AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

FACULTY OF SCIENCE & TECHNOLOGY

DIGITAL LOGIC AND CIRCUITS LAB

Summer 2022-2023

Section: F
Group Number: 02

EXPERIMENT NO. 1

NAME OF THE EXPERIMENT

Studying different digital logic gates and designing basic logic gates using Universal gates

Supervised By

SHAHRIYAR MASUD RIZVI

Faculty of Engineering, AIUB

Submitted By:

Name of the Student	ID Number
1. NOKIBUL ARFIN SIAM	21-44793-1
2. AHNAF AHMED	20-42173-1
3. SAIFUL ISLAM	20-42585-1

Date of Submission: **12th June 2023**

Abstract:

This experiment aims to learn about the characteristics of different kinds of universal gates.

Introduction:

A logic gate is a simple switching circuit that determines whether an input pulse can pass through to the output in digital circuits [1]. On the other hand, a universal gate can be used or implemented for any Boolean function without the help of any other gate. NAND and NOR are two universal gates. NOT, OR, AND, XOR, XNOR can be created by those two universal gates.

Theory and Methodology:

In analog signals, information is represented by varying electric pulses, while digital signals translate information into binary format (zero or one), where each bit represents two distinct amplitudes. The main advantage of digital signals is their immunity to imperfections in electronic systems that can degrade analog signals. Codes are often used in information transmission for secrecy or breaking information into manageable pieces. Digital signals offer noise immunity as each component is determined by the presence or absence of a data bit. Digital circuits are cost-effective, easily produced, and consume less power. Integrated circuits (ICs) have lower costs and higher performance due to photolithography printing and compact component size. Logic gates, the building blocks of digital circuits, have binary states of low (0V) and high (5V). Basic logic gates include AND, OR, NOT, NOR, NAND, XOR, and XNOR.

AND operation:

The AND operation produces a high if and only if all the inputs are high. An AND gate can have two or more inputs and performs AND operation or logical multiplication.

Fig1.1: Symbol of AND gate

Truth Table:

Input, A	Input, B	Output, F
0	0	0
0	1	0
1	0	0
1	1	1

OR operation:

The OR operation produces a high output when any of the inputs are high. It has two or more inputs and one output which performs OR operation or logical addition.

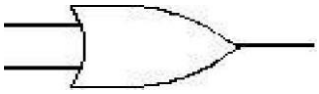


Fig 1.2: Symbol of OR gate

Truth Table:

Input, A	Input, B	Output, F
0	0	0
0	1	1
1	0	1
1	1	1

NAND gate:

The graphic symbol for the NAND gate consists of an AND symbol with a bubble on the output, denoting that a complement operation is performed on the output of the AND gate.

Fig-1: Symbol of NAND gate

Truth Table

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

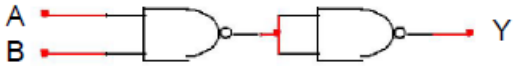
It is possible to construct other gates using NAND gates which are shown in the Experimental procedure part.

Implementing various logic functions using NAND Gates:

1) Implementing NOT gate using NAND gate:

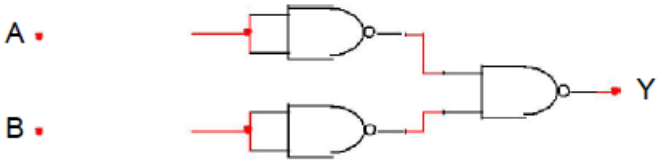
NOT gate using NAND gate

2) Implementing AND gate using NAND gate:

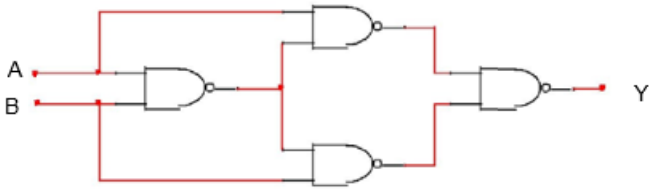


AND gate using NAND gates

3) Implementing OR gate using NAND gate:

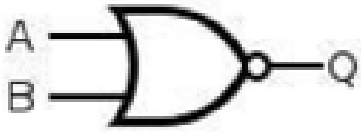


4) Implementing XOR gate using NAND gate:



NOR gate:

It's the abbreviation of NOT and OR. The output of a NOR gate is LOW whenever one or more inputs are HIGH.

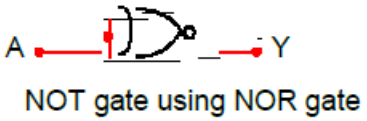


Truth Table

A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

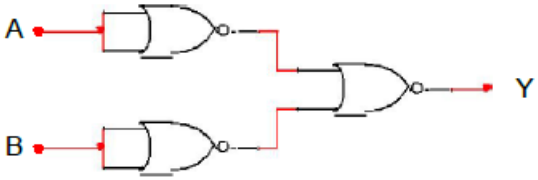
Implementing various logic functions using NOR Gates:

1) Implementing NOT gate using NOR gate:



NOT gate using NOR gate

2) Implementing AND gate using NOR gate:

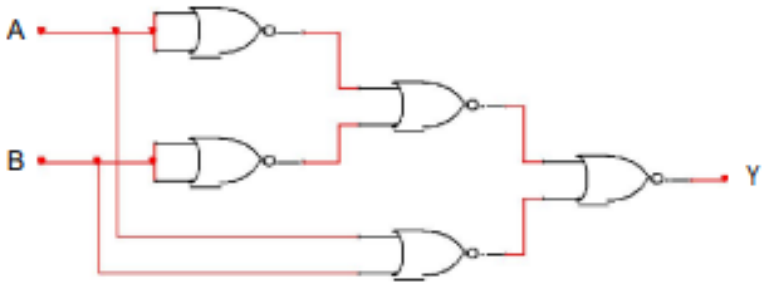


AND gate using NOR gates

3) Implementing OR gate using NOR gate:

OR gate using NOR gates

4) Implementing XOR gate using NOR gate:



XOR gate using NOR gates

XOR operation:

The XOR (exclusive OR) gate acts in the same way as the logical "either/or" .The output is high if either, but not both, of the inputs are high. The output is low if both inputs are low or if both inputs are high. Another way of looking at this circuit

is to observe that the output is 1 if the inputs are different, but 0 if the inputs are the same.

Fig 1.6: Symbol of XOR gate

Truth Table:

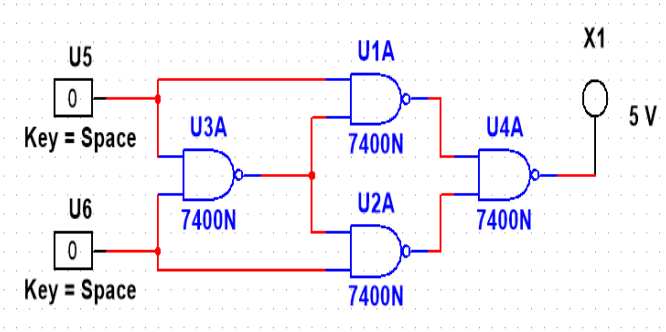
Input, A	Input, B	Output, F
0	0	0
0	1	1
1	0	1
1	1	0

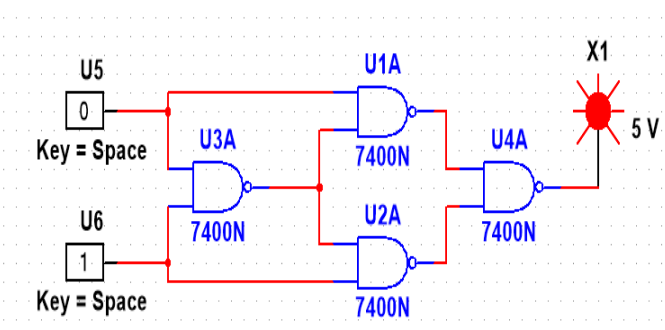
Apparatus:

