

AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH

Faculty of Engineering

Laboratory Report Cover Sheet



Students must complete all details except the faculty use part.

Please submit all reports to your subject supervisor or the office of the concerned faculty.

Lab Title: Studying different digital logic gates and designing of basic logic gates using Universal gates

Experiment Number: 01 Due Date: 04/10/2023 Semester: Fall 2022-2023

Subject Code: EEE _____ Subject Name: DIGITAL LOGIC AND CIRCUITS LAB Section: L

Course Instructor: **NUZAT NUARY ALAM** Degree Program: **B.Sc. CSE**

Declaration and Statement of Authorship:

1. I/we hold a copy of this report, which can be produced if the original is lost/ damaged.
2. This report is my/our original work and no part of it has been copied from any other student's work or from any other source except where due acknowledgement is made.
3. No part of this report has been written for me/us by any other person except where such collaboration has been authorized by the lecturer/teacher concerned and is clearly acknowledged in the report.
4. I/we have not previously submitted or currently submitting this work for any other course/unit.
5. This work may be reproduced, communicated, compared and archived for the purpose of detecting plagiarism.
6. I/we give permission for a copy of my/our marked work to be retained by the school for review and comparison, including review by external examiners.

I/we understand that

7. Plagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to expulsion from the University. Plagiarized material can be drawn from, and presented in, written, graphic and visual form, including electronic data, and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited.
8. Enabling plagiarism is the act of assisting or allowing another person to plagiarize or to copy your work.

Group Number (if applicable):	Individual Submission	Group Submission
-------------------------------	-----------------------	------------------

No.	Student Name	Student Number	Student Signature	Date
Submitted by:				
1	A.H.M Tanvir	22-47034-1		04/10/2023
Group Members:				
2				
3				
4				
5				

For faculty use only:

Total Marks: _____ Marks Obtained: _____

Faculty comments _____

Title: Studying different digital logic gates and designing of basic logic gates using Universal gates

Abstract:

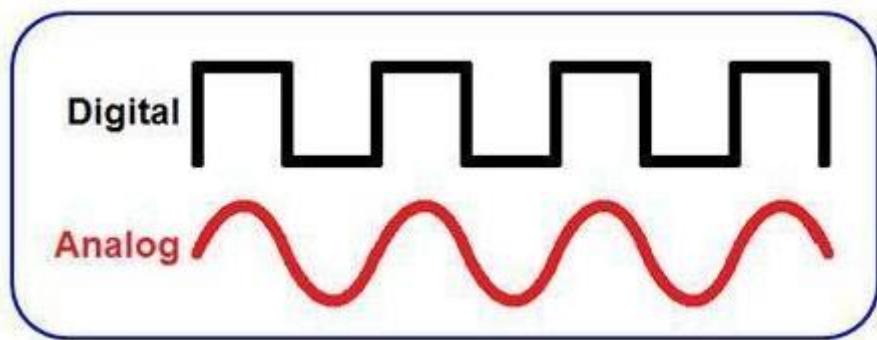
To learn the characteristics of several logic gates and to get familiar with the digital trainer board and digital ICs.

Part I (Basic Logic IC's):

An integrated circuit (also referred to as an IC, a chip, or a microchip) is a set of electronic circuits on one small plate ("chip") of semiconductor material, normally silicon. This can be made much smaller than a discrete circuit made from independent components. Different integrated circuits are used to implement different logical operations in the trainer board which will be introduced in this experiment.

Theory and Methodology:

In analog signals, information is translated into electric pulses of varying amplitude but in case of digital, translation of information is in binary format (zero or one) where each bit is representative of two distinct amplitudes.



The main advantage of digital signals over analog signals is that the precise signal level of the digital signal is not vital. This means that digital signals are immune to the imperfections of real electronic systems which tend to spoil analog signals. Codes are often used in the transmission of information. These codes can be used either as a means of keeping the information secret or as a means of breaking the information into pieces that are manageable by the technology used to transmit the code. It can convey information with greater noise immunity, because each information component (byte etc.) is determined by the presence or absence of a data bit (0 or one). Analog signals vary continuously, and their value is affected by all levels of noise. Digital signals can be processed by digital circuit components, which are cheap and easily produced in many components on a single chip. It uses typically less bandwidth with less electromagnetic interference. Moreover, Information storage can be easier in digital systems than in analog ones. The noise-immunity of digital systems permits data to be stored and retrieved without degradation.

There are two sorts of circuits which are known as integrated circuit and discrete circuit. The two main advantages of ICs over discrete circuits are cost and performance. Cost is low because the chips, with all their components, are printed as a unit by photolithography rather than being constructed one transistor at a time. Furthermore, much less material is used to construct a packaged IC die than to construct a discrete circuit. Performance is high because the components switch quickly and consume little power (compared to their discrete counterparts) as a result of the small size and proximity of the components.

A logic gate is an elementary building block of a digital circuit. Most logic gates have two inputs and one output. At any given moment, every terminal is in one of the two binary conditions low (0V) and high (5V), represented by different voltage levels. The logical state of a terminal can, and generally does, change often, as the circuit processes data. In most logic gates, the low state is approximately zero volts (0 V), while the high state is approximately five volts positive (+5 V).

There are seven basic logic gates: AND, OR, NOT, NOR, NAND, XOR and XNOR. Different logic operations of different IC's will be introduced which perform the following characteristics:

Operation	Expression
AND	$Y=AB$
OR	$Y=A+B$
NOT	$Y=\bar{A}$
NOR	$Y=\bar{A} + \bar{B} = \bar{A} \bar{B}$
NAND	$Y=\bar{A}\bar{B} = \bar{A} \bar{B}$
XOR	$Y=A\oplus B = \bar{A} B + A \bar{B}$
XNOR	$Y= AB+\bar{A} \bar{B} = \bar{A} \bar{B}$

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

Pin configuration for IC-74HC08N:

For a quadrature 2input AND gate HC08 device code is used. 74HC series devices are designed to work with a 5 V power supply, voltages from 2 V to 5 V are allowed and most circuits work well using 5 V.

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

Pin configuration for IC-74HC32N:

HC32 is the device code. 74HC32 is a Quad 2-input OR gate (High Speed CMOS version) which has lower current consumption/wider Voltage range from 2 to 5V. It requires a low input current of 1 μ A with high noise immunity characteristics of CMOS devices.

NOT operation:

The NOT operation changes one logic level to the opposite logic level. It is implemented by a logic circuit known as an inverter.



Fig1.3: Symbol of NOT gate

Truth Table:

Input, A	Output, F
0	1
1	0

Pin configuration for IC-74HC04N :

The 74HC04 is a hex inverter which consists of six inverters which perform logical invert action. The inputs include clamp diodes that enable the use of current limiting resistors to interface inputs to voltages in excess of Vcc. The Input level for 74HC04 is CMOS level.

NAND operation:

The NAND gate operates as an AND gate followed by a NOT gate. It acts in the manner of the logical operation "AND" followed by negation. The output will be low if both inputs are high. Otherwise, the output is high.



Fig 1.4: Symbol of NAND gate

Truth Table:

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

Pin configuration for IC-74HC00N:

HC00 is the device code. The device inputs are compatible with Standard CMOS outputs, with pullup resistors. The operating voltage range is 2.0 to 5.0 V and low input current is 1.0 μ A.

NOR operation:

The NOR gate is a combination OR gate followed by an inverter. Its output is high if both inputs are low. Otherwise, the output is low.



Fig 1.5: Symbol of NOR gate

Truth Table:

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

Pin configuration for IC-74HC02N:

The 74HC02 is a high-speed Si-gate CMOS device that provides a quadrature 2 –input NOR function. CMOS level is the input level for this sort of IC's. The operating Voltage Range is 2.0 to 5.0 V and low input current is 1.0 μ A.

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.

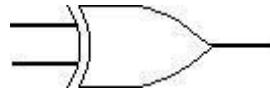


Fig 1.6: Symbol of XOR gate

Truth Table:

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

Pin configuration for IC-74HC86N :

HC86 is the device code for a quad 2-input xor gate which utilizes advanced silicon gate CMOS technology. It maintains low power consumption and high noise immunity characteristic of standard CMOS integrated circuits. The 74HC logic family has a voltage range of 2V to 5V and the operating temperature is -40°C to 125°C with input current of 1 μ A.

XNOR operation:

The XNOR (*exclusive-NOR*) gate is a combination XOR gate followed by an inverter. Its output is high if the inputs are the same, and low if the inputs are different.

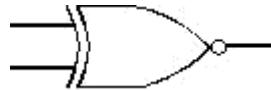


Fig 1.7: Symbol of XNOR gate

Truth Table:

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

Using combinations of logic gates, complex operations can be performed. Arrays of logic gates are found in digital integrated circuits (ICs). As IC technology advances, the required physical volume for each individual logic gate decreases and digital devices of the same or smaller size become capable of performing ever more complicated operations at ever-increasing speeds.

Apparatus:

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

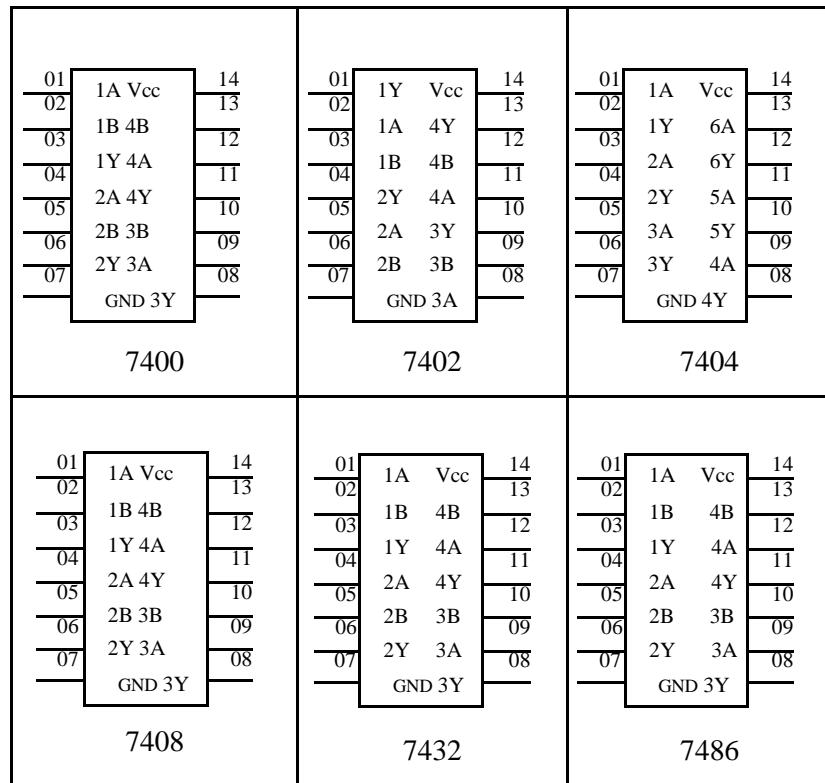
Integrated Circuits (ICs):

7400 : 1 pcs
 7402 : 1 pcs
 7404 : 1 pcs
 7408: 1 pcs
 7432 : 1 pcs
 7486 : 1 pcs

Precautions:

The IC contains protection circuitry to guard against damage due to high static voltages or electric fields. However, precautions must be taken to avoid applications of any voltage higher than maximum rated voltages. For proper operation, V_{in} and V_{out} should be constrained to the range GND (V_{in} or V_{out}) to V_{CC} .

