

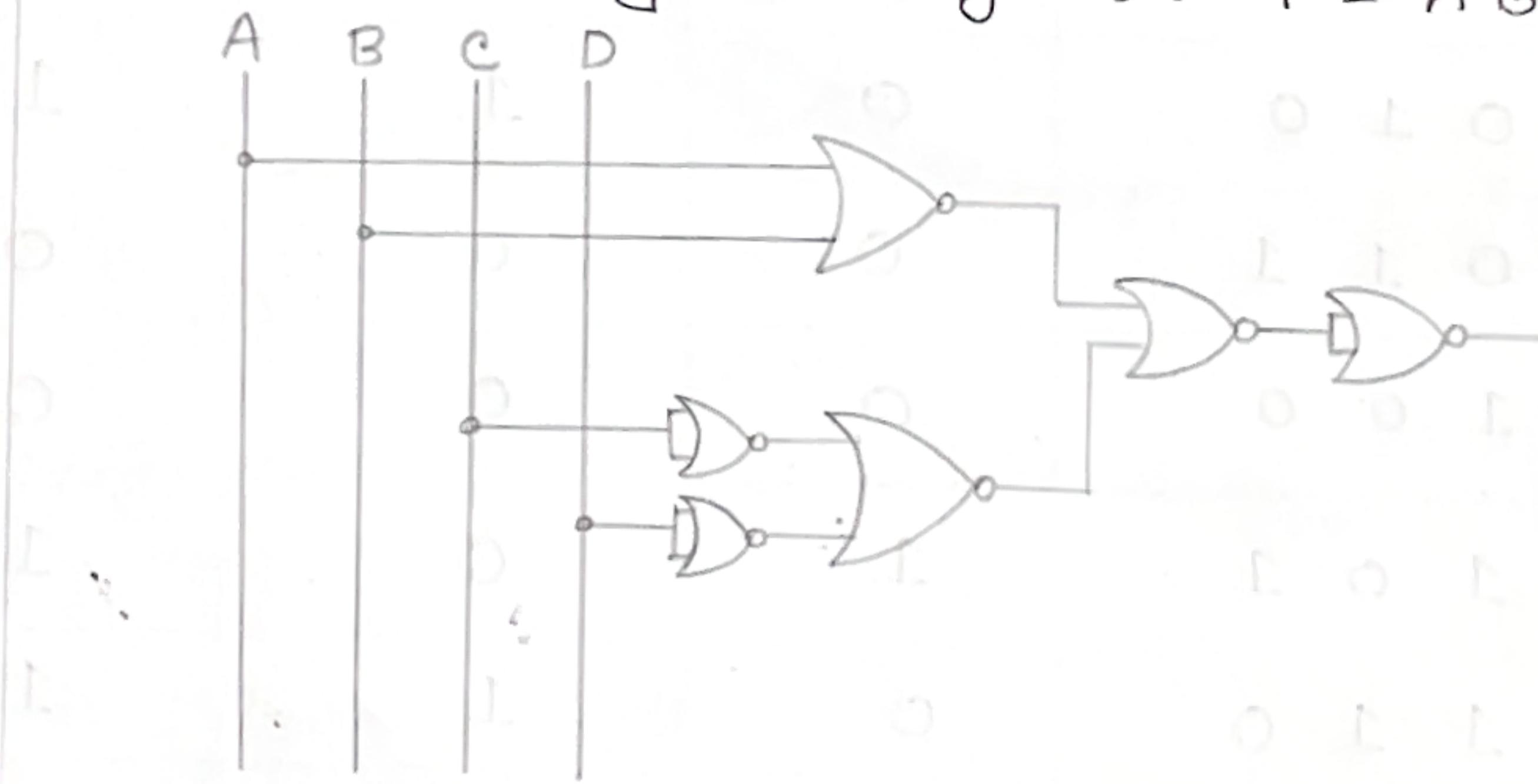
A	B	C	$I_1 = AC$	$I_2 = BC'$	$F = I_1 + I_2$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	1	0	1
1	1	0	0	1	1
1	1	1	1	0	1

Questions and Answers:

1. NAND and NOR gates are called universal gates because they are functionally complete, meaning any other logic gate or boolean function can be built using only NAND gates or only NOR gates. This allows for the design of any digital circuit using a single type of gate, which is efficient for manufacturing and simplifies circuit design.

