North South University

CSE231L

Experiment # 1

Name of Experiment: **Digital Logic Gates and Boolean Functions**

Date of Performance: 2 October, 2019

Date of Submission: 16 October, 2019

Section: 13

Group: 3

Submitted To: Farhana Saleh

Submitted By:

ID	Name
1530486042	Abdul Zabbar
1711038042	MD. ASHRAFUL KABIR
1712747042	Ashik Iqbal
1731046042	Nahian -Al Sabri

Objectives

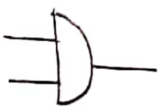
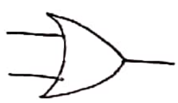
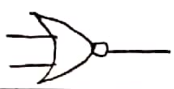
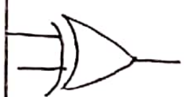
- We have to study the basic logic gates - AND, OR, NOT, NAND, NOR, XOR.
- We have to get acquinted with the representation of Boolean functions using truth tables, logic diagrams and Boolean Algebra.
- We have to prove the extension of inputs of AND and OR gates using the associate law.
- We have to become familiarized with combinational logic circuits.

Equipments

- IC 7400 Quadruple 2-input NAND gates.
- IC 7402 Quadruple 2-input NOR gates.
- IC 7404 Hex Inverters (NOT gates).
- IC 7408 Quadruple 2-input AND gates.
- IC 7432 Quadruple 2-input OR gates.
- IC 7486 Quadruple 2-input XOR gates.
- Trainer Board.
- Wires.

Theory

Logic Gates: Logic gates are the elementary building blocks of digital circuits. Digital logic gates operate at two discrete voltage levels representing the binary values 0 (logical LOW) and 1 (logical HIGH).

Gate	Description	IC#	Symbol
AND	Multi-input circuit producing an output of 1 if all inputs are 1	7408	
OR	Multi-input circuit producing an output of 1 when any of its inputs is 1.	7432	
NOT	Single-input circuit that inverts the input (also called an Inverter). The input is 0 if the input is 1 and vice versa.	7404	
NAND	AND followed by an Inverter.	7400	
NOR	OR followed by an Inverter.	7402	
XOR	The Exclusive-OR or Ex-OR is a two-input circuit that produces an output of 0 if both inputs are same and 1 if the inputs are different.	7486	

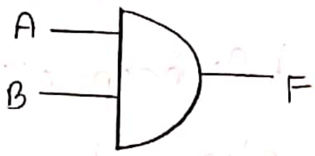
Scanned by CamScanner

Boolean Algebra: Boolean algebra is a branch of mathematical logic that formalizes the relation between variables that take the truth values of true or false, denoted by 1 and 0 respectively. It is fundamental in the development of digital electronics.

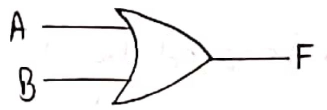
Postulates and Theorems		Name
$A+0=A$	$A\cdot 1=A$	Identity
$A+A'=1$	$A\cdot A'=0$	
$A+A=A$	$A\cdot A=A$	
$A+1=1$	$A\cdot 0=0$	
$(A')'=A$		Involution
$A+B=B+A$	$AB=BA$	Commutative
$A+(B+C)=(A+B)+C$	$A(BC)=(AB)C$	Associative
$A(B+C)=AB+AC$	$A+BC=(A+B)(A+C)$	Distributive
$(A+B)'=A'B'$	$(AB)'=A'+B'$	De Morgan
$A+AB=A$	$A(A+B)=A$	Absorption.

Circuit diagram

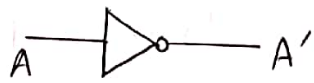
Experiment 1:



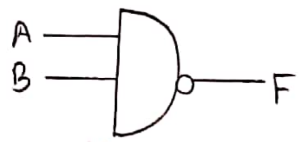
AND GATE



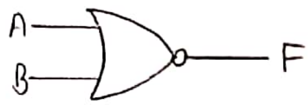
OR GATE



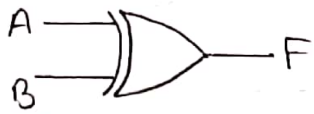
NOT GATE



NAND GATE

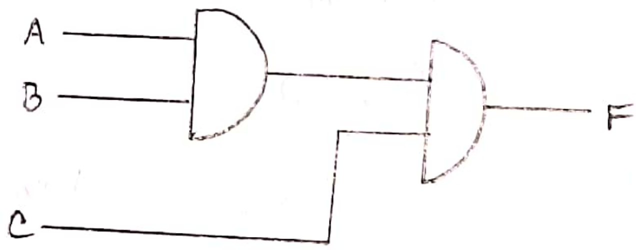


NOR GATE



XOR GATE

Experiment 2:



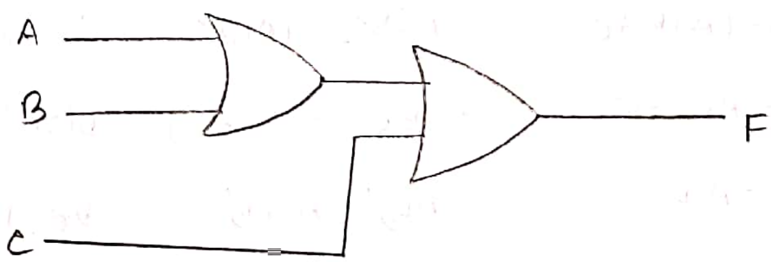


Figure: Extension of inputs of AND and OR Gates.

