FR. Conceicao Rodrigues College of Engineering
Department of Computer Engineering

1.Study of various Logic Gates using IC's

1. Course, Subject & Experiment Details

Academic Year	**2023-24**	**Estimated Time**	**Experiment No. 1– 02 Hours**
Course & Semester	**S.E. (Computers) – Sem. III**	**Subject Name**	**Digital Logic & Computer Organization and Architecture**
Chapter No.	**1**	**Chapter Title**	**Computer Fundamentals**
Experiment Type	**Hardware**	**Subject Code**	**CSC304**

Rubrics

Roll No.	Date of Perfomance:	**Timeline (2)**	**Practical Skill & Applied Knowledge (4)**	**Output (4)**	**Total (10)**
	Date of Submission:				

2. Aim & Objective of Experiment

To study and verify the truth table of logic gates.

Identify various ICs and their specification.

3. Problem Statement

a) Verify the truth table of AND, OR, NOT, NAND NOR, EXOR gates.

b) Realization of different gates using Universal gates

4. Hardware Required

.

A) Breadboard , Power Supply.

5. Components Required:

SL No.	COMPONENT	SPECIFICATION	QTY
1.	AND GATE	IC 7408	1
2.	OR GATE	IC 7432	1
3.	NOT GATE	IC 7404	1
4.	NAND GATE 2 I/P	IC 7400	1
5.	NOR GATE	IC 7402	1
6.	X-OR GATE	IC 7486	1
7.	NAND GATE 3 I/P	IC 7410	1
8.	PATCH CORD	-	

5. Brief Theoretical Description

Circuit that takes the logical decision and the process are called logic gates. Each gate has one or more input and only one output.
OR, AND and NOT are basic gates. NAND, NOR and X-OR are known as universal gates. Basic gates form these gates.

AND GATE:

The AND gate performs a logical multiplication commonly known as AND function. The output is high when both the inputs are high. The output is low level when any one of the inputs is low.

OR GATE:

The OR gate performs a logical addition commonly known as OR function. The output is high when any one of the inputs is high. The output is low level when both the inputs are low.

NOT GATE:

The NOT gate is called an inverter. The output is high when the input is low. The output is low when the input is high.

NAND GATE:

The NAND gate is a contraction of AND-NOT. The output is high when both inputs are low and any one of the input is low .The output is low level when both inputs are high.

NOR GATE:

The NOR gate is a contraction of OR-NOT. The output is high when both inputs are low. The output is low when one or both inputs are high.

XOR GATE:

The output is high when any one of the inputs is high. The output is low when both the inputs are low and both the inputs are high.

PROCEDURE:

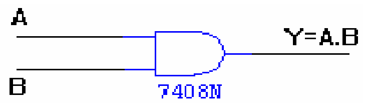
(i) Make the connections given as per circuit diagram.

(ii) Logical inputs are given as per circuit diagram.

(iii) Observe the output and verify the truth table.

AND GATE:

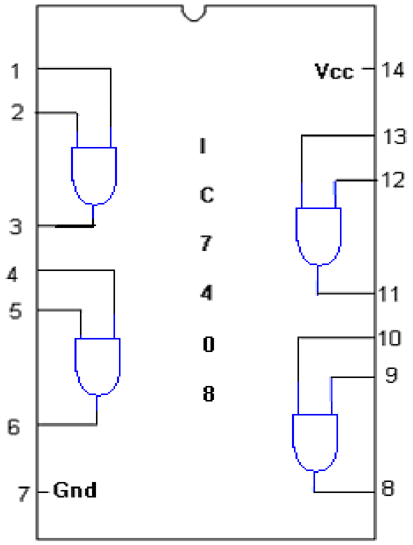
SYMBOL:



TRUTH TABLE

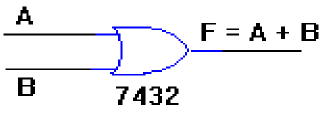
A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

PIN DIAGRAM:



OR GATE:

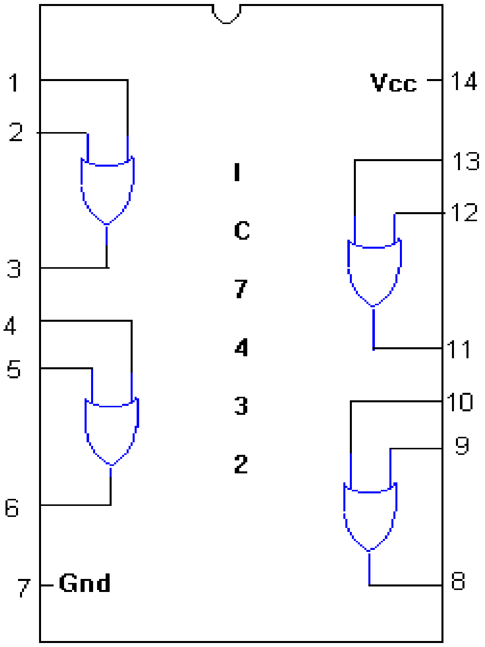
SYMBOL :



TRUTH TABLE

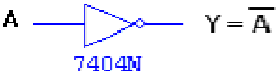
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

PIN DIAGRAM :

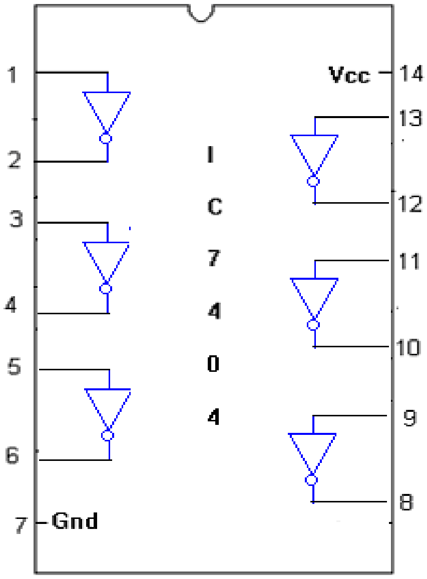


NOT GATE:

SYMBOL:



PIN DIAGRAM:

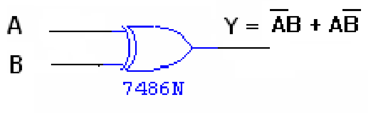


TRUTH TABLE :

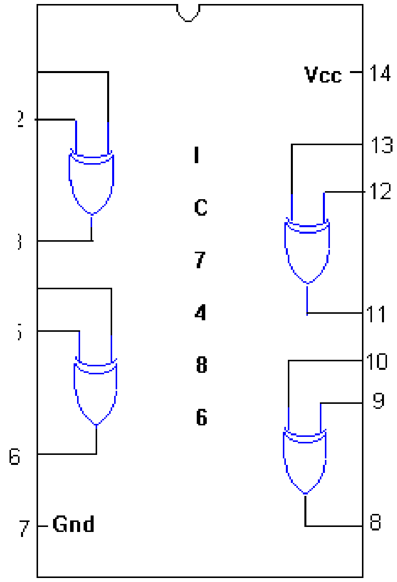
A	$\overline{A}$
0	1
1	0

X-OR GATE :

SYMBOL :



PIN DIAGRAM :

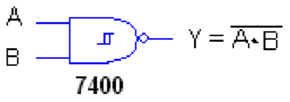


TRUTH TABLE :

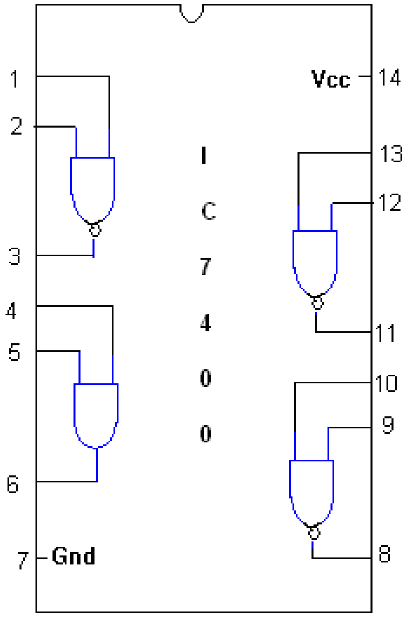
A	B	$\overline{A}B + A\overline{B}$
0	0	0
0	1	1
1	0	1
1	1	0

2-INPUT NAND GATE:

SYMBOL:



PIN DIAGRAM:

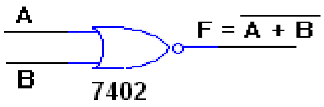


TRUTH TABLE

A	B	$\overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

NOR GATE:

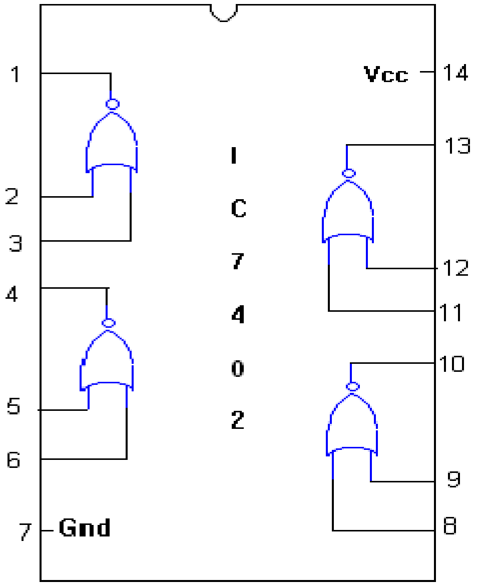
SYMBOL :



TRUTH TABLE

A	B	$\overline{A+B}$
0	0	1
0	1	1
1	0	1
1	1	0

PIN DIAGRAM :



7. Conclusion:

In this experiment, we verified logic gate truth tables using ICs like the 7400 series (e.g., 7400, 7402, 7404, 7486), gaining practical insights into digital logic behavior and common IC specifications.

8. Post-lab:

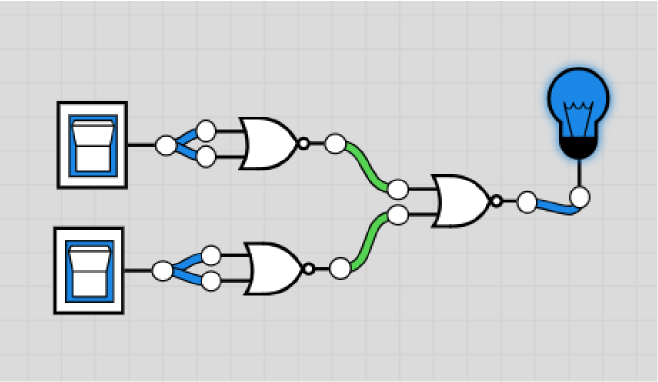
1. Why NAND and NOR gates are called as universal gates.

⇒ NAND and NOR gates are referred to as universal gates due to their capability to implement any logic function. By properly connecting these gates, it's possible to replicate the behavior of other basic gates like AND, OR, and NOT. This property simplifies circuit design, as complex functions can be realized using just one type of gate, leading to more efficient and versatile digital systems.

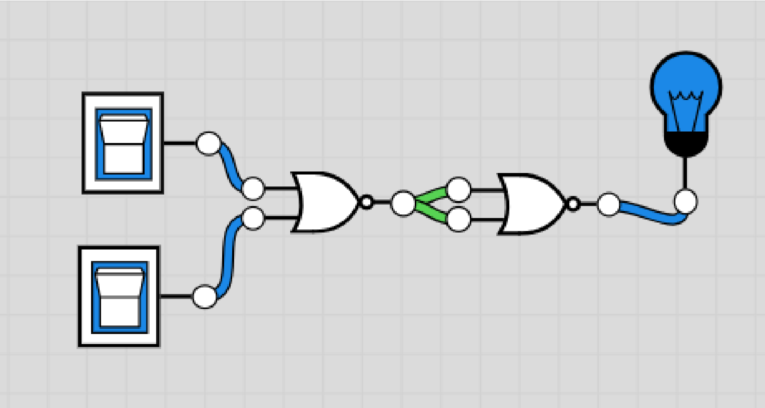
2. Realize AND, OR & Ex-OR gate using NOR and NAND gates.

Using NOR GATES:

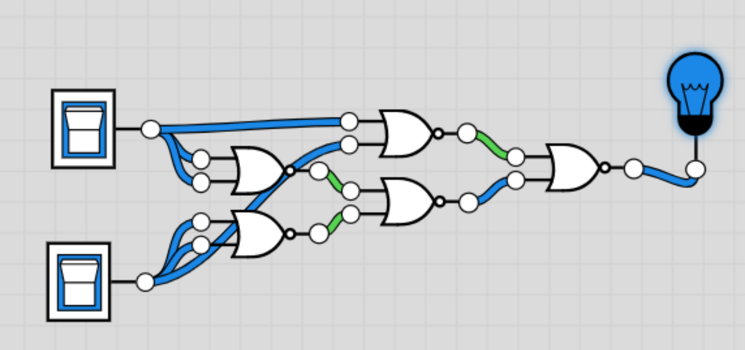
1.AND gate:A.B = ((A.B)`)`
= (A` + B`)`



2. OR gate : (A + B) = ((A + B)`)`

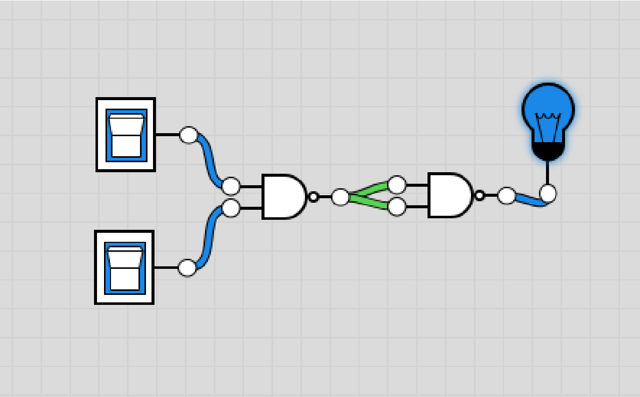


3. ExoR : A.B` + B.A` = AB + A`B`
= ((A+B)`+(A` + B`)`)`

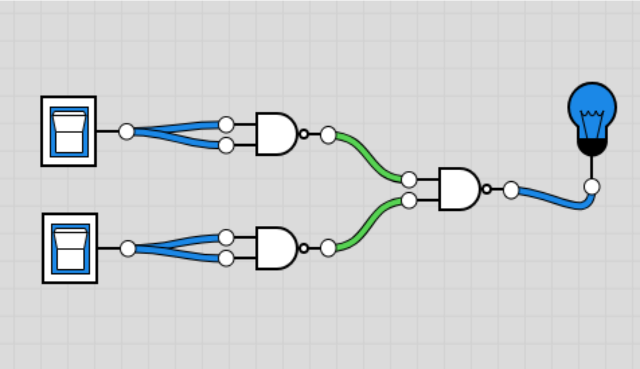


Using NAND gates:

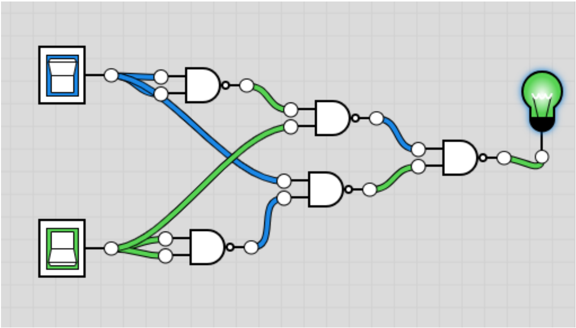
1.AND gate: A.B = ((A.B)`)`



2.OR gate: A + B = ((A + B)`)`
= (A`.B`)`

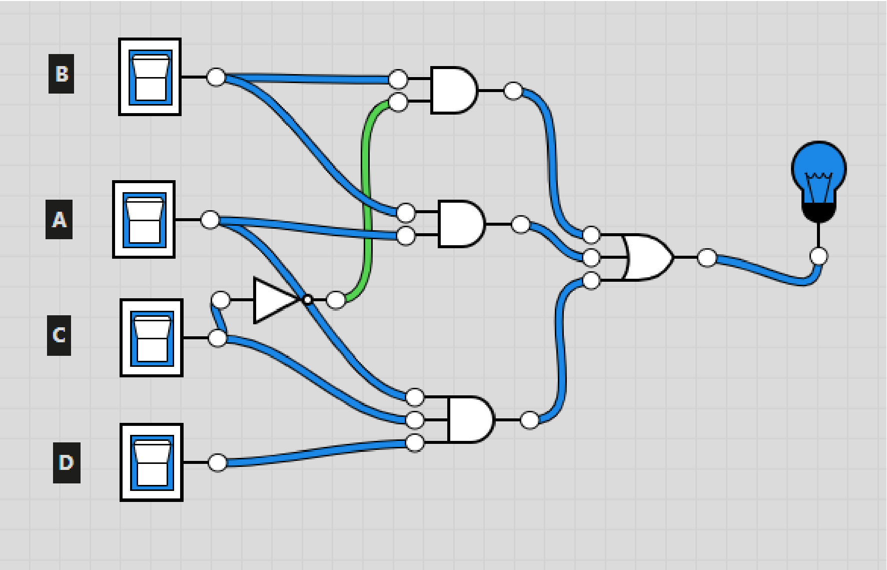


3. ExOR gate: AB` + A`B = ((A`B)`. (A.B`)`)`



3. Draw the logic diagram corresponding to the following Boolean expression without simplifying them.

1) B C’ + AB + ACD



2) (A + B) (C + D) (A’ + B + D)