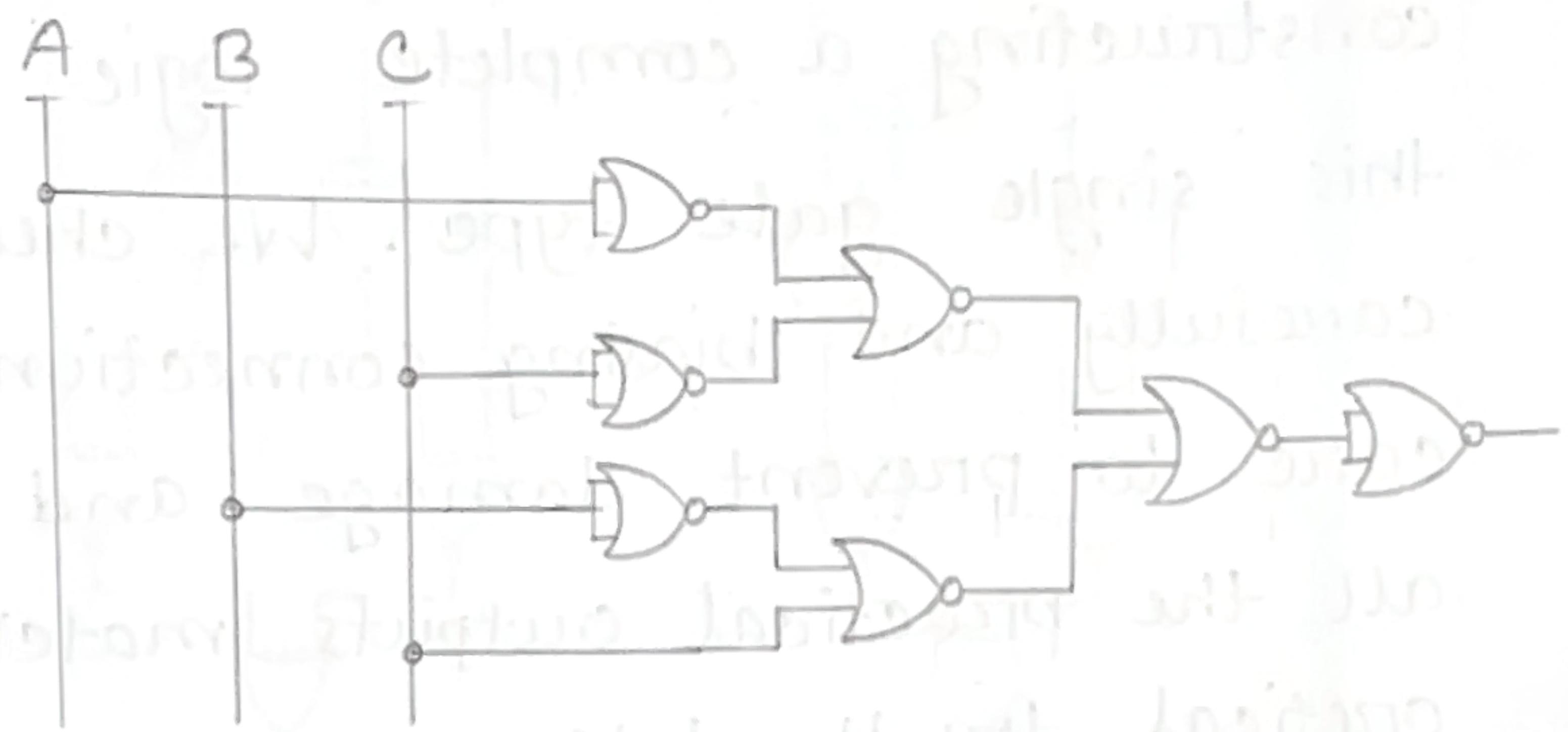
Using only one type of gate is cost efficient because it simplifies manufacturing, reduces component variety and production cost. So its very economical.