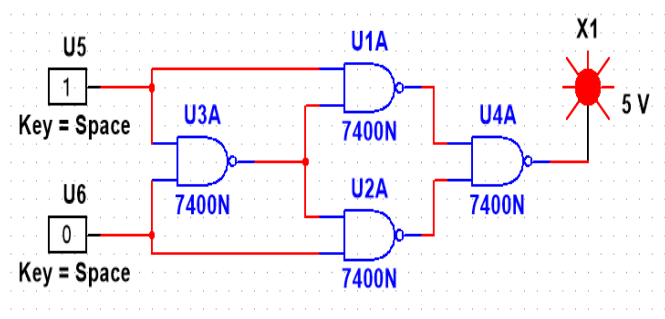
1. Digital trainer board.
2. Integrated Circuits (ICs).
3. Power supply.
4. Connecting wires

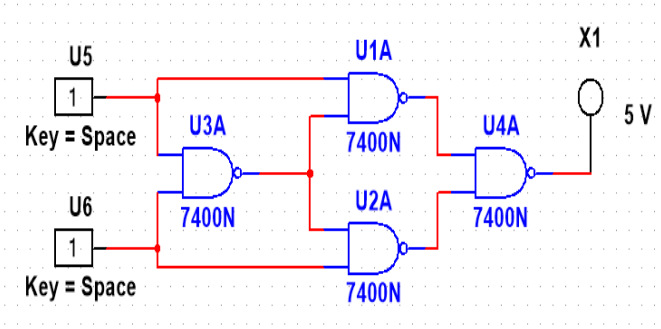
Simulation:

1. **Implementation of XOR gate using NAND gate:**

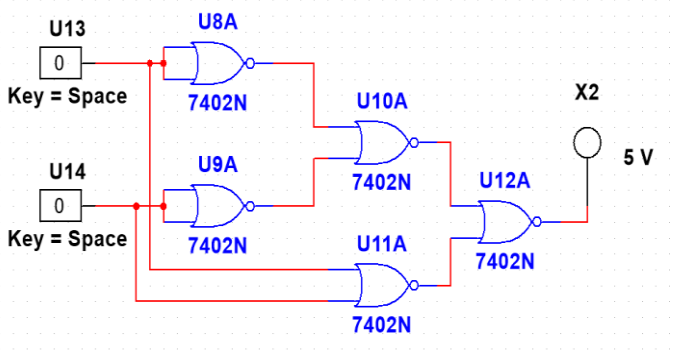


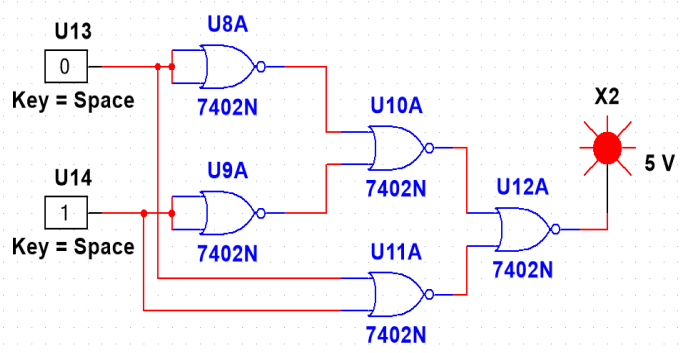


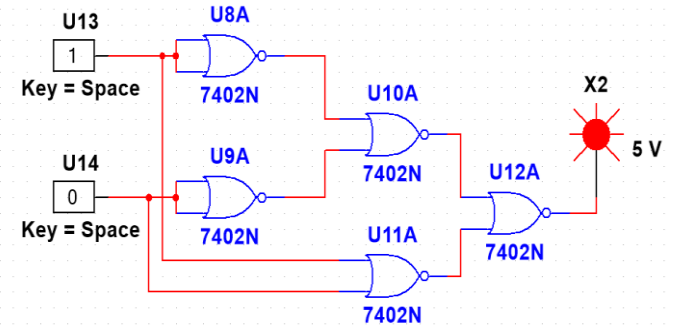


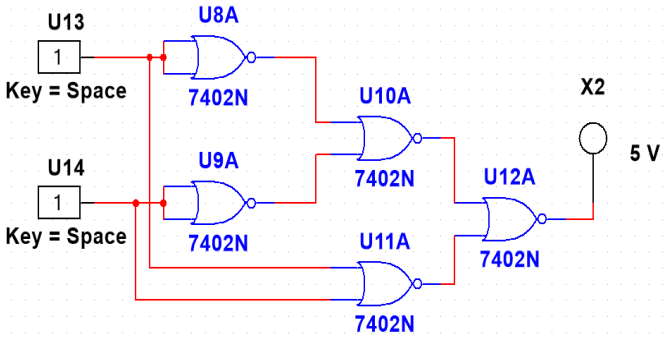


2. Implementation of XOR gate using NOR gate:





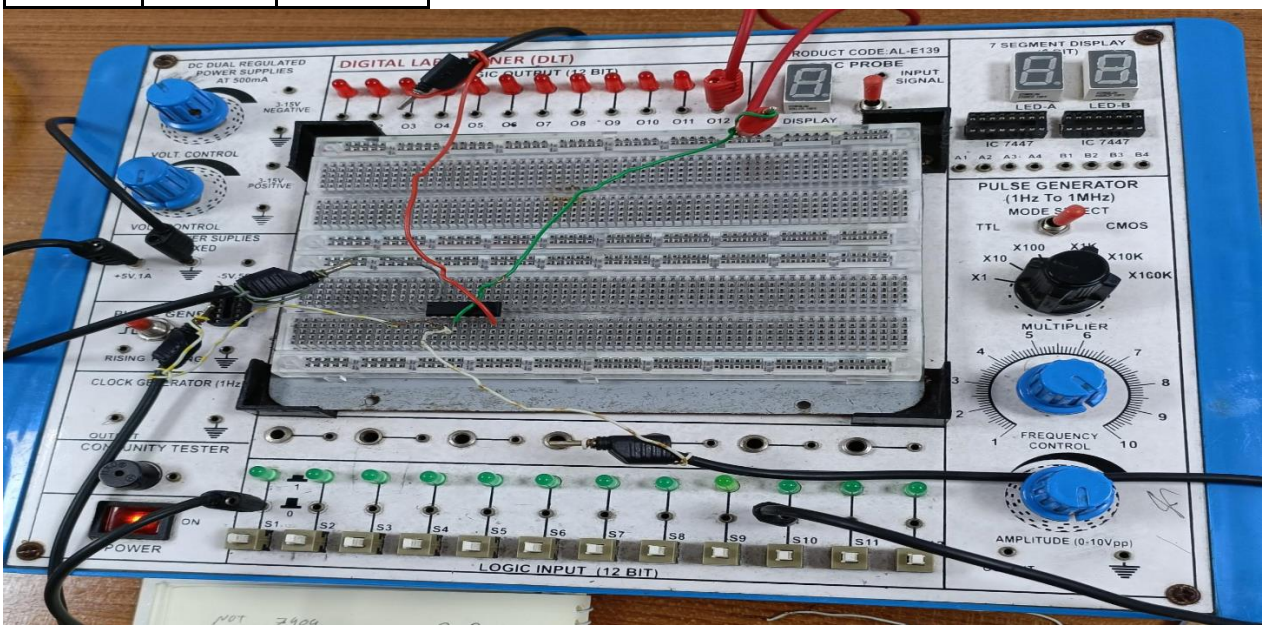




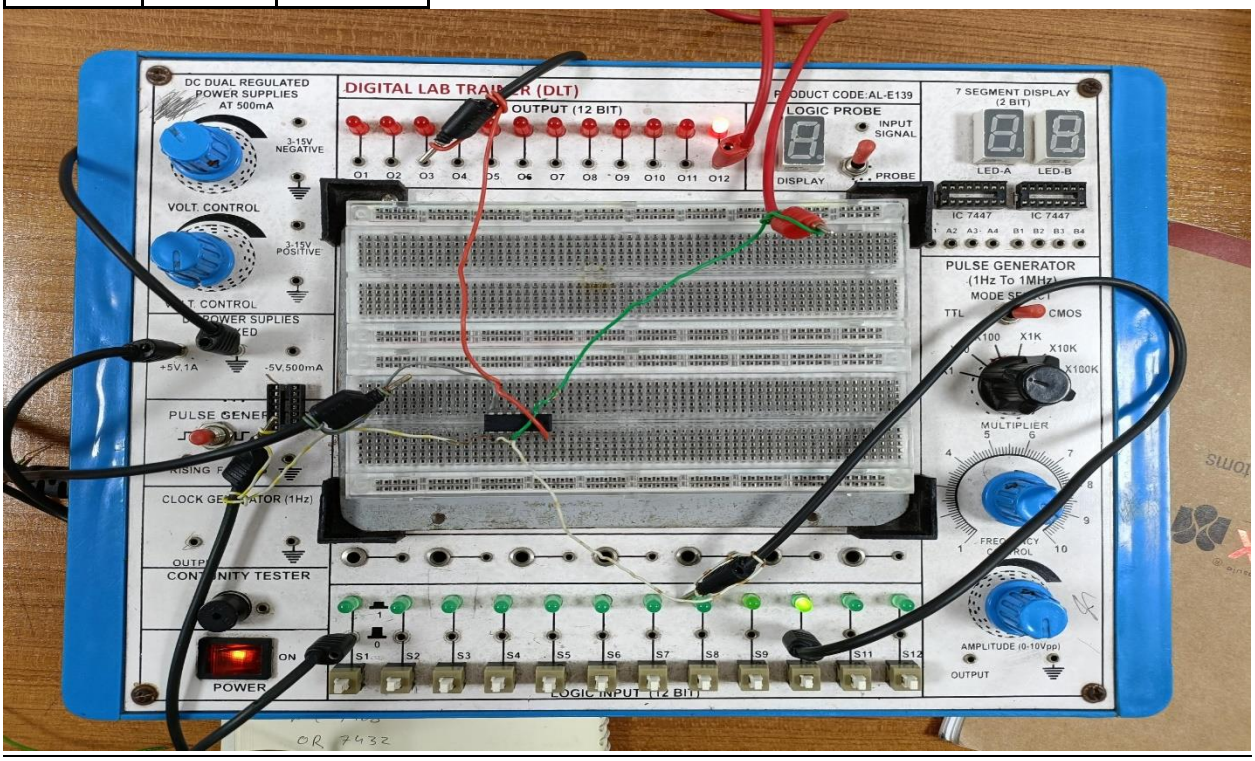
Hardware Implementation:

1. Implementation OR gate:

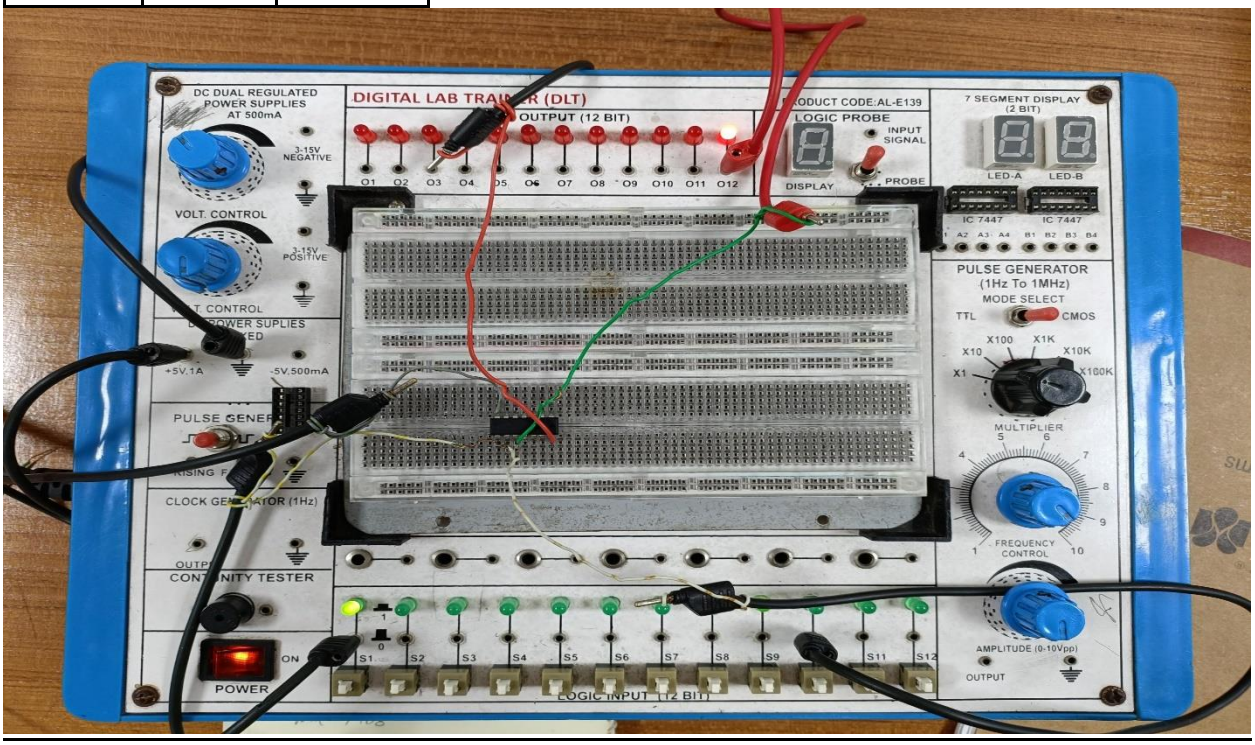
Input, A	Input, B	Output, F
0	0	0



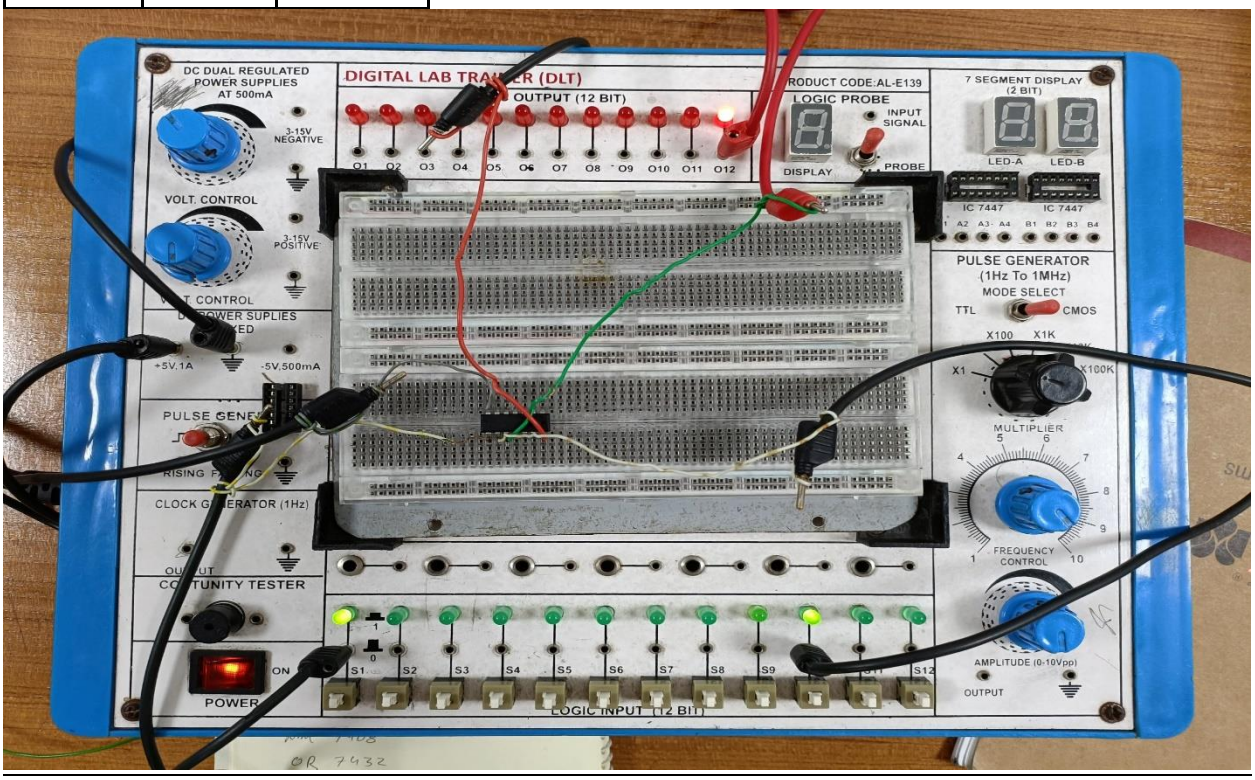
Input, A	Input, B	Output, F
0	1	1



Input, A	Input, B	Output, F
1	0	1

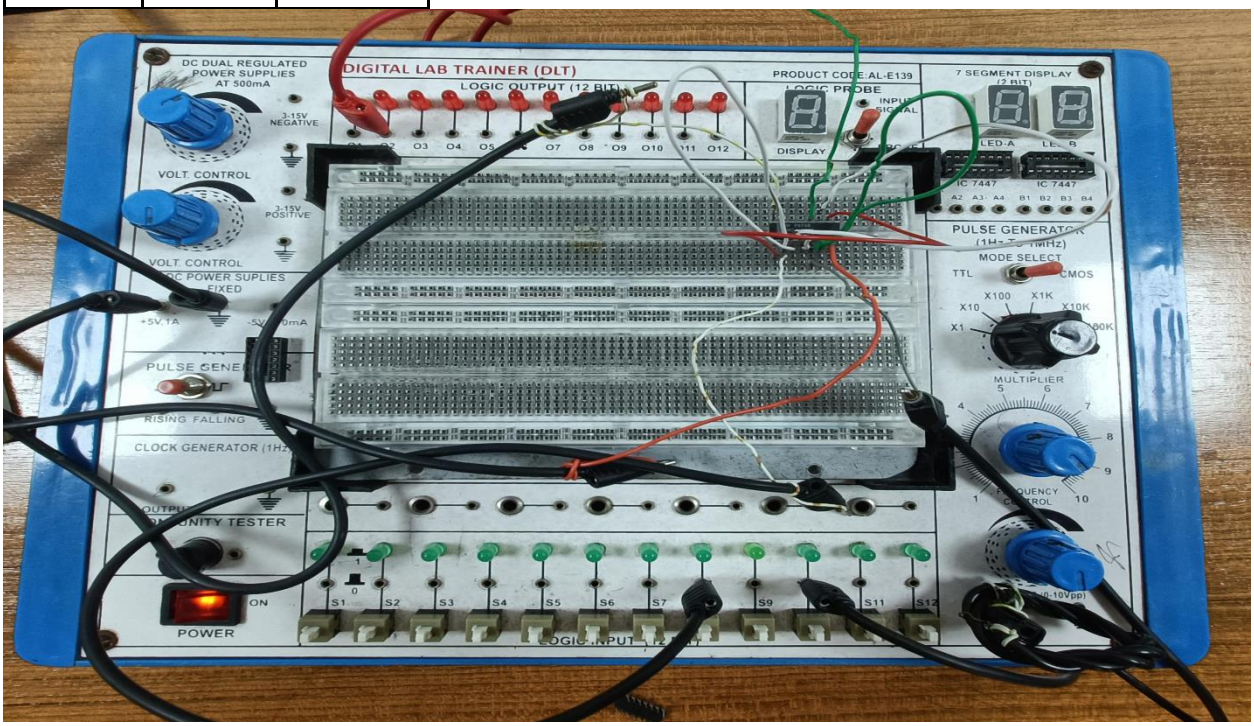


Input, A	Input, B	Output, F
1	1	1

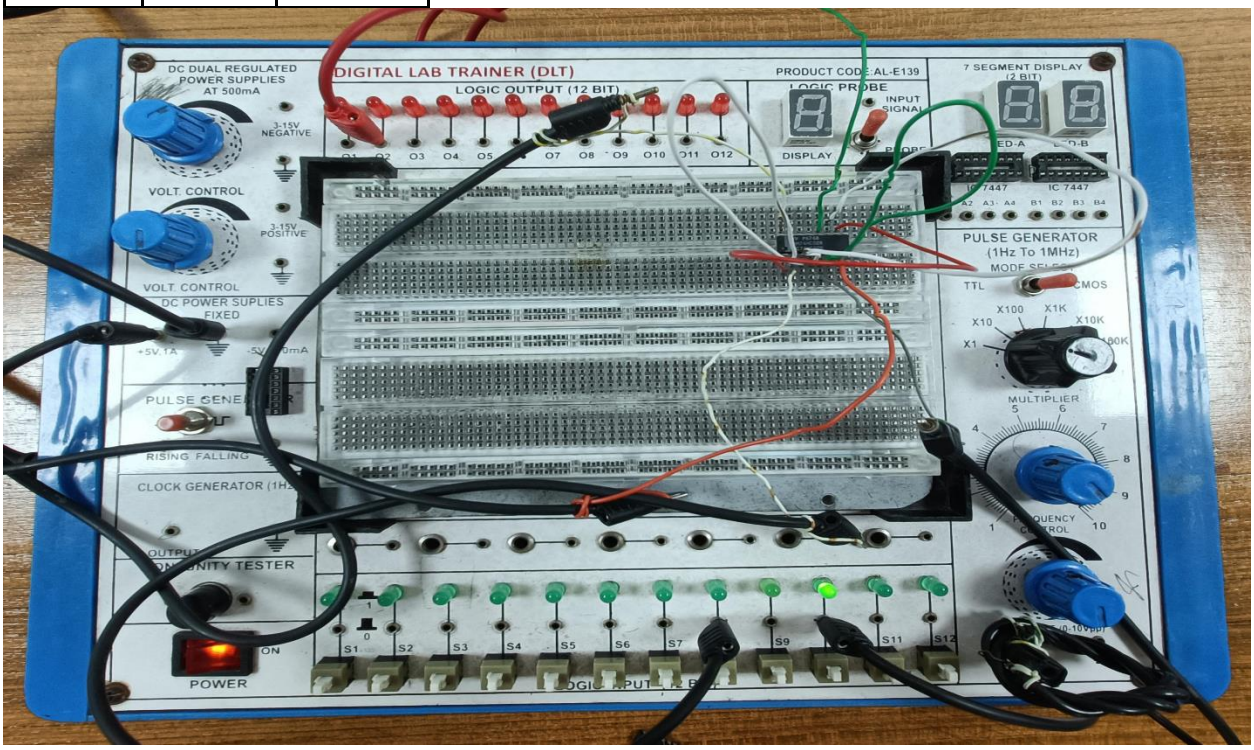


2. Implementation of AND gate with NOR gate:

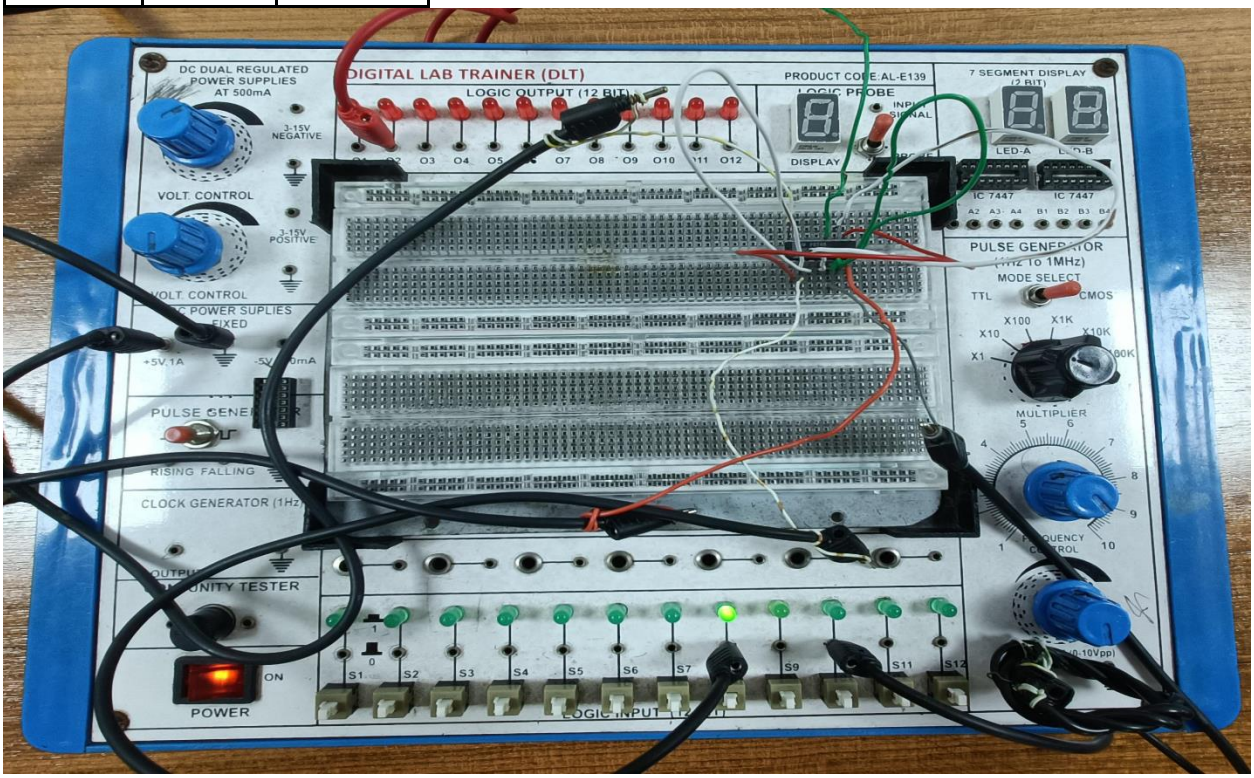
Input, A	Input, B	Output, F
0	0	0



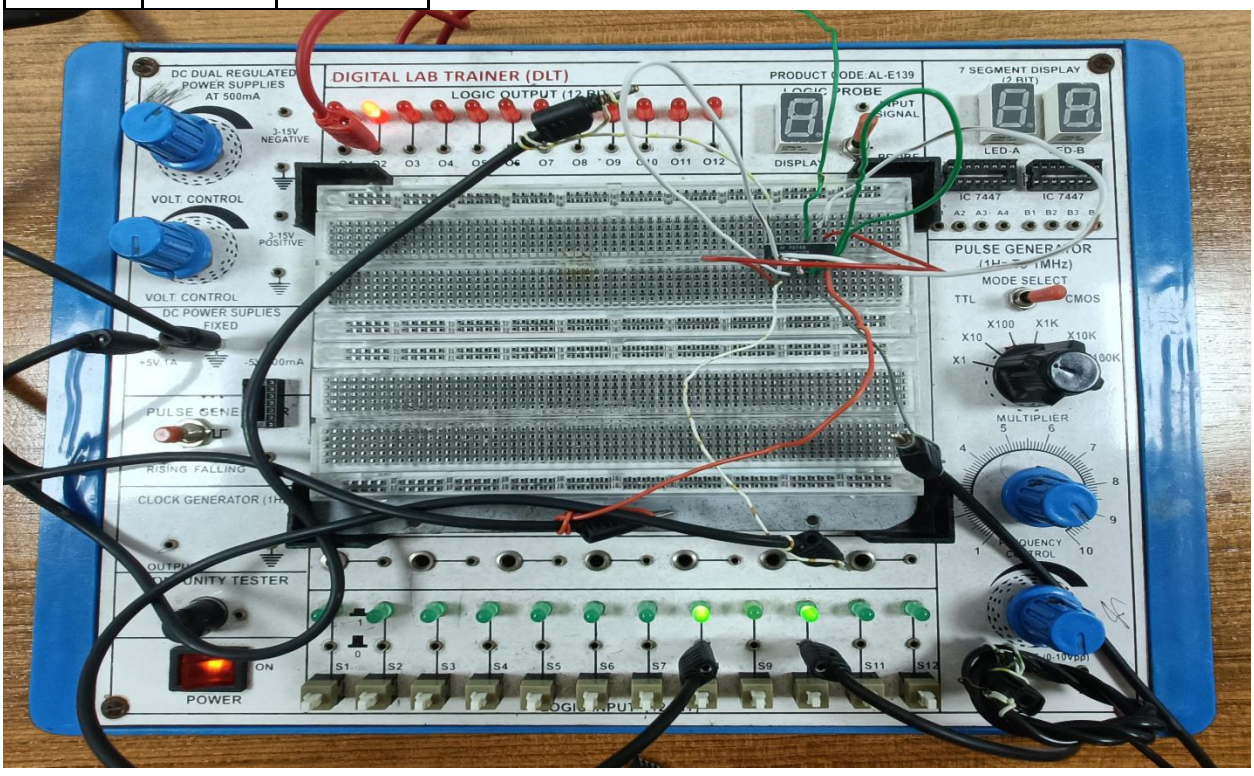
Input, A	Input, B	Output, F