IC configurations:



Part II: Study of Universal Gates

A Logic Gate which can infer any of the gates among Logic Gates or a gate which can be used to create any Logic gate is called Universal Gate. **NAND** and **NOR** Gates are called Universal Gates because all the other gates such as NOT, AND, OR, XOR, XNOR etc. can be created by using these gates.

The Objective of this lab is to implement different logic functions using universal gates.

Theory and Methodology:

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 2.1: Symbol of NAND gate

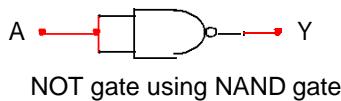
$$\text{Output, } Q = \quad = \quad +$$

Truth Table			
Input A	Input B	Output Q	
0	0	1	
0	1	1	
1	0	1	
1	1	0	

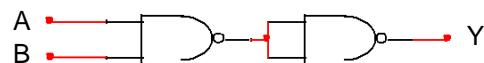
It is possible to construct other gates using NAND gates which are shown in the Experimental procedure part.

Implementing various logic functions using NAND Gates:

1) Implementing NOT gate using NAND gate:

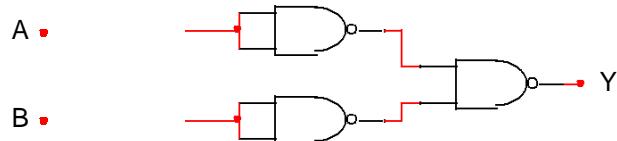


2) Implementing AND gate using NAND gate:



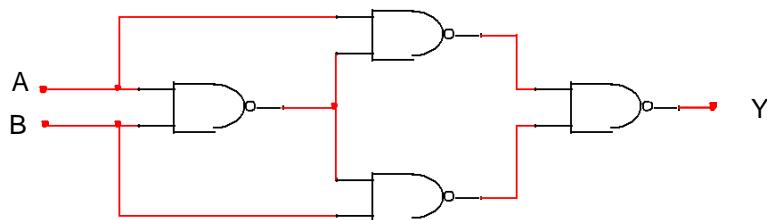
AND gate using NAND gates

3) Implementing OR gate using NAND gate:



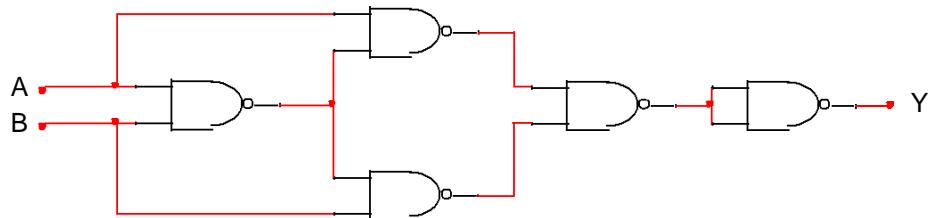
OR gate using NAND gates

4) Implementing XOR gate using NAND gate:



XOR gate using NAND gates

5) Implementing XNOR gate using NAND gate:



XNOR gate using NAND gates

NOR gate:

The **NOR** gate represents the complement of the OR operation. Its name is an abbreviation of **NOT OR**. The graphic symbol for the NOR gate consists of an OR symbol with a bubble on the output, denoting that a complement operation is performed on the output of the OR gate. The truth table and the graphic symbol of NOR gate is shown in the figure.



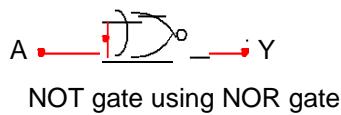
Fig 2.2: Symbol of NOR gate

$$\text{Output, } Q = \quad =$$

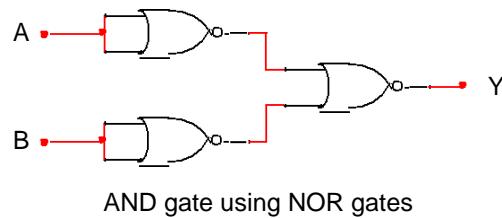
Truth Table		
Input A	Input B	Output Q
0	0	1
0	1	0
1	0	0
1	1	0

Implementing various logic functions using NOR Gates:

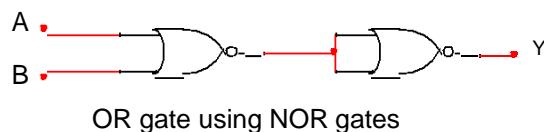
- 1) Implementing NOT gate using NOR gate:



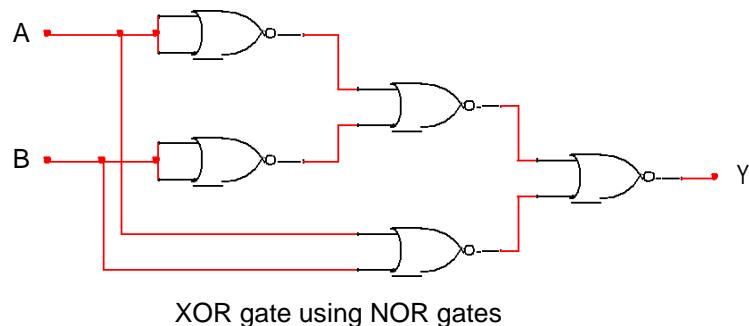
- 2) Implementing AND gate using NOR gate:



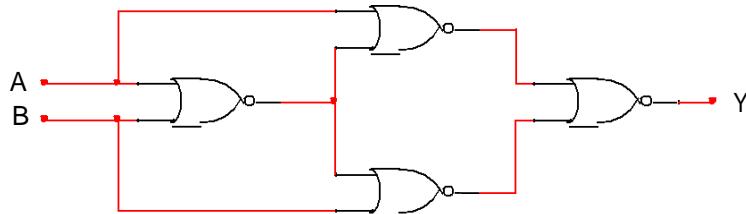
- 3) Implementing OR gate using NOR gate:



- 4) Implementing XOR gate using NOR gate:



5) Implementing XNOR gate using NOR gate:



XNOR gate using NOR gates

Pre-Lab Homework:

The Boolean algebra rules and universal gates must be studied by students. Simulation of the circuits shown in the circuit diagram section using Power Sim 9.1.1 (PSIM) must be performed, and the simulation results must be presented to the instructor before the start of the experiment.

Apparatus:

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

Precautions:

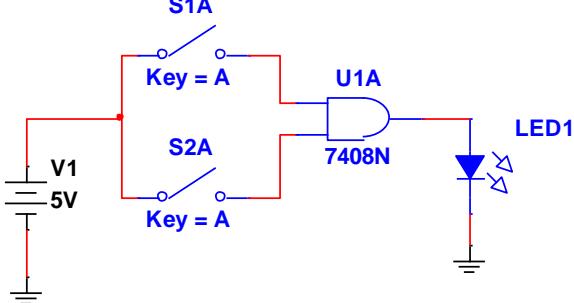
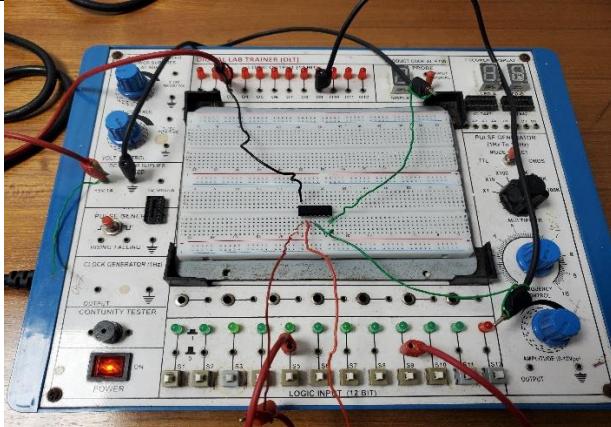
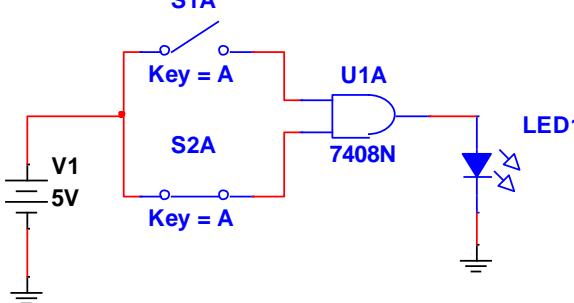
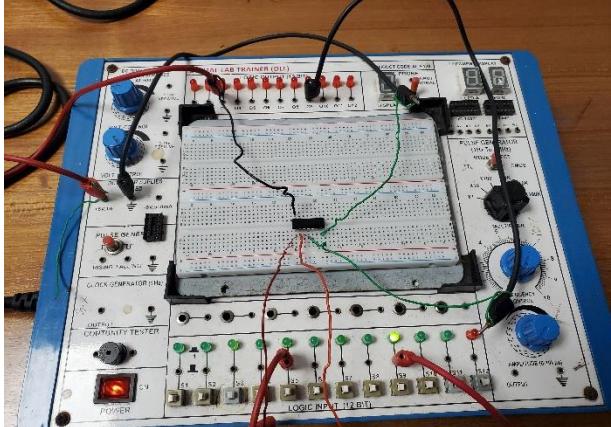
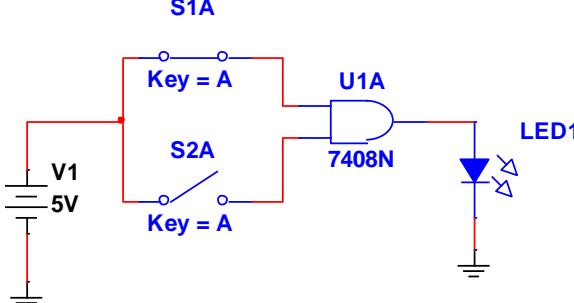
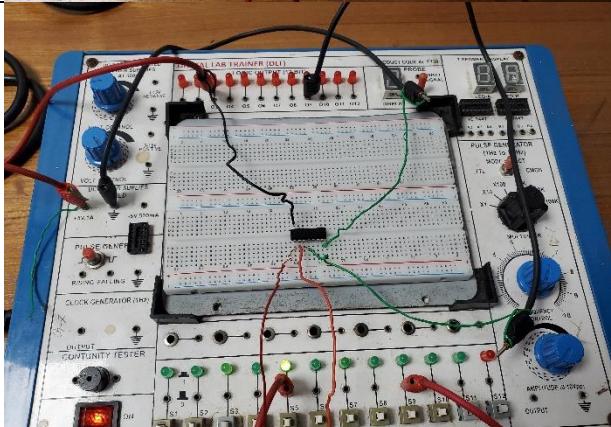
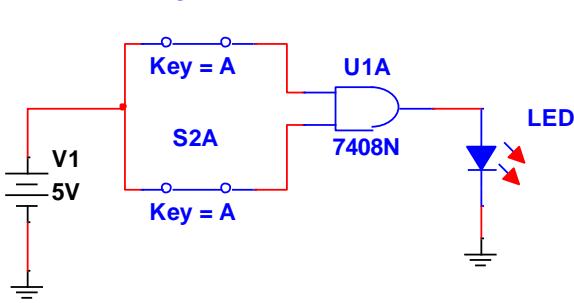
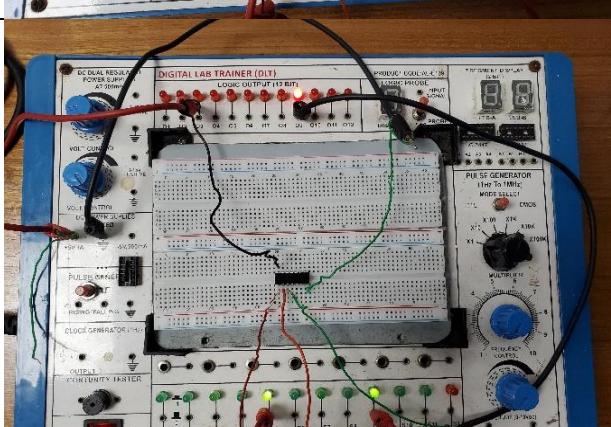
After setting up the circuit, all connections should be checked by the instructor, and it should be ensured that only enough voltage is applied to turn on the chip; otherwise, it may get damaged.