2. Only using NOR gates: $F = A'B' + CD$

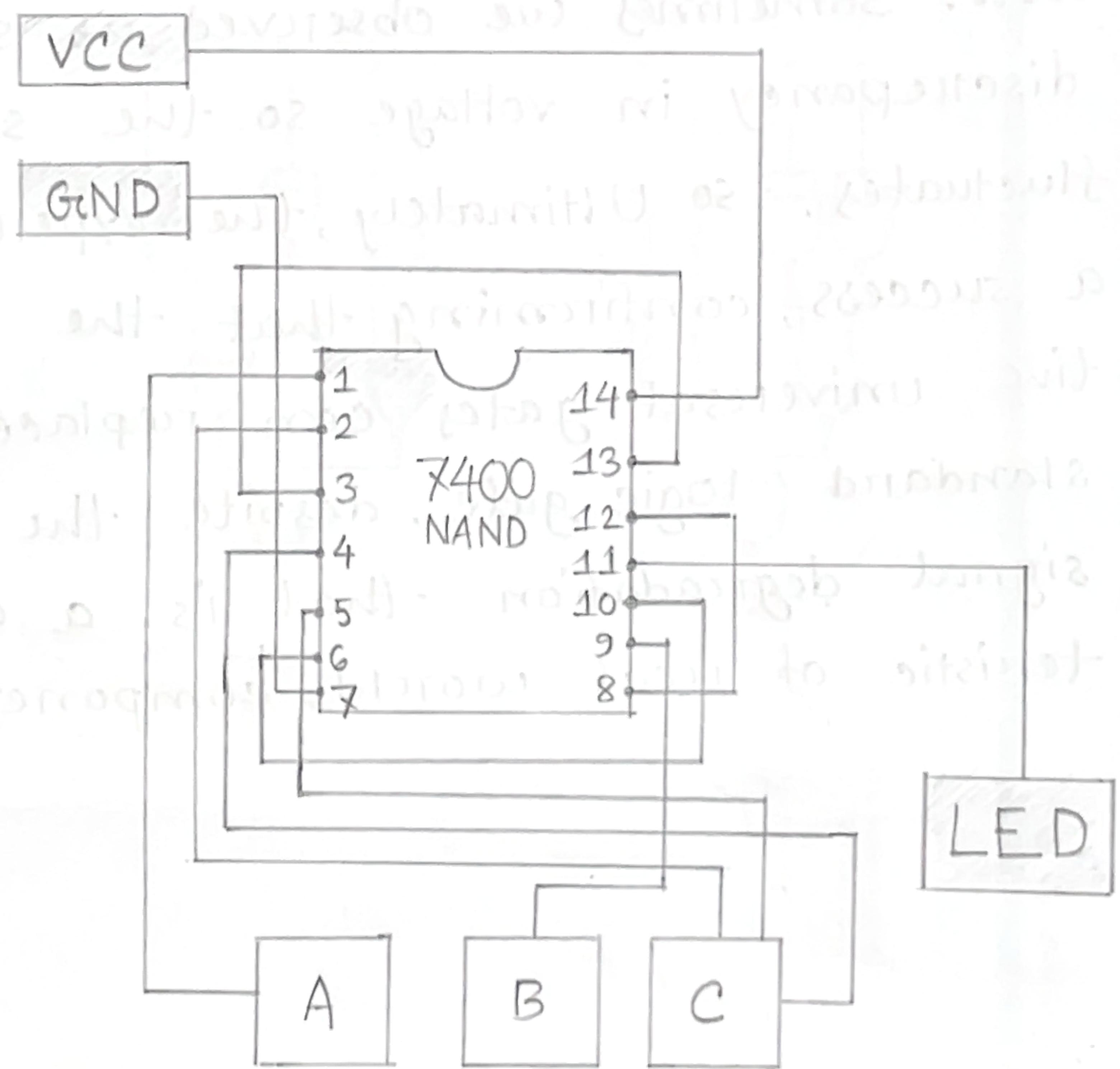


$$F = A'B' + CD$$

3. Converting the circuit in Figure D2 to a minimized NOR gate equivalent circuit.



4. The IC diagram for the circuit in Figure F3 - Part 2.

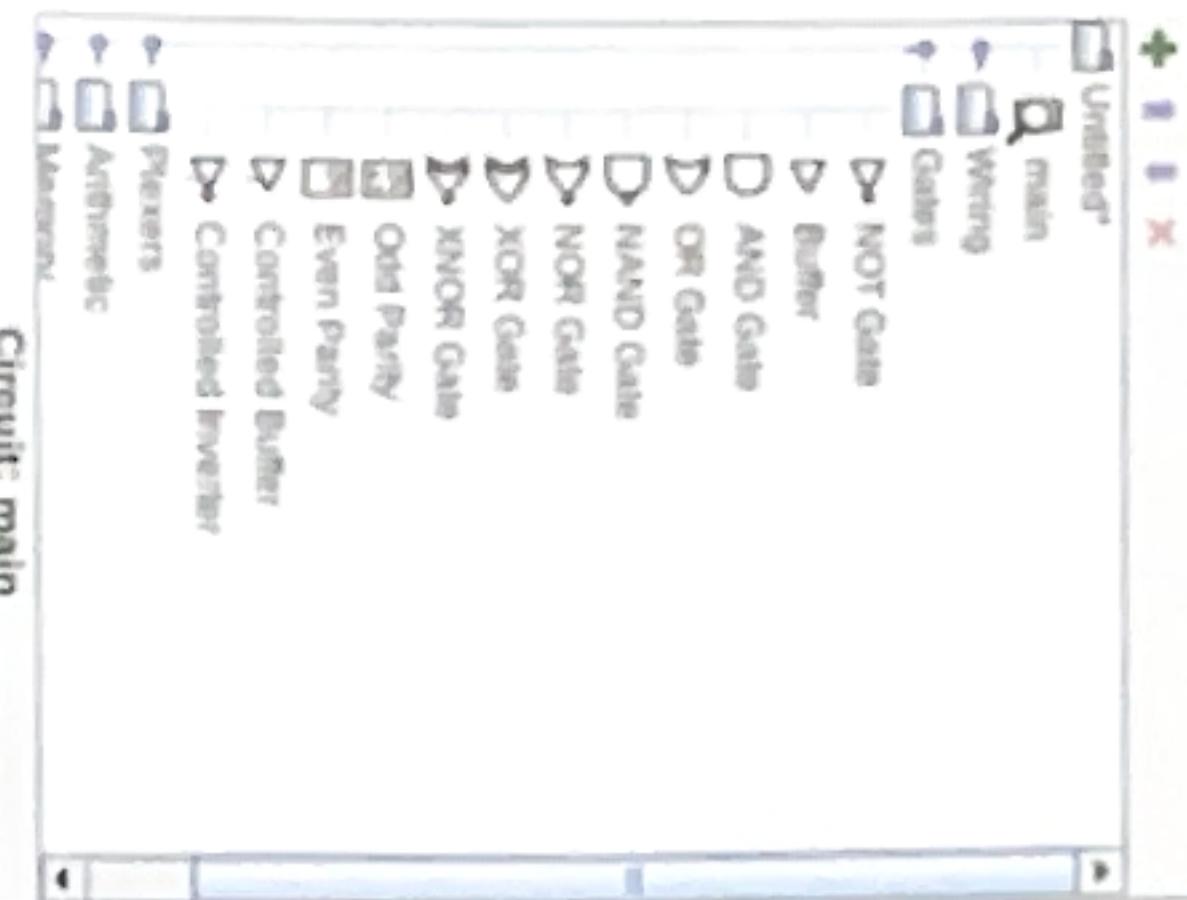
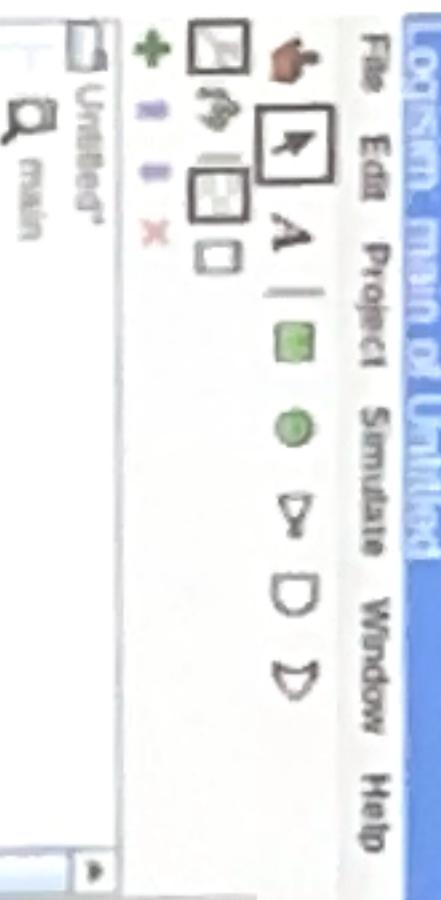


Discussion:

This lab experiment demonstrated the universality of the NAND/NOR gate by constructing a complete logic function using this single gate type. We checked the ICs carefully and biasing connections with great care to prevent damage and confirmed all the practical outputs matched the theoretical truth table, which confirmed the concept of gate universality. Using universal gates simplifies manufacturing and reduces costs. Sometimes we observed a slight discrepancy in voltage so the signal light fluctuates. So ultimately, the experiment was a success, confirming that the cost-effective universal gates can replace any standard logic gate, despite the minor signal degradation that is a characteristic of real world components.