0	1	0



Input, A	Input, B	Output, F
1	0	0

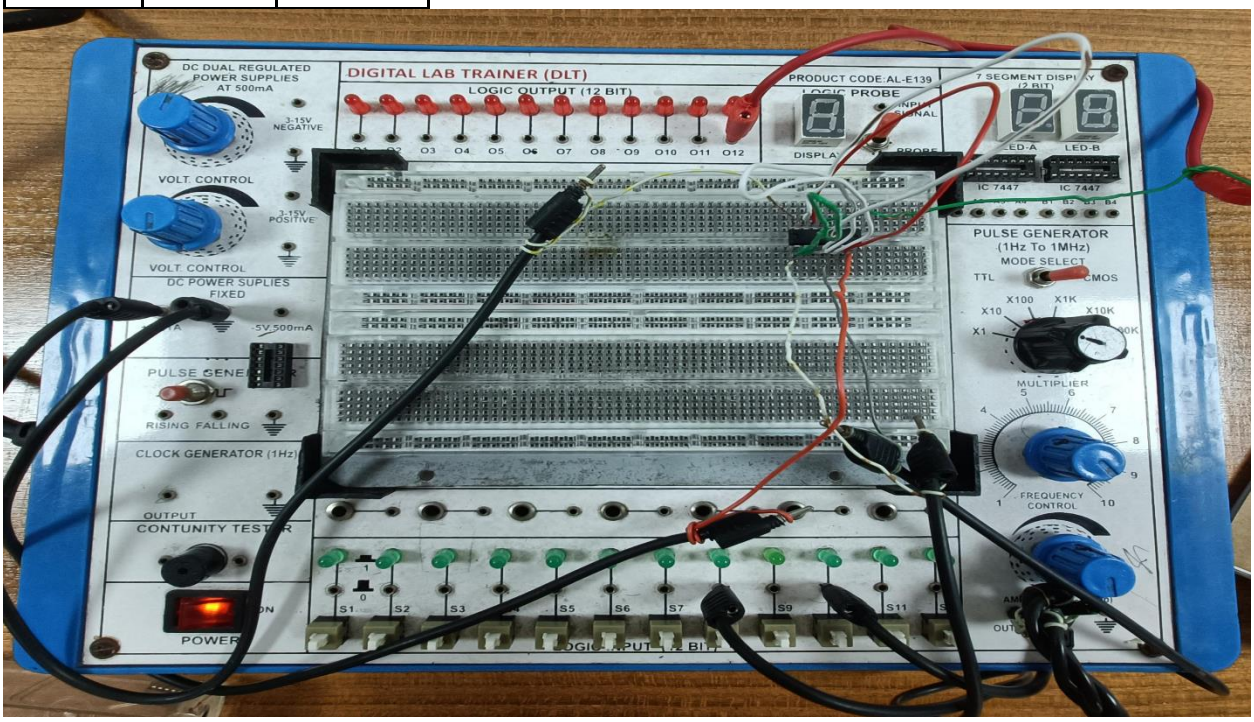


Input, A	Input, B	Output, F
1	1	1

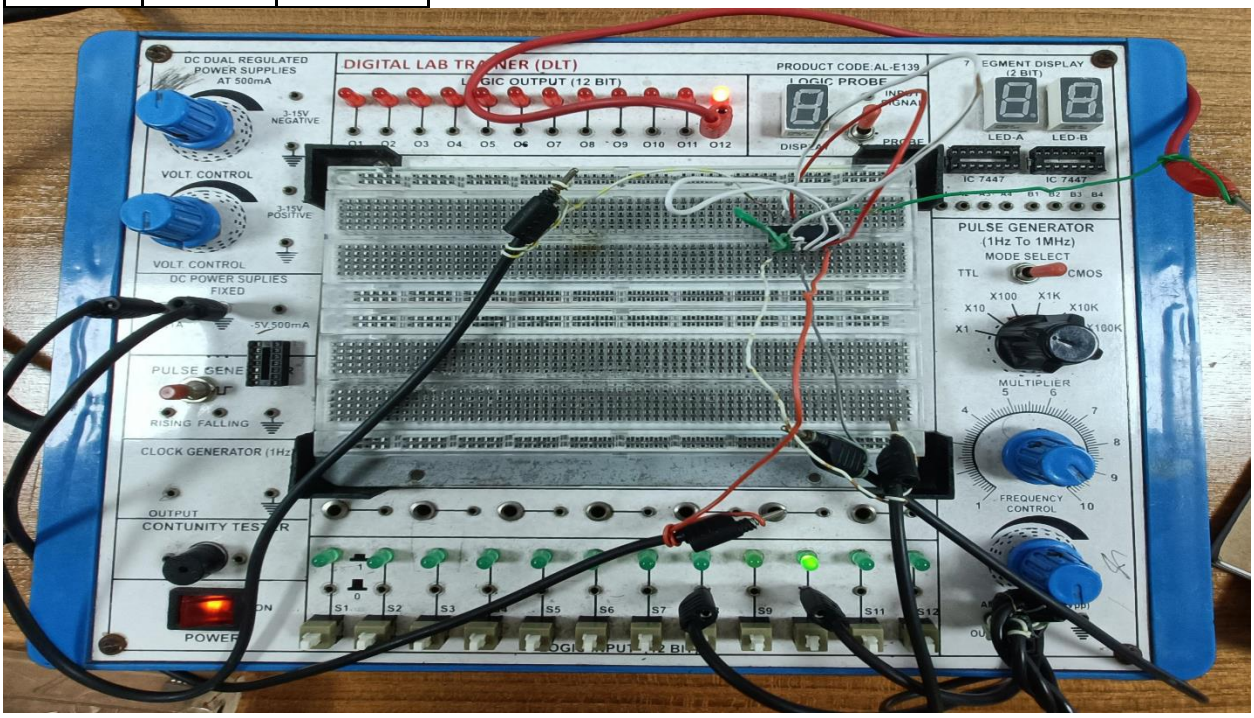


3. Implementation of OR gate with NAND gate:

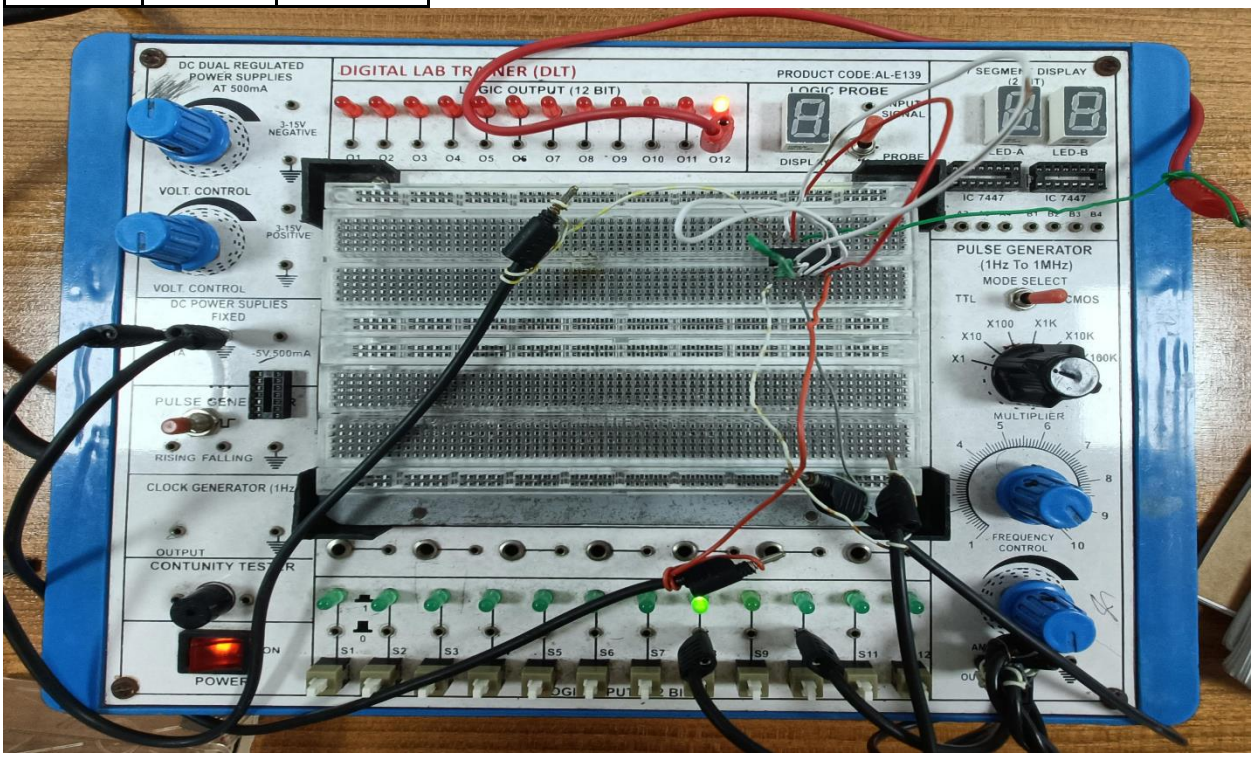
Input, A	Input, B	Output, F
0	0	0



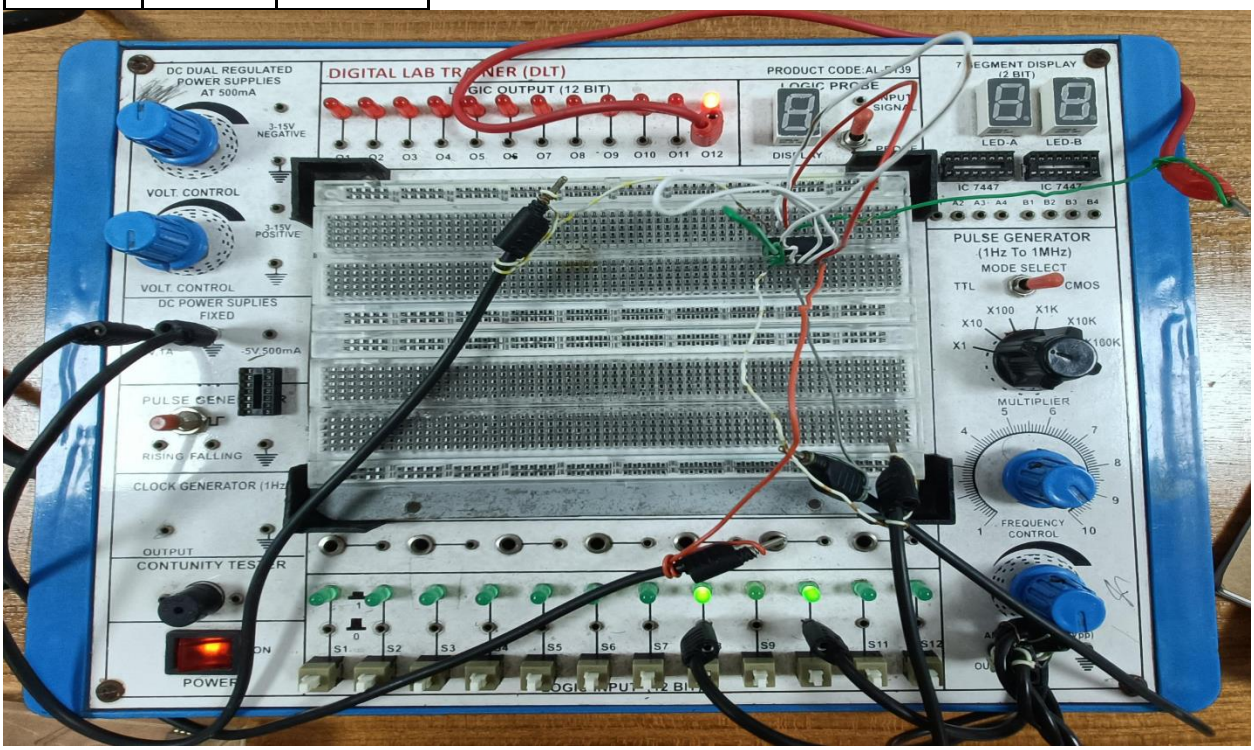
Input, A	Input, B	Output, F
0	1	1



Input, A	Input, B	Output, F
1	0	1



Input, A	Input, B	Output, F
1	1	1



Results: The Simulation results matched all our theoretical truth table findings.

Report Questions:

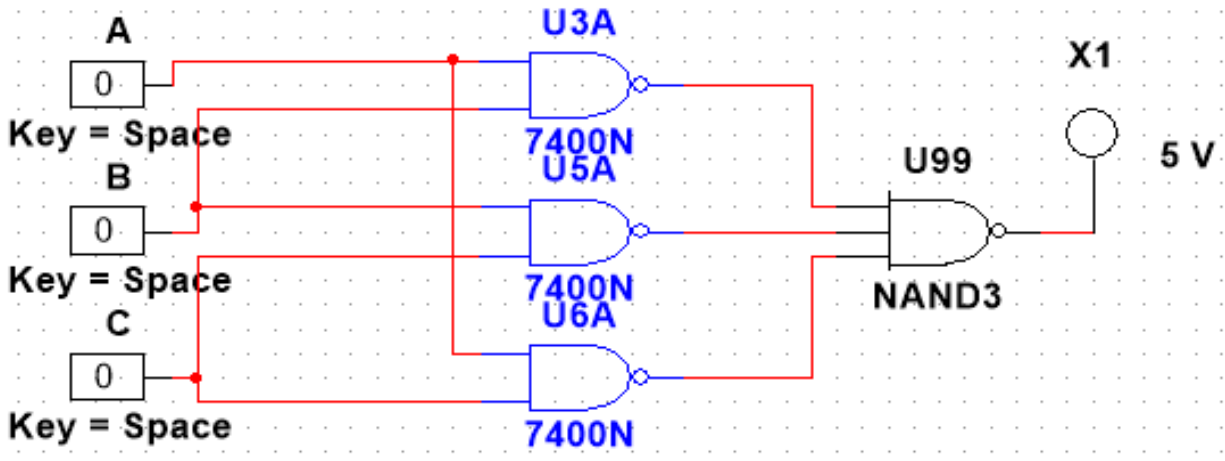
1) What do you mean by the universal gate?
Ans: A universal gate is a gate that can implement any Boolean function without the need to use any other gate type. The NAND and NOR gates are universal gates. This is advantageous in practice since NAND and NOR gates are economical and easier to fabricate, and basic gates are used in all IC digital logic families.

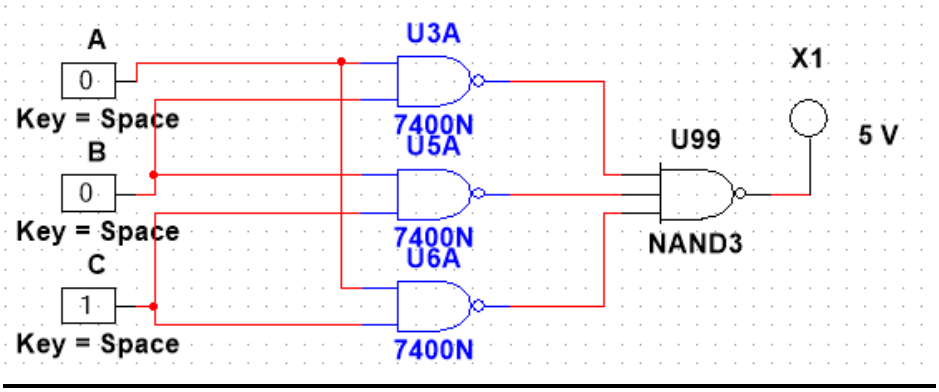
2) What are the ICs required in this experiment?