Experimental Procedure:

1. An X-OR and X-NOR gate were constructed on the trainer board using only NAND gates with the required IC.
2. Equivalent NOT, OR, and AND gates were derived using NOR gates only. Subsequently, an X-OR and X-NOR gate were constructed on the trainer board using only NOR gates with the necessary IC.
3. The given expressions were converted using universal gates and implemented on the trainer board. The results were then compared with the truth table of the equations.
 - i) $A (+) B$
 - ii) $(A(+B)) + C$
 - iii) $(AB + CD)'$

Simulation and Measurement:

AND:

Simulation	Hardware
 <p style="text-align: center;">$A=0, B=0; Y=0$</p>	
 <p style="text-align: center;">$A=0, B=1; Y=0$</p>	
 <p style="text-align: center;">$A=1, B=0; Y=0$</p>	
 <p style="text-align: center;">$A=1, B=1; Y=1$</p>	

OR:

Simulation	Hardware
<p style="text-align: center;">$A=0, B=0; Y=0$</p>	
<p style="text-align: center;">$A=0, B=1; Y=1$</p>	
<p style="text-align: center;">$A=1, B=0; Y=1$</p>	
<p style="text-align: center;">$A=1, B=1; Y=1$</p>	

NOT:

Simulation	Hardware
<p style="margin-top: 10px;">$A=0; Y=1$</p>	
<p style="margin-top: 10px;">$A=1; Y=0$</p>	

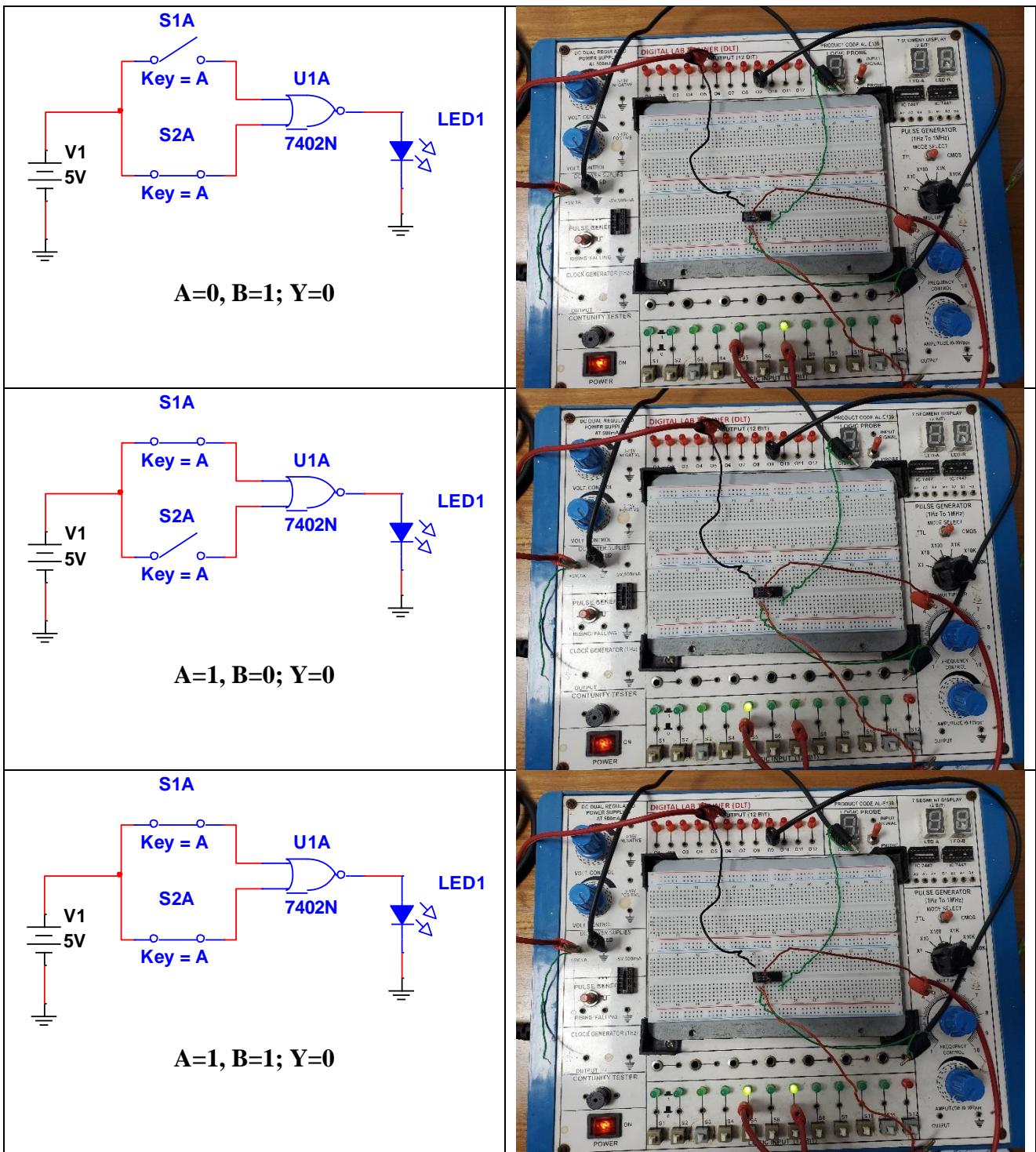
NAND:

Simulation	Hardware
<p style="margin-top: 10px;">$A=0, B=0; Y=1$</p>	

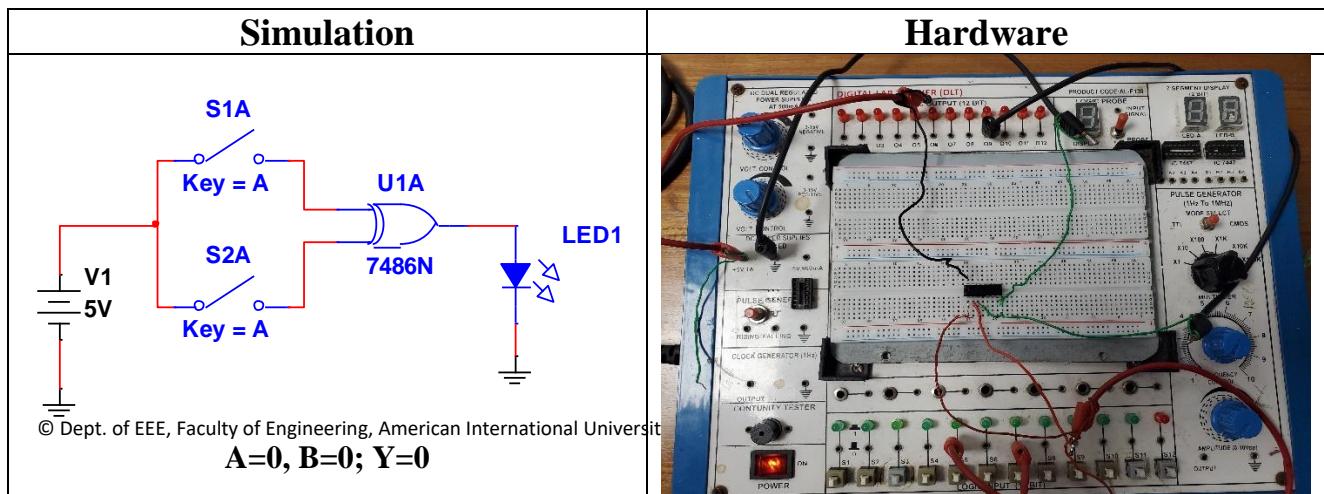
<p>A=0, B=1; Y=1</p>	
<p>A=1, B=0; Y=1</p>	
<p>A=1, B=1; Y=0</p>	