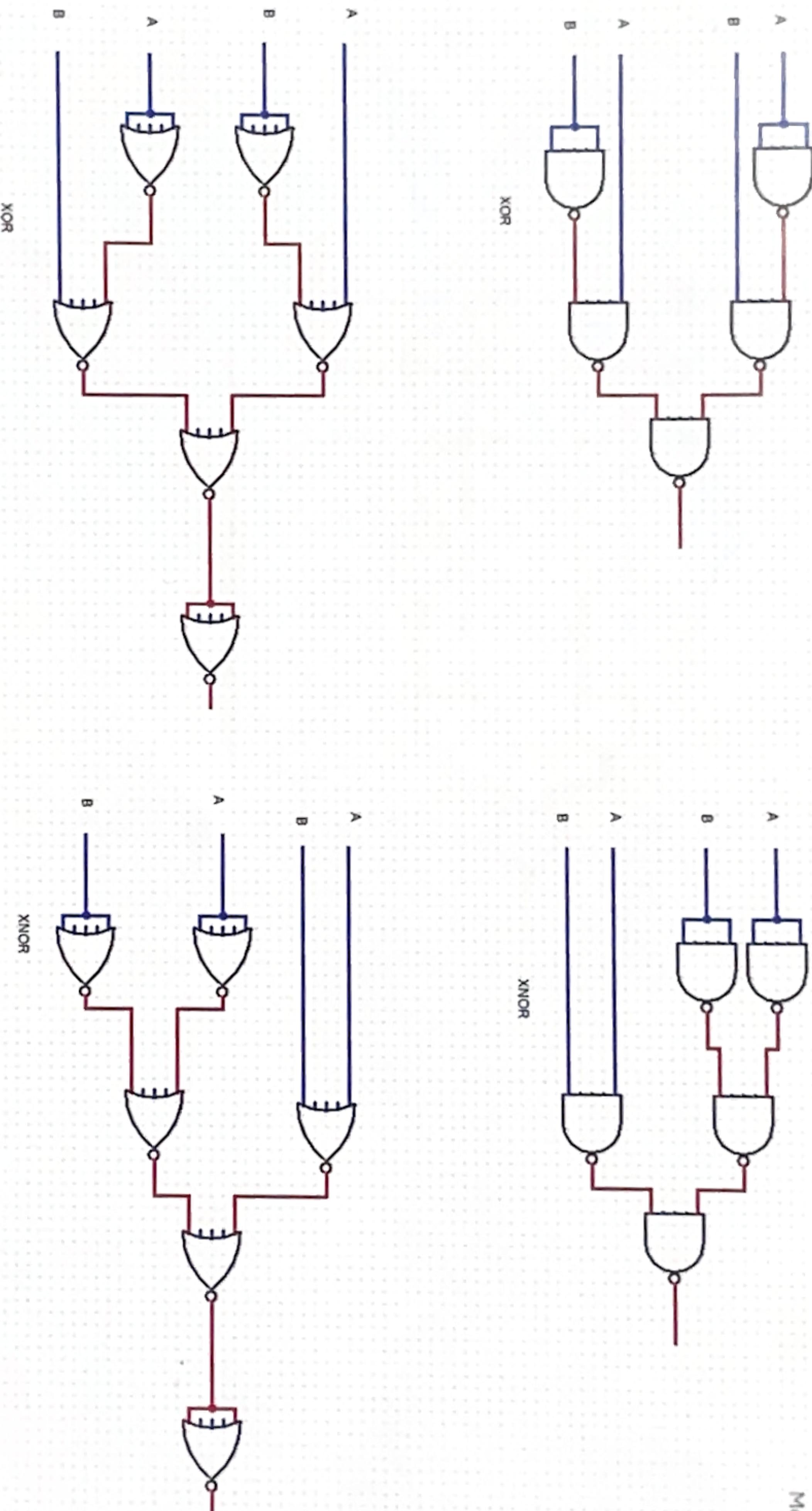
Logisim.app

Report an issue



Circuit: main

Circuit Name	main
Shared Label	
Shared Label Facing	East
Shared Label Font	SansSerif Plain 12