Ans: In this experiment required ICs are 7404 IC for NOT gate, 7408 IC for AND gate, 7432 IC for OR gate, 7400 IC for NAND gate, 7402 IC for NOR gate, 7486 IC for XOR gate.

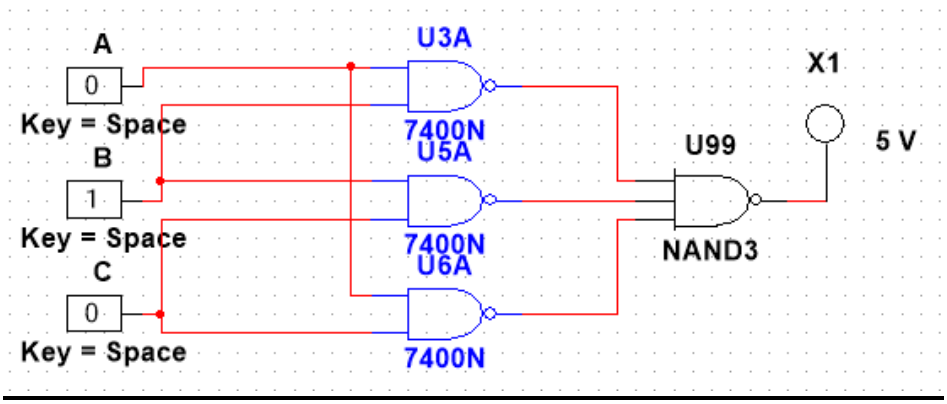
3) Construct a circuit of output F, where F=AB + BC + CA, by using NAND gates only in the Multisim Software and show the output states for each available condition.
Ans:

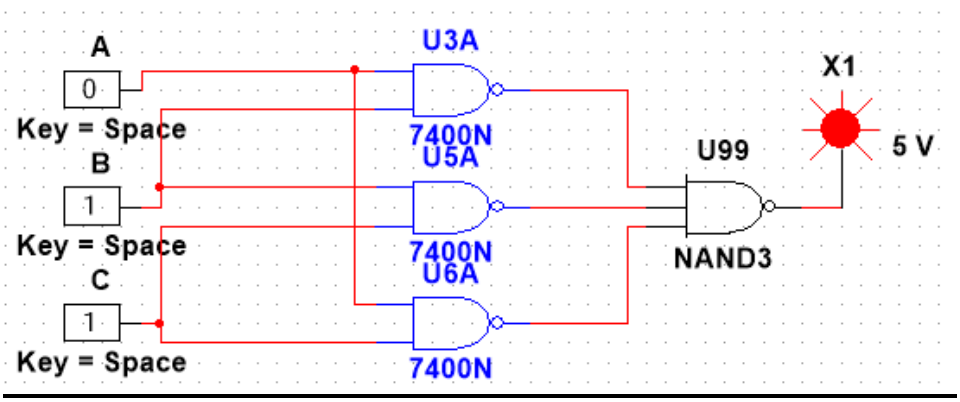
A	B	C	AB	BC	CA	AB + BC + CA
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	1
1	0	0	0	0	0	0
1	0	1	0	0	1	1
1	1	0	1	0	0	1
1	1	1	1	1	1	1