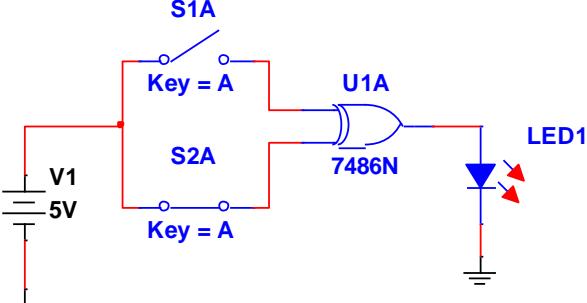
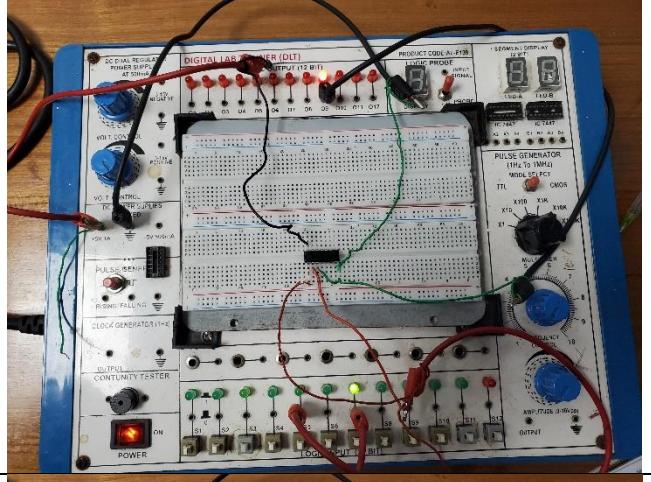
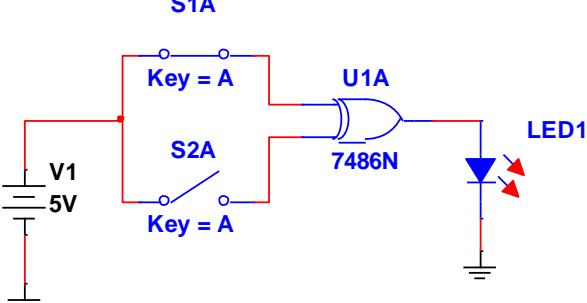
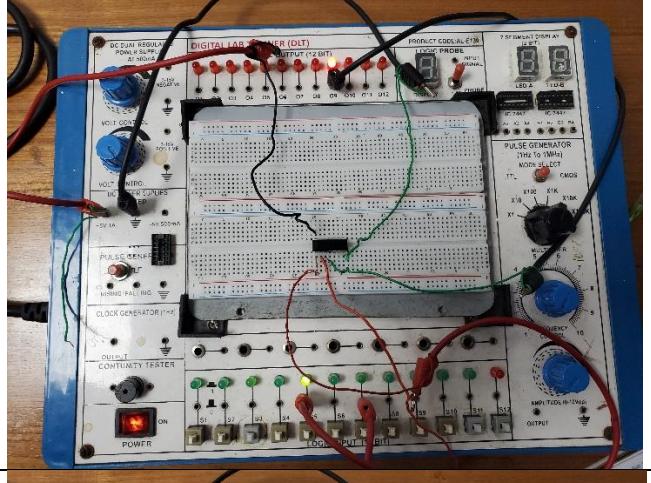
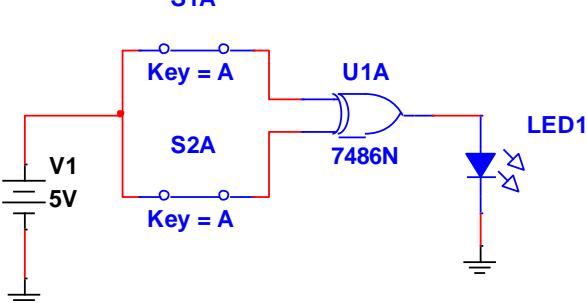
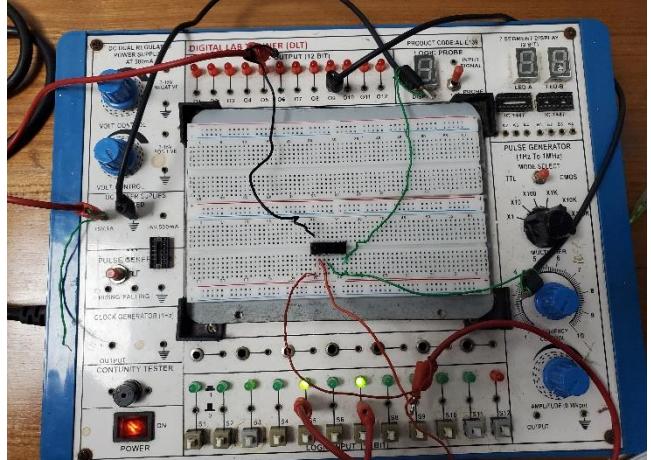
NOR:

Simulation	Hardware
<p>A=0, B=0; Y=1</p>	

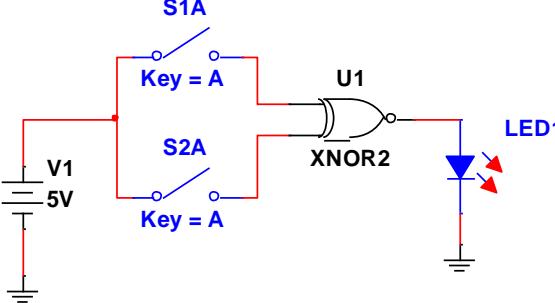
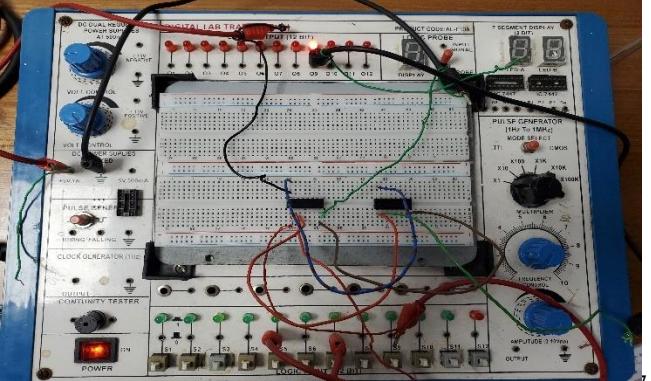


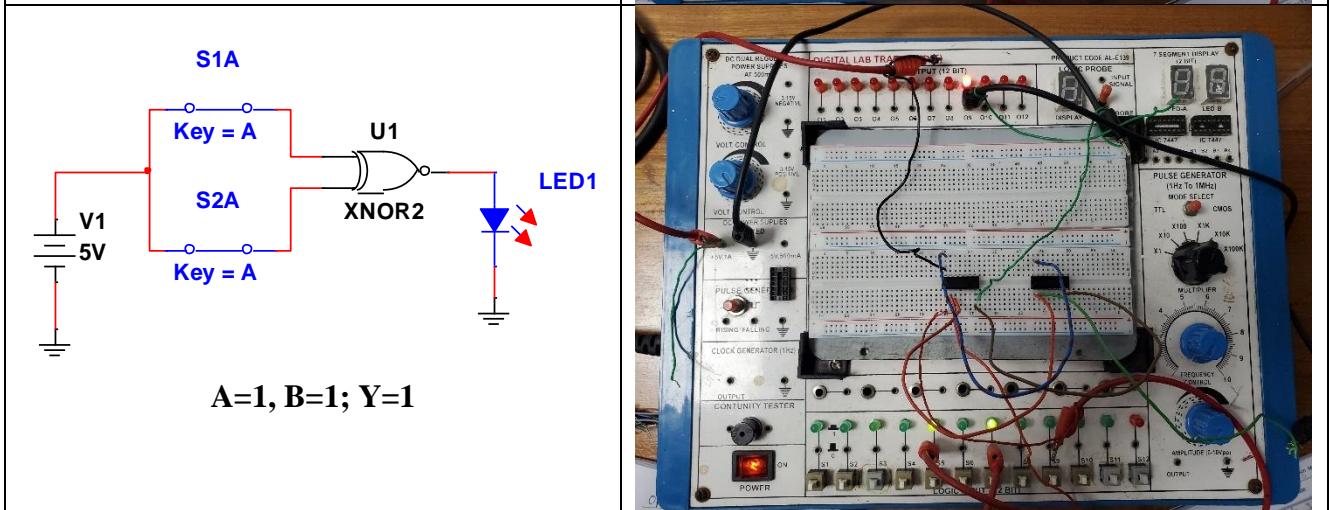
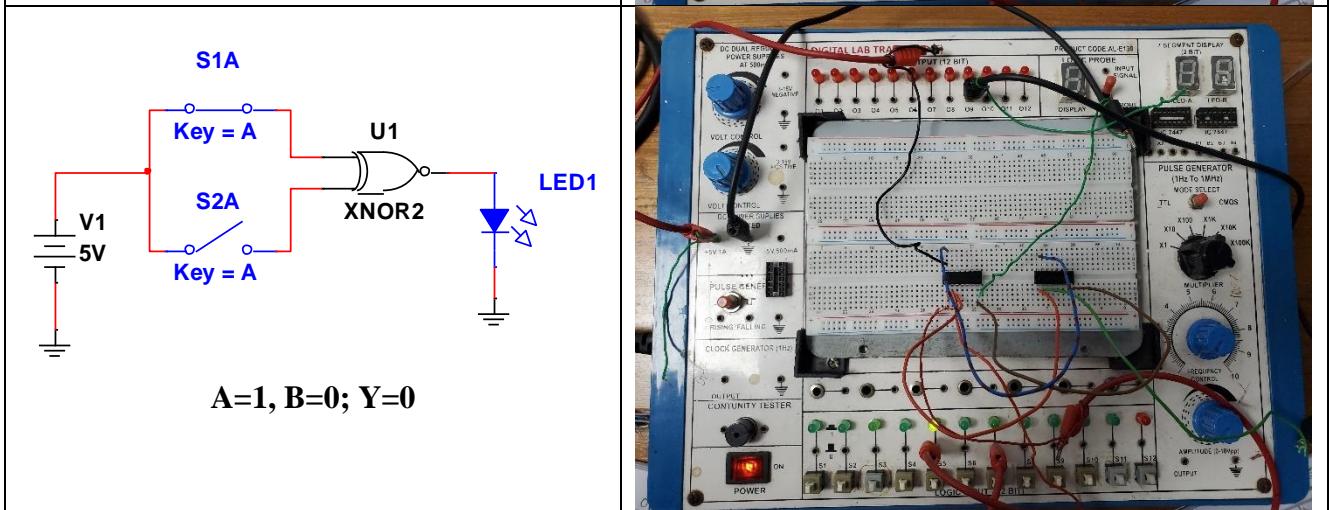
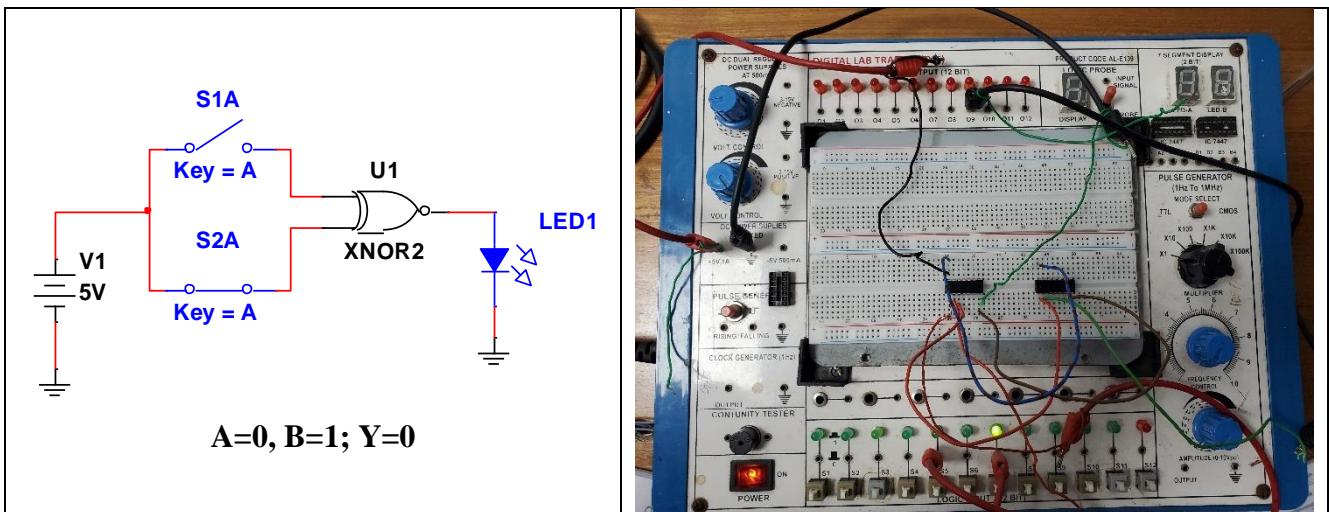
XOR:



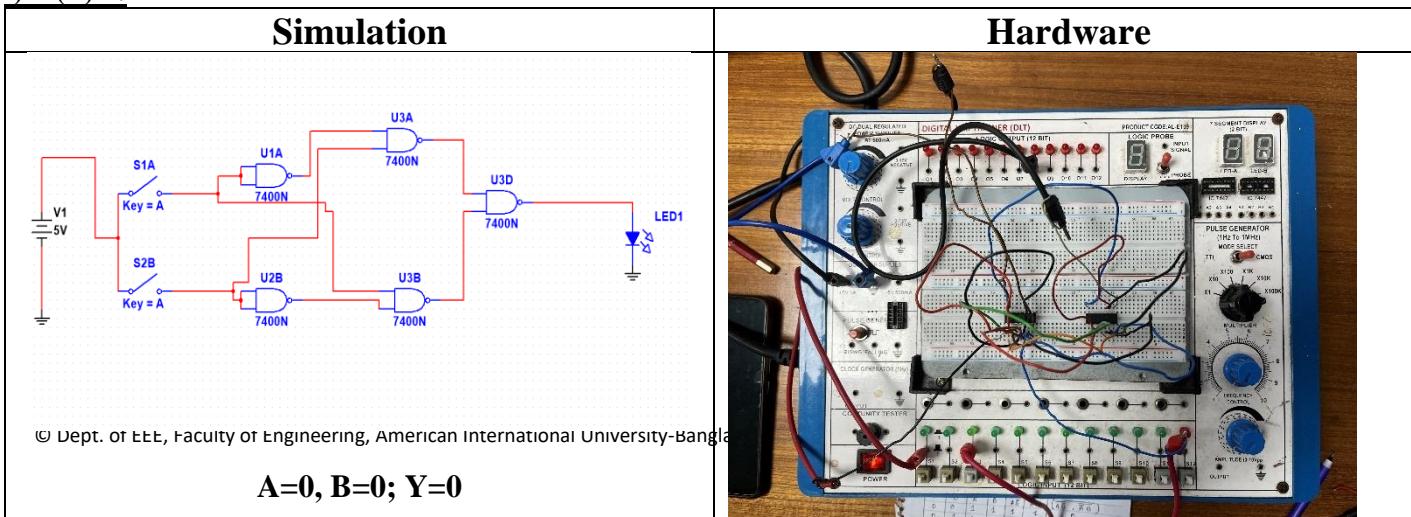
 <p>A=0, B=1; Y=1</p>	
 <p>A=1, B=0; Y=1</p>	
 <p>A=1, B=1; Y=0</p>	

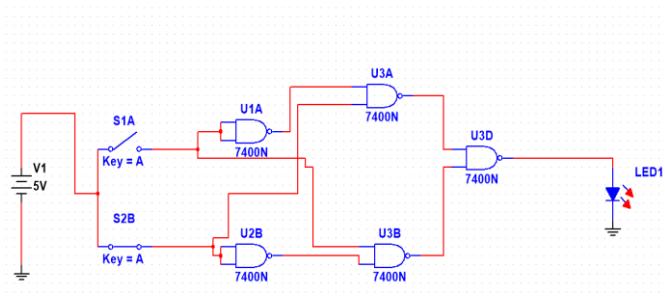
XNOR:

Simulation	Hardware
 <p>A=0, B=0; Y=1</p>	

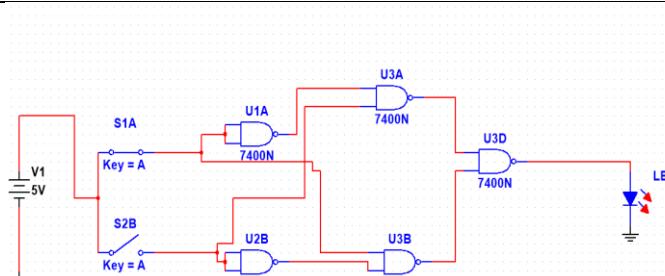
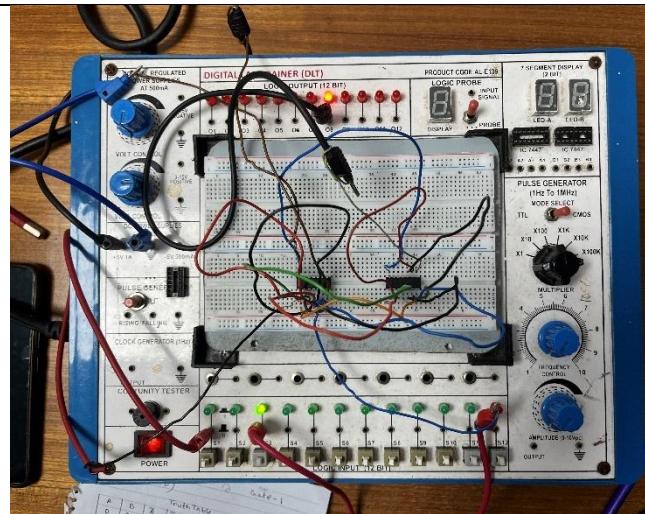


i) A(+)-B:

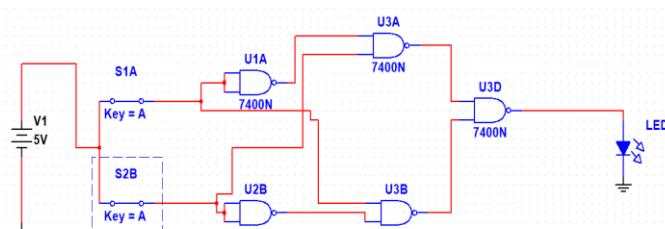
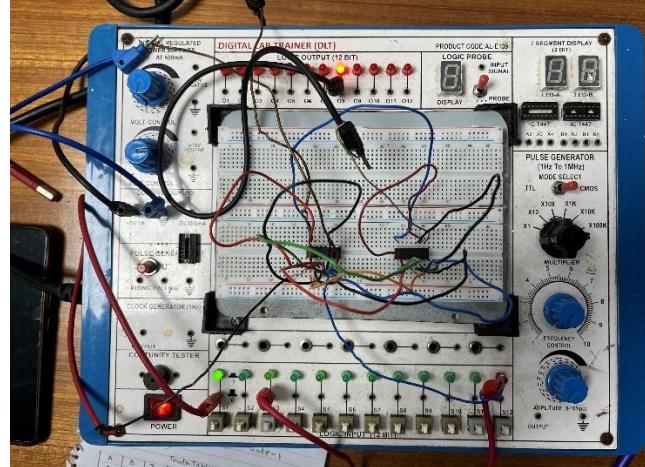




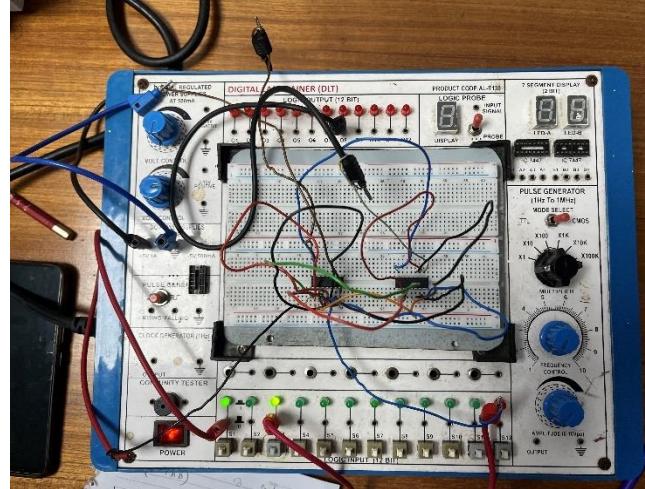
A=0, B=1; Y=1



A=1, B=0; Y=1

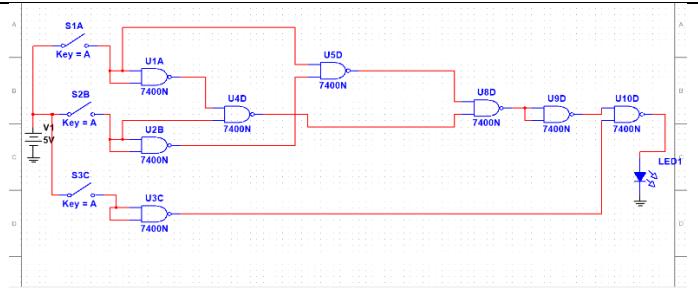


A=1, B=1; Y=0



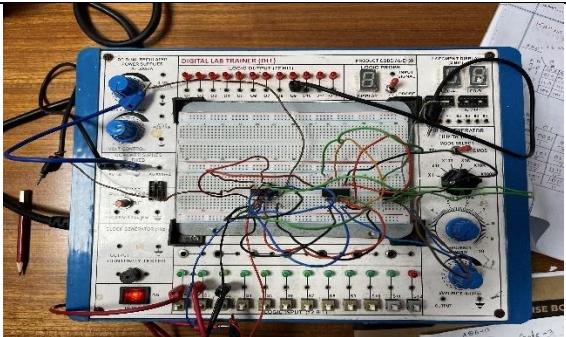
ii) $(A+B)+C$:

Simulation

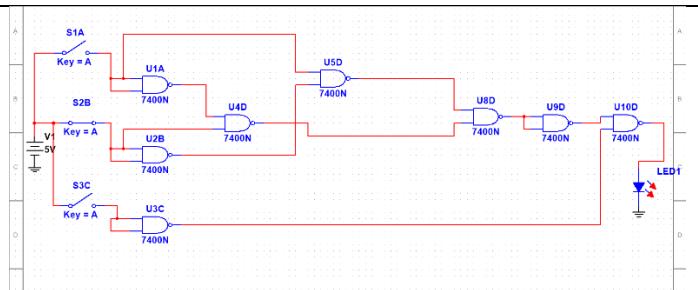


$A=0, B=0, C=0; Y=0$

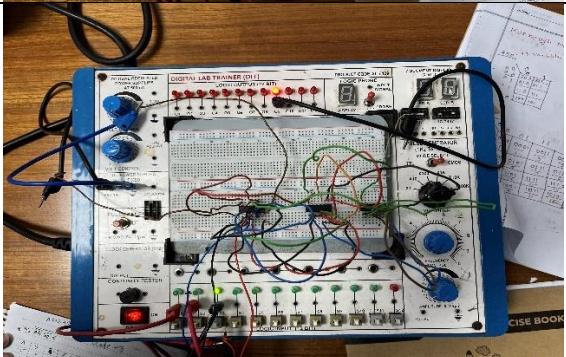
Hardware



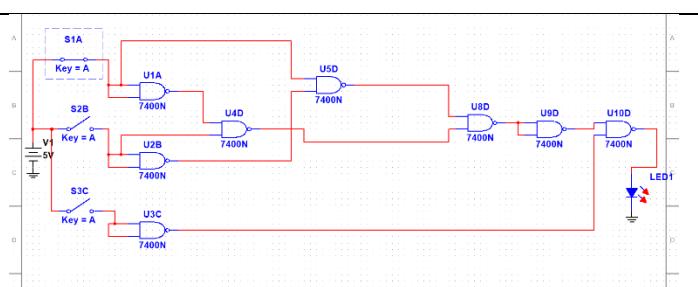
$A=0, B=0, C1; Y=1$



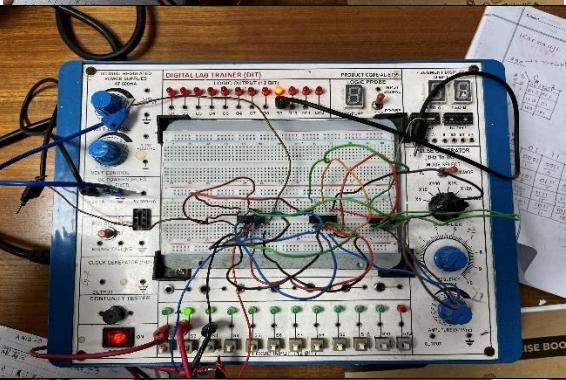
$A=0, B=1, C=0; Y=1$

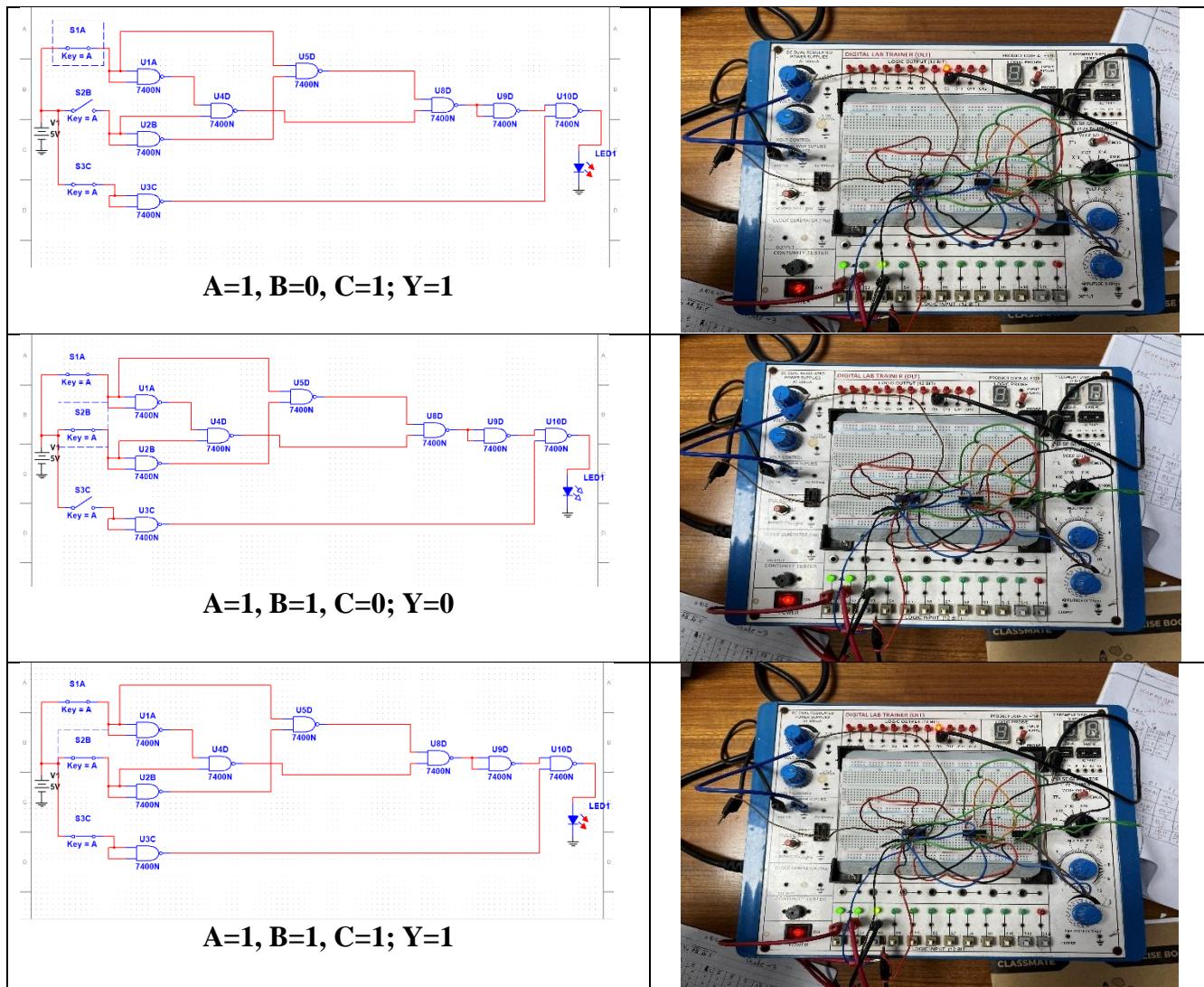


$A=0, B=1, C=1; Y=1$

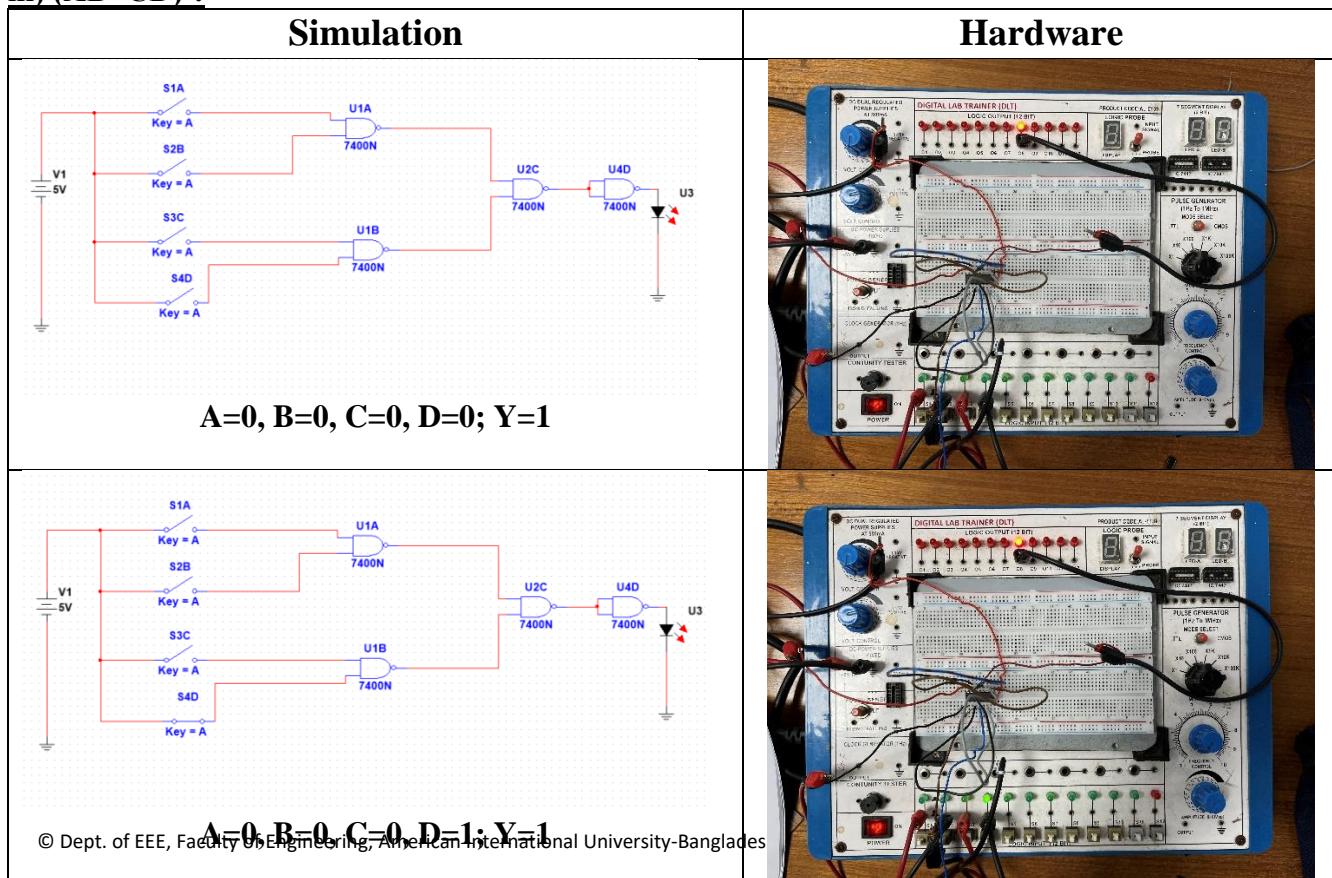


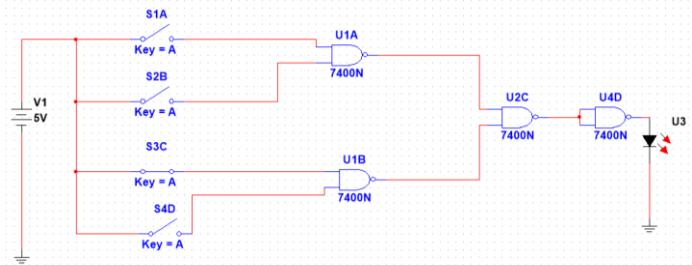
$A=1, B=0, C=0; Y=1$



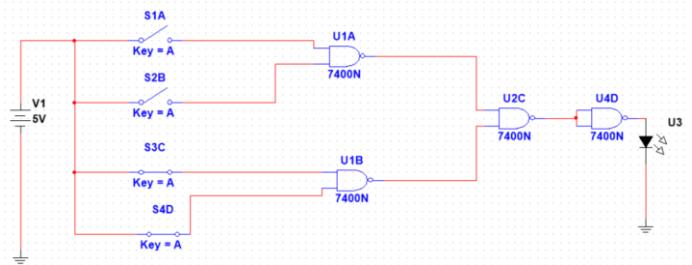
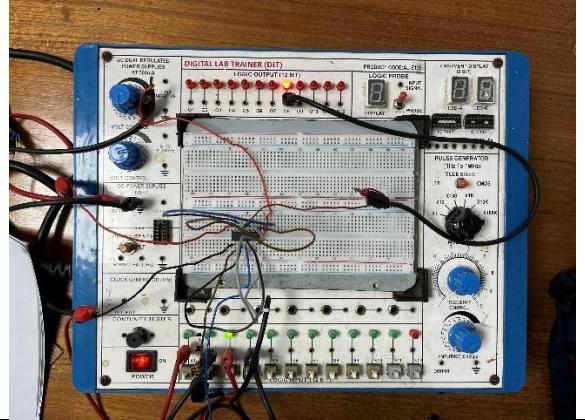


iii) $(AB+CD)'$:

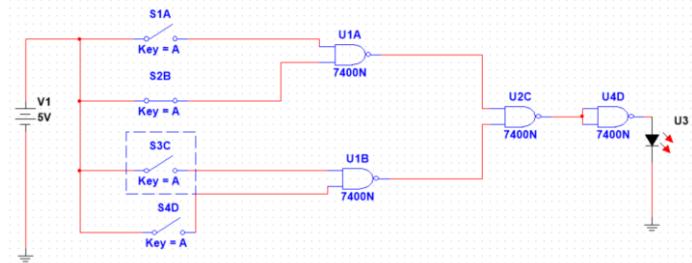
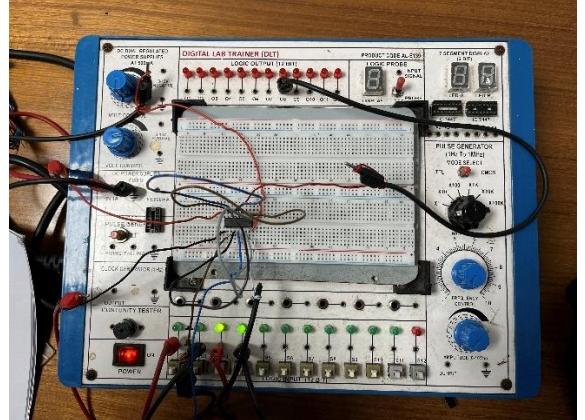




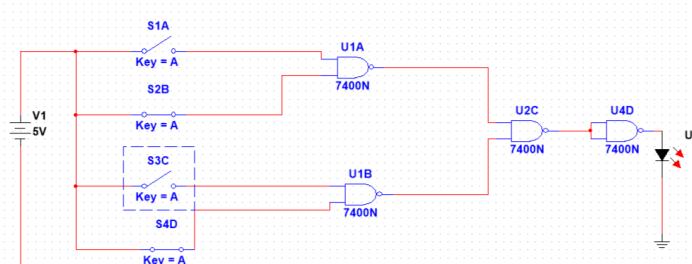
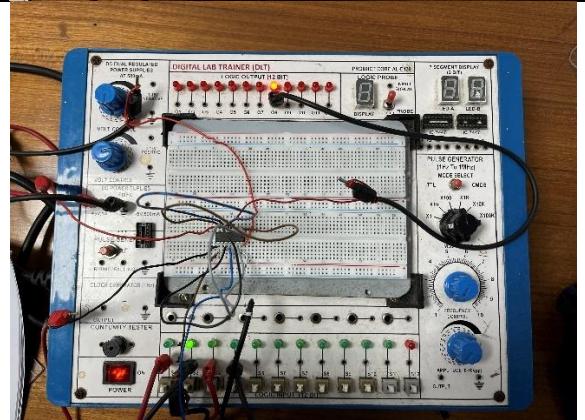
A=0, B=0, C=1, D=0; Y=1



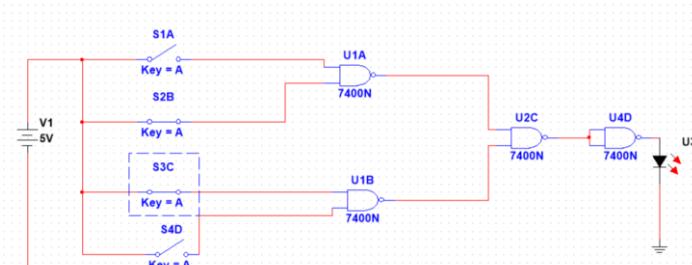
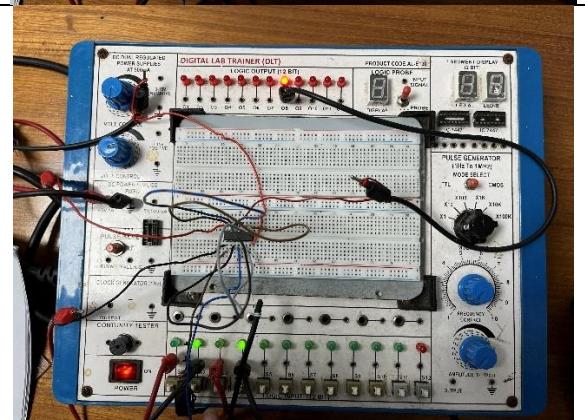
A=0, B=0, C=1, D=1; Y=1



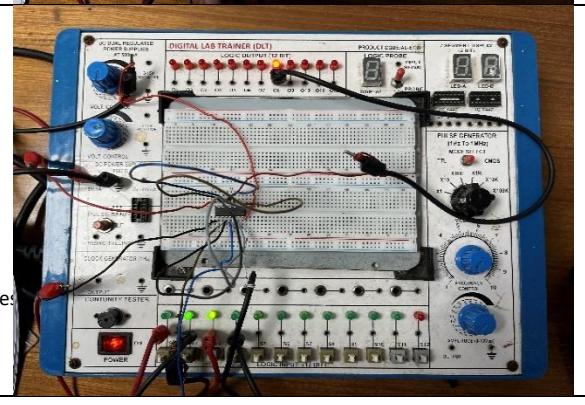
A=0, B=1, C=0, D=0; Y=1

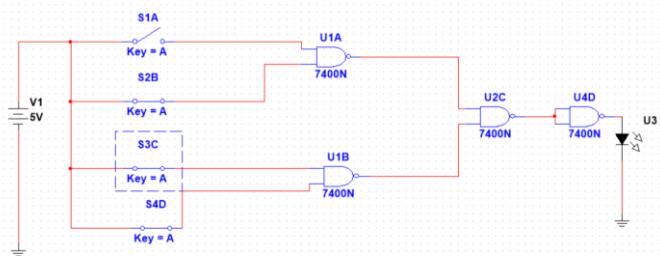


A=0, B=1, C=0, D=1; Y=1

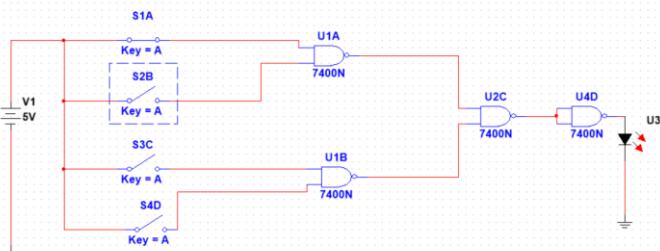
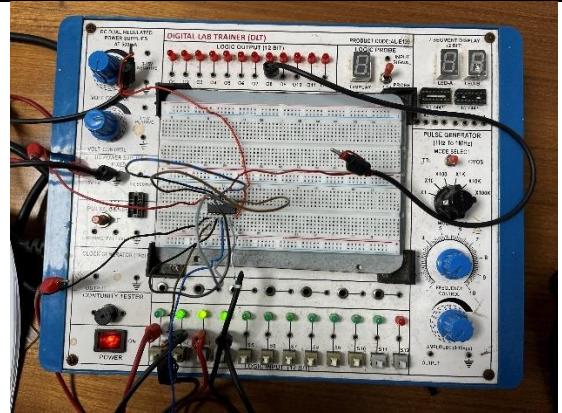