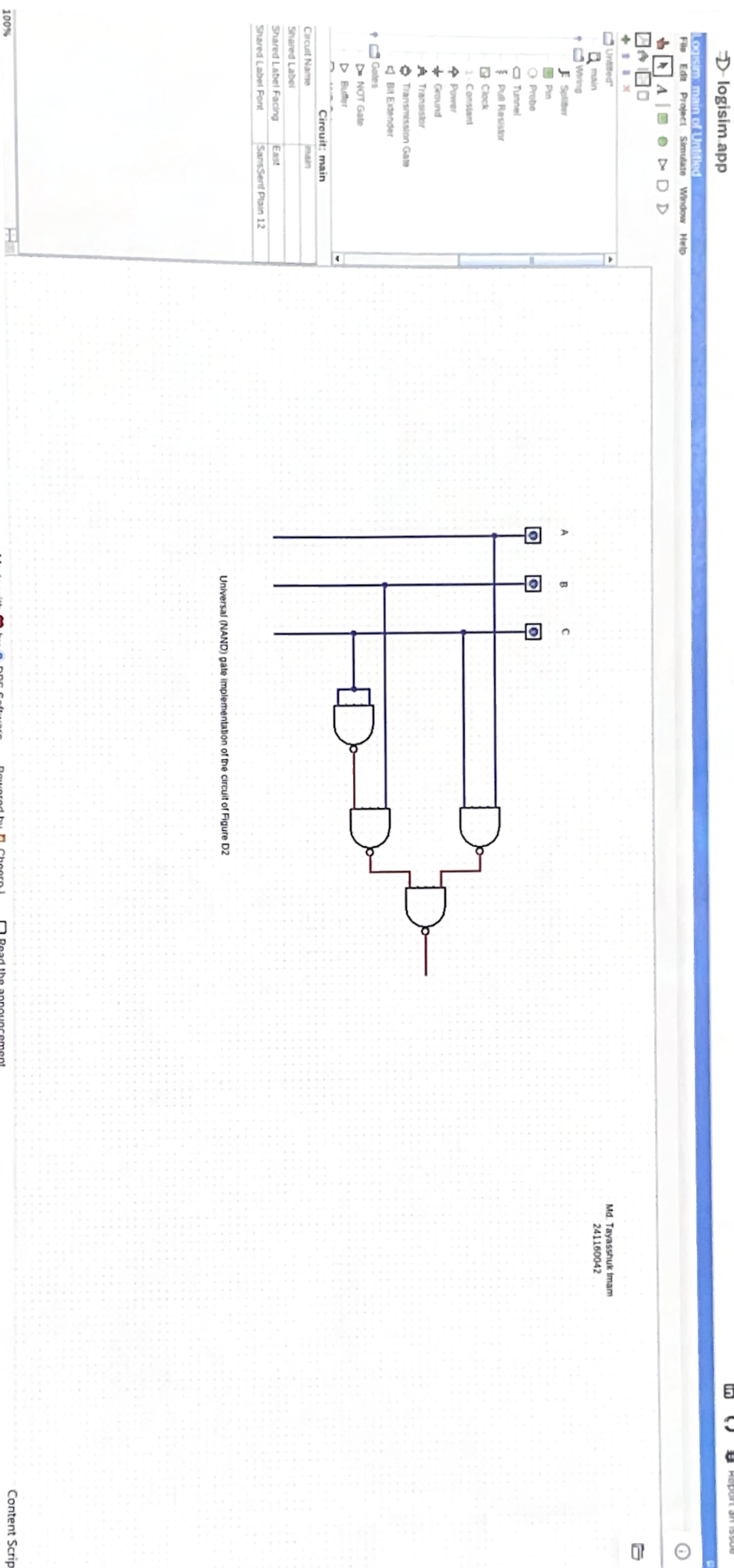
Made with ❤ by DRS Software

Powered by 🚗 CheerpJ

▢ Read the announcement

Content Script

100%



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

E. Experimental Data