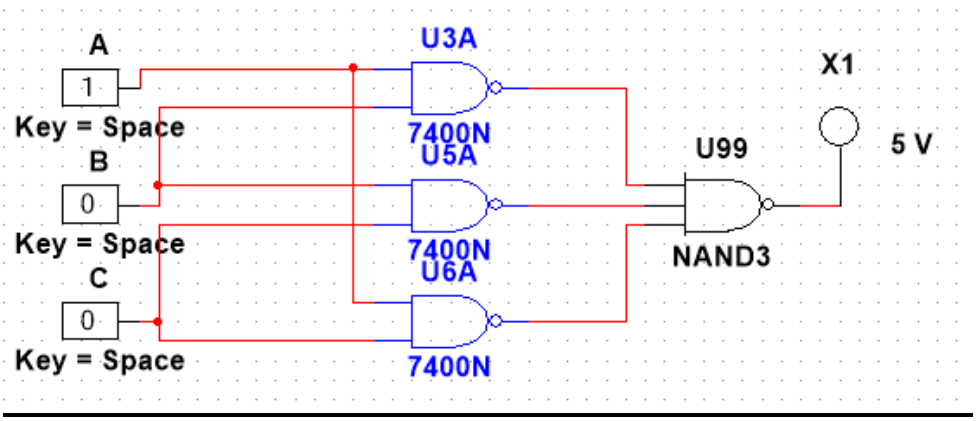
Simulation of AB + BC + CA:

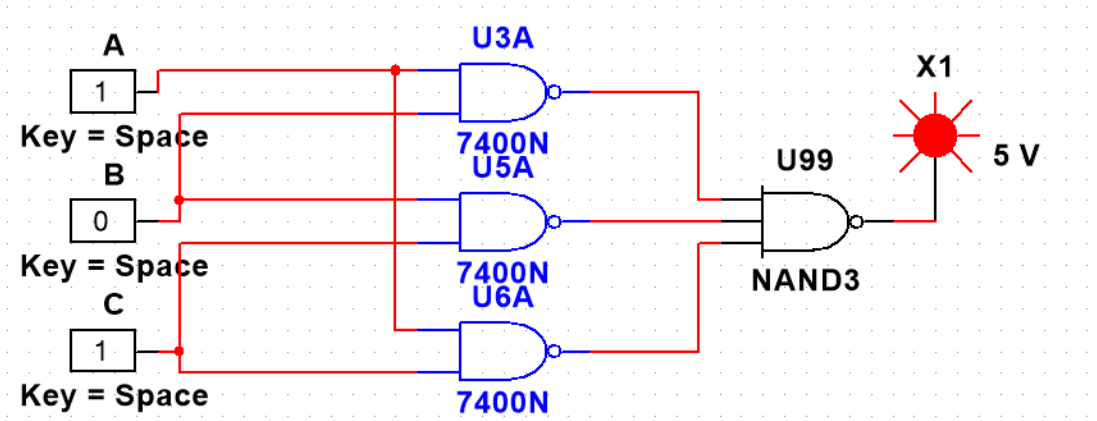


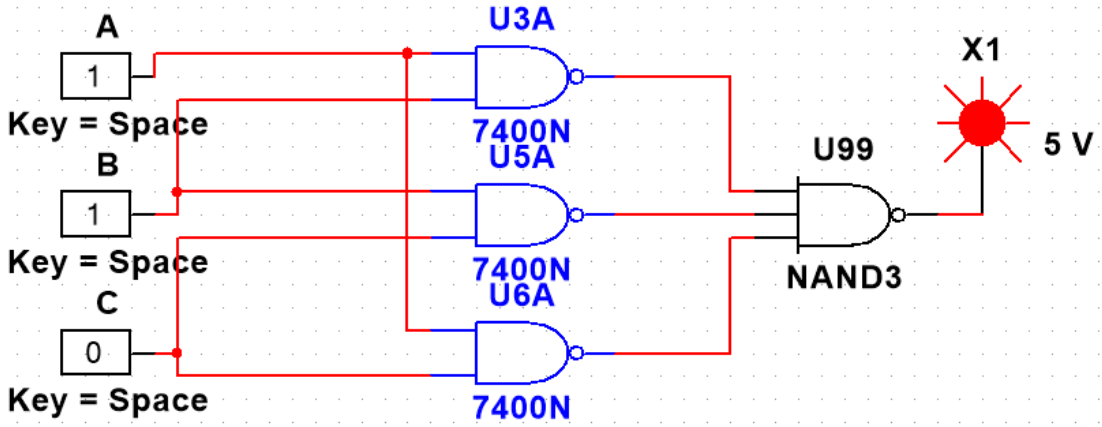

A
0
Key = Space
B
0
Key = Space
C
1
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V

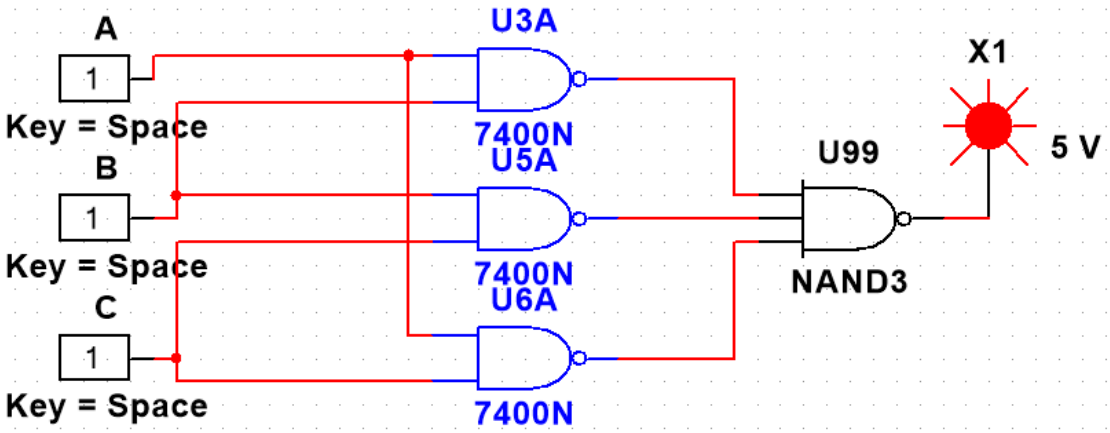

A
0
Key = Space
B
1
Key = Space
C
0
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V


A
0
Key = Space
B
1
Key = Space
C
1
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V


A
1
Key = Space
B
0
Key = Space
C
0
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V


A
1
Key = Space
B
0
Key = Space
C
1
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V


A
1
Key = Space
B
1
Key = Space
C
0
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V


A
1
Key = Space
B
1
Key = Space
C
1
Key = Space
U3A
7400N
U5A
7400N
U6A
7400N
U99
NAND3
X1
5 V

Discussion & Conclusion:

In this experiment, basic knowledge was gained about learning and implementing Integrated circuits and different logic circuits using logic gates. According to the truth table, the NAND gate behaved perfectly during the simulation. Finally, the results of the simulations and our Truth-table were compared and found to be identical. NAND and NOR gates are universal gates. Any other gates can be made by using only these two gates. In this experiment, we have implemented the AND with NOR, OR with NAND, XOR with NAND, XOR with NOR gates.

Reference:

1) www.tutorialspoint.com
2) www.electronics-tutorials.ws
3) faculty.kfupm.edu.sa
4) “Digital Fundamentals” by Thomas L. Floyd