A=0, B=1, C=1, D=0; Y=1

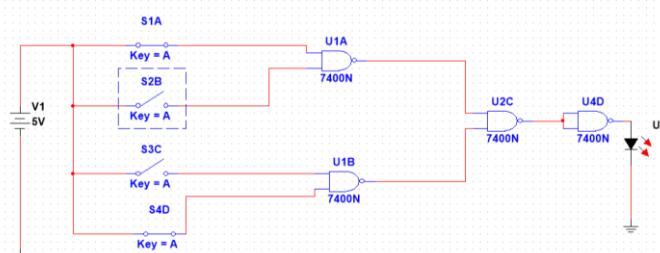
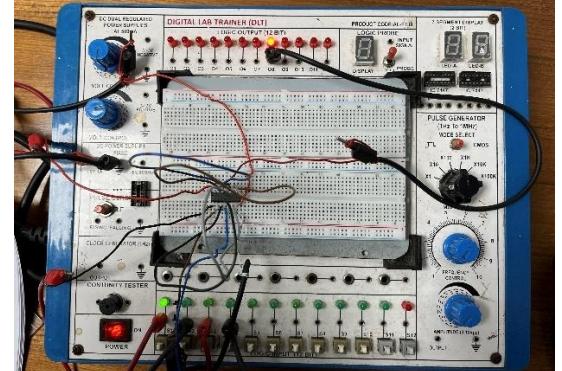




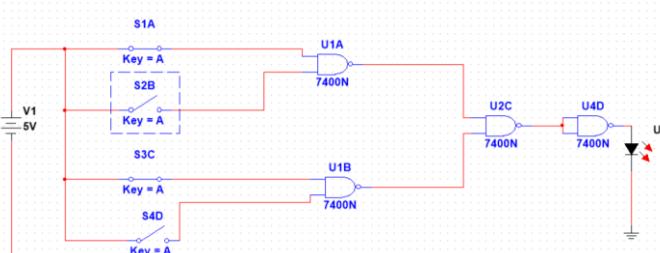
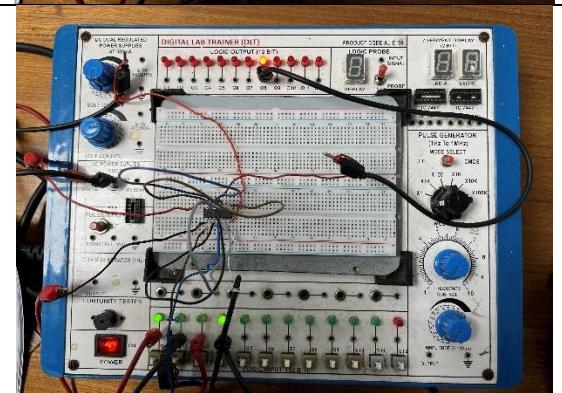
A=0, B=1, C=1, D=1; Y=0



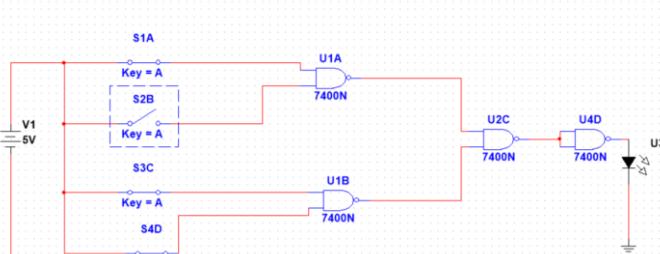
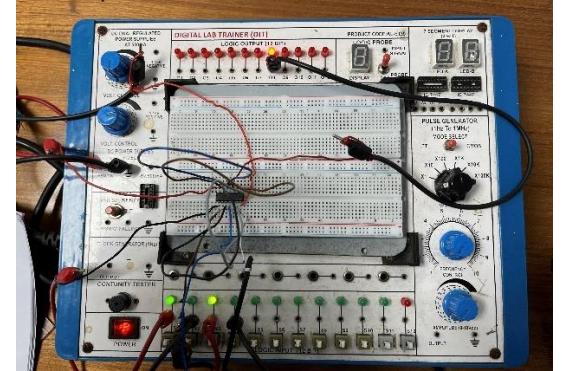
A=1, B=0, C=0, D=0; Y=1

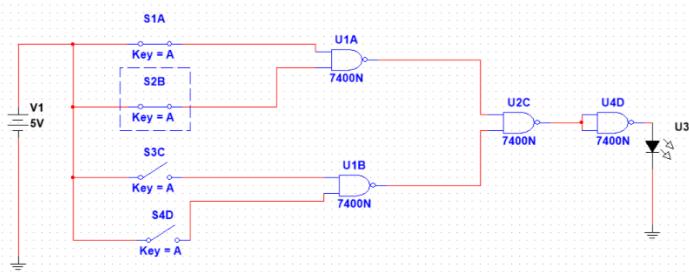


A=1, B=0, C=0, D=1; Y=1

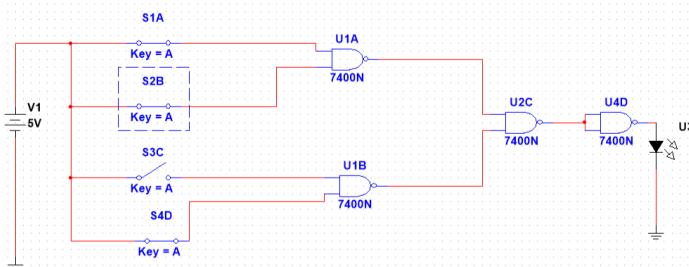
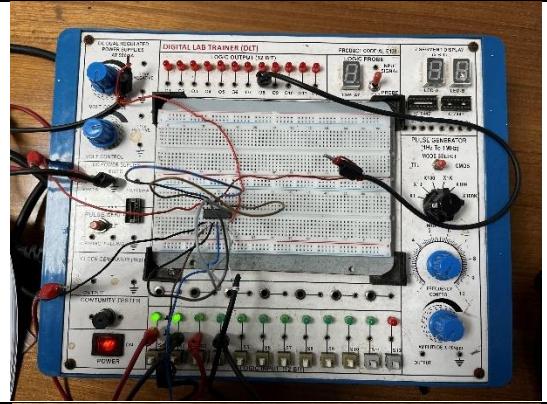


A=1, B=0, C=1, D=0; Y=1

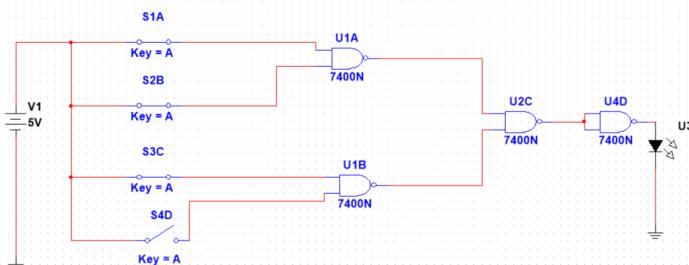
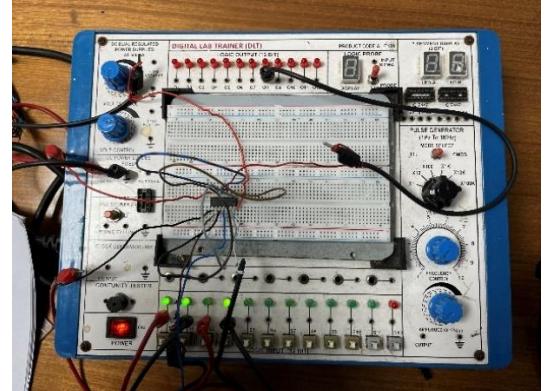




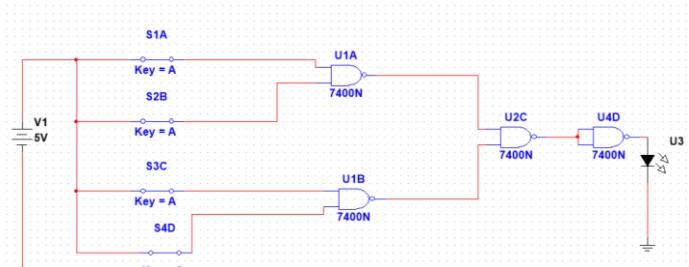
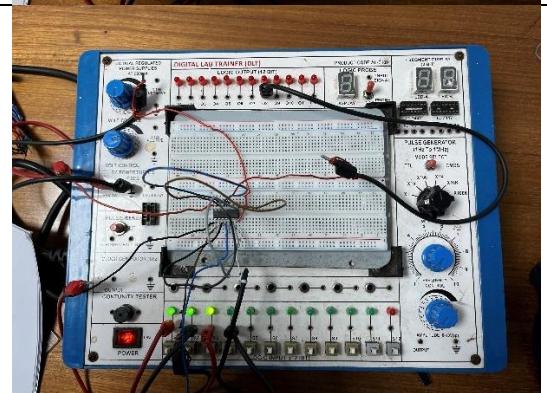
A=1, B=1, C=0, D=0; Y=0



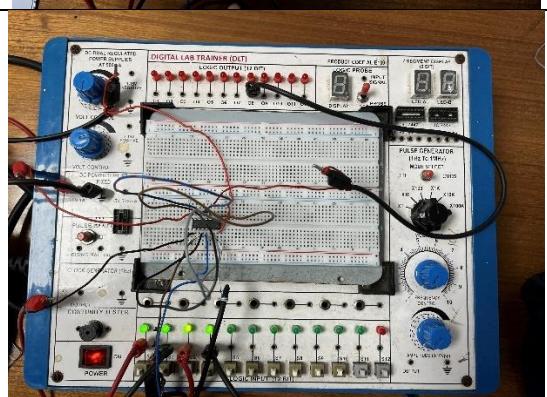
A=1, B=1, C=0, D=1; Y=0



A=1, B=1, C=1, D=0; Y=0



A=1, B=1, C=1, D=1; Y=0



Questions with answers for report writing:

- 1) What do you mean by universal gate?
- 2) What are the ICs required in this experiment?
- 3) Construct a circuit of output F, where $F=AB + BC + CA$, by using NAND gates only in the PSIM Software and show the output states for each of the available conditions.

Discussion and Conclusion:

The data was interpreted to assess the extent of compliance with the initial goal. Mistakes were made in control group selection and participant assignment, impacting the results. Improvements could have been made through rigorous control group selection, randomization, increased sample size, diverse sampling, consistent data collection, and double-blind procedures.

Reference(s):

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