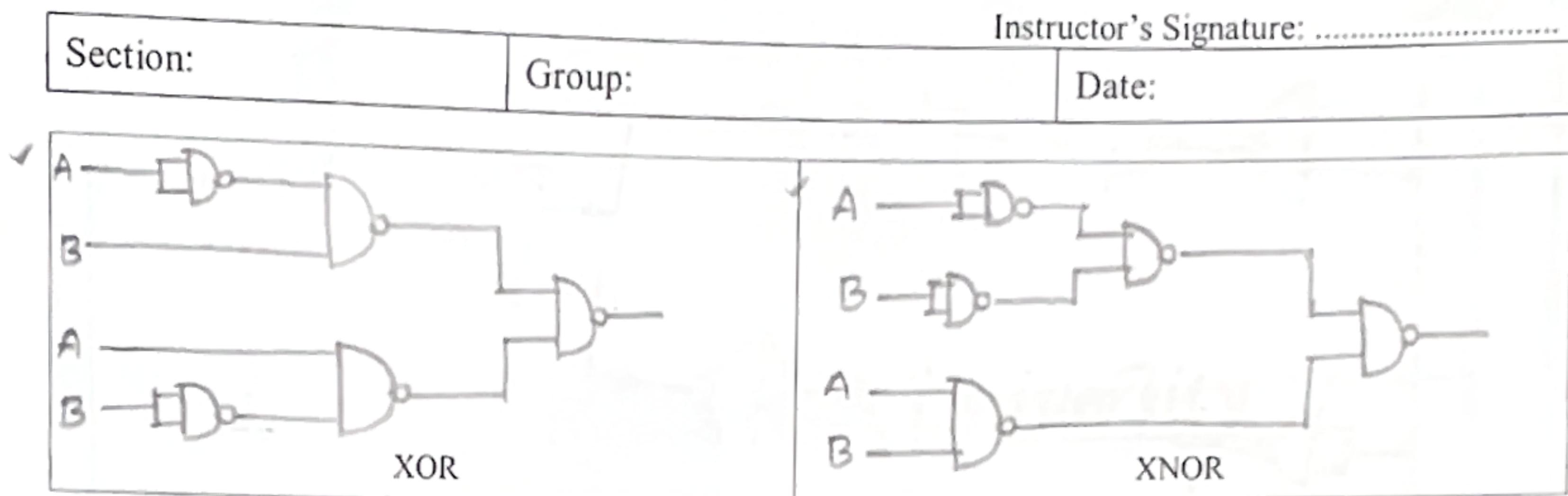


Figure F1: Implementation of XOR and XNOR using NAND gates

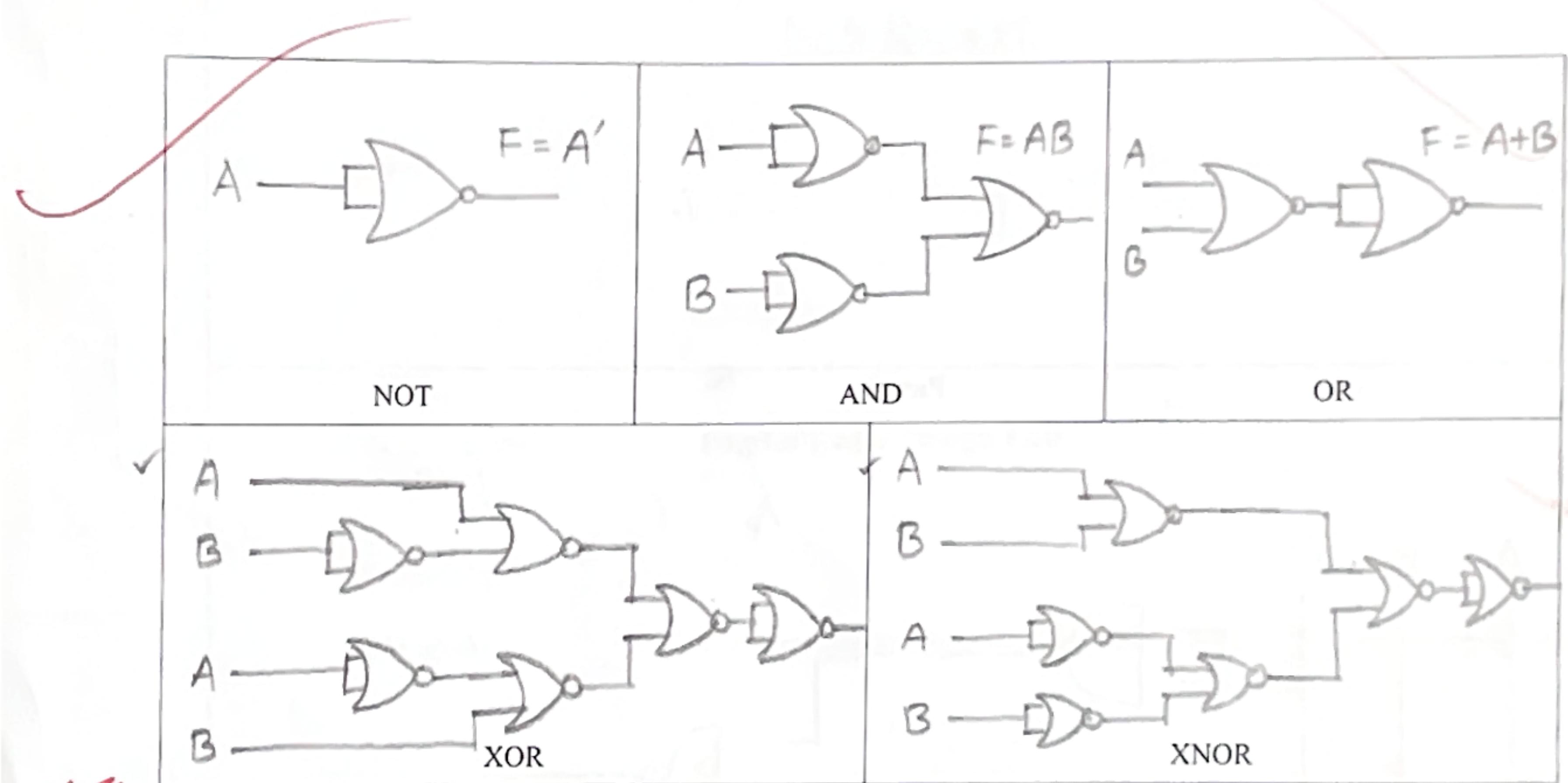


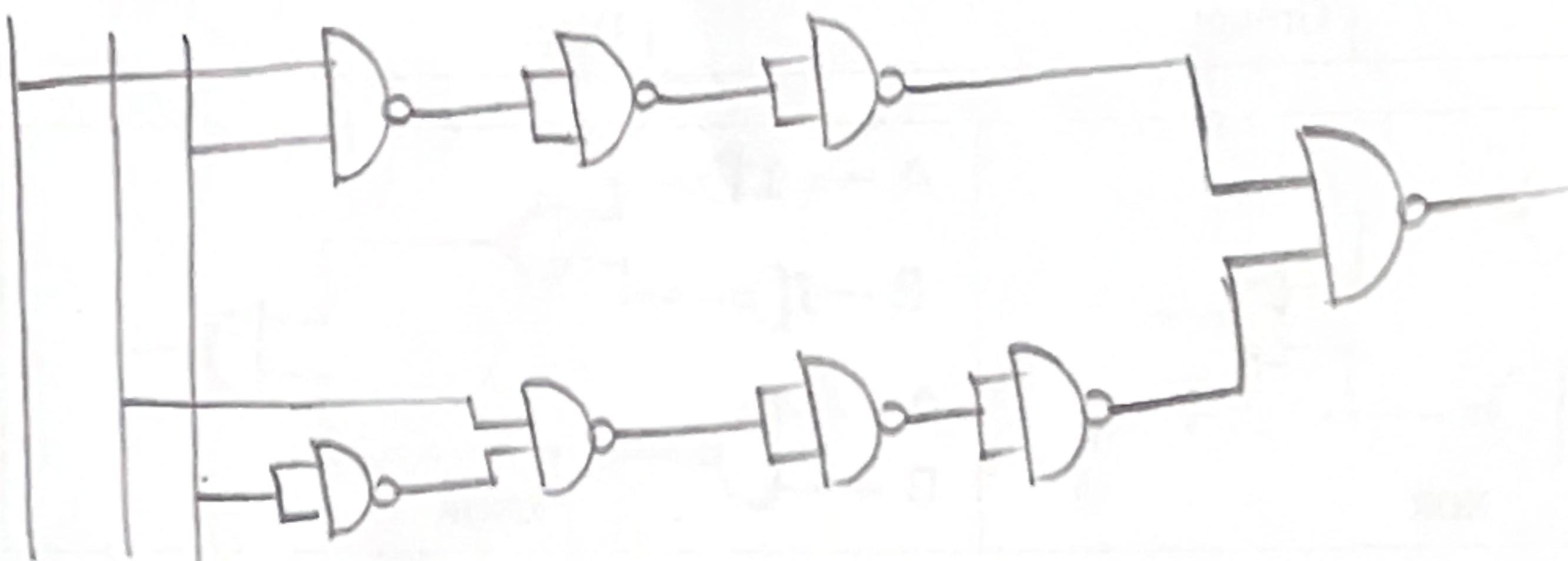
Figure F2: Implementation of NOT, AND, OR, XOR and XNOR using NOR gates

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

Part 1

A B C



Part 2

A B C

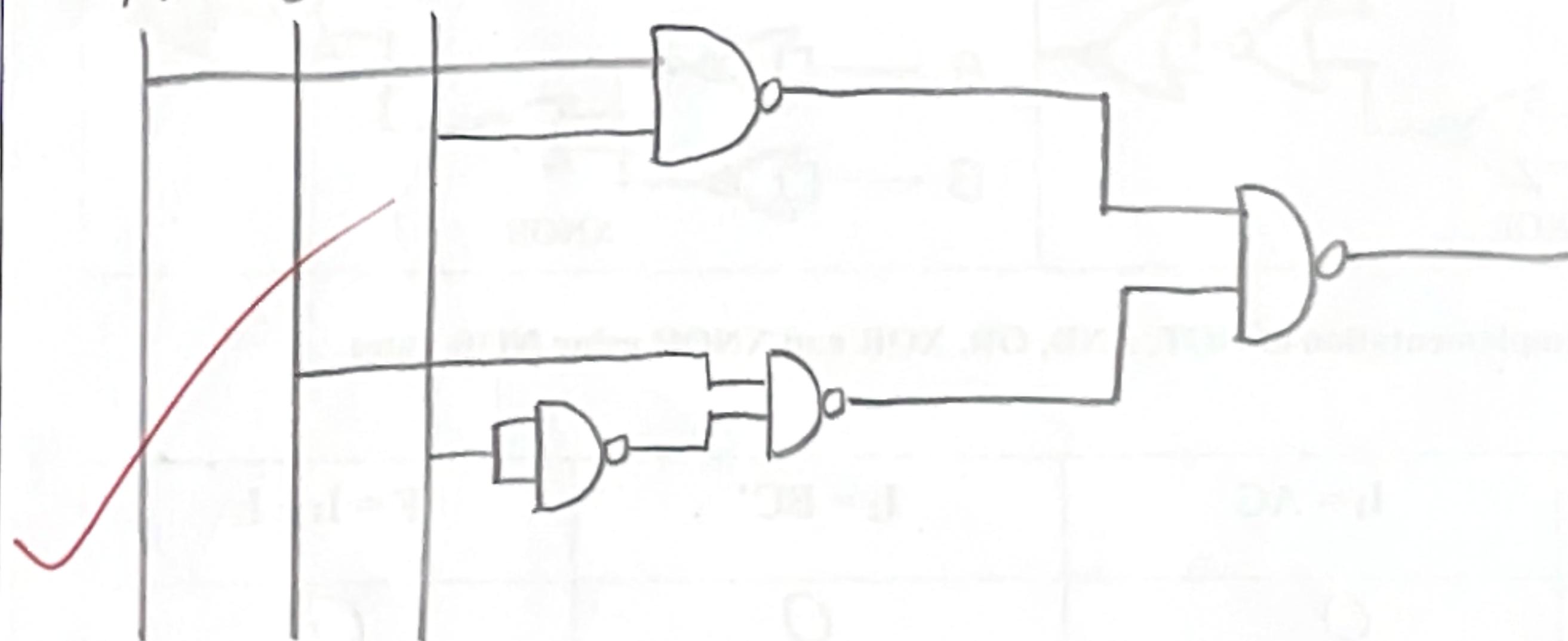


Figure F3: Universal (NAND) gate implementation of the circuit of Figure D2