Experiment 3:

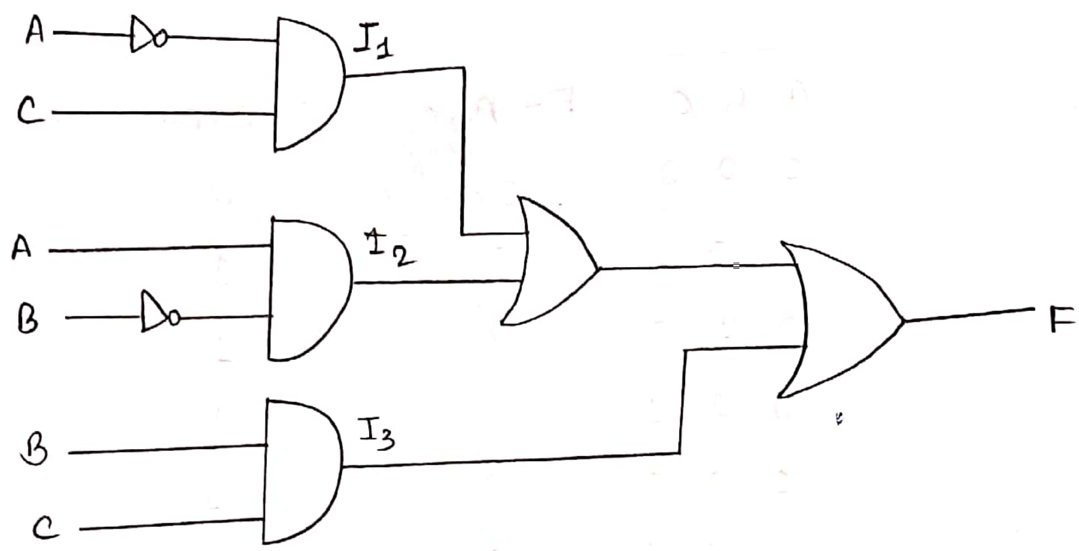


Figure: Logic Diagram for the given Boolean function.

Results

F.1| Introduction to Basic Logic Gates

Input A	Input B	AND $F=A.B$	OR $F=A+B$	NAND $F=\overline{A.B}$	XOR $F=A\oplus B$	NOR $F=\overline{A+B}$
0	0	0	0	1	0	1
0	1	0	1	1	1	0
1	0	0	1	1	1	0
1	1	1	1	0	0	0

Input A	NOT $F=\bar{A}$
0	1
1	0

Table: Truth table of Logic Gates.

F.2| Constructing 3-input AND & OR gates from 2-input AND & OR gates.

A B C	$F = ABC$	$F = A+B+C$
0 0 0	0	0
0 0 1	0	1
0 1 0	0	1
0 1 1	0	1
1 0 0	0	1
1 0 1	0	1
1 1 0	0	1
1 1 1	1	1

Table: Truth tables for 3-input And and OR

F.3| Implementation of Boolean Functions

$$F = A'C + AB' + BC.$$

A B C	$I_1 = A'C$	$I_2 = AB'$	$I_3 = BC$	$F = I_1 + I_2 + I_3$
0 0 0	0	0	0	0
0 0 1	1	0	0	1
0 1 0	0	0	0	0
0 1 1	1	0	1	1
1 0 0	0	1	0	1
1 0 1	0	1	0	1
1 1 0	0	0	0	0
1 1 1	0	0	1	1

Table: Truth table for the given boolean function.

Questions

1) IC 7408 Quadruple AND gates, IC 7404 Hex inverters and IC 7402 Quadruple NOR gates are the names of the ICs for AND, NOT and NOR gates.

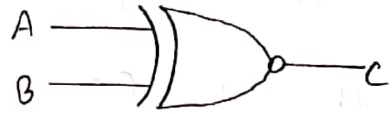
⇒ For 17 AND gates we need $(17/4 = 4.25) = 5$ AND Gates
⇒ For 22 NOT gates we need $(22/6 = 3.67) = 4$ NOT Gates
⇒ For 18 NOR gates we need $(18/4 = 4.5) = 5$ NOR gates

2) If the +5 port of our trainer board stops working, then we can power our logic ICs from any of the switches of the trainer board.

3) Truth table: A truth table is a tabular representation of all the combinations of values for inputs and their corresponding outputs. It is a mathematical table that shows all possible outcomes that would occur from all possible scenerios that are considered factual, hence the name. Truth tables are usually

used for logic problems as in Boolean algebra and electronic circuits.

Input		Output
A	B	A XNOR B
0	0	1
0	1	0
1	0	0
1	1	1



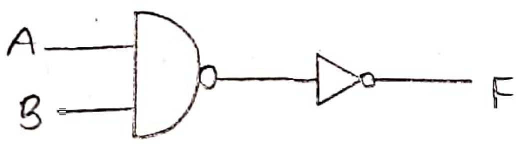
$C = A \odot B$

B — Input 2

A — Input 1

C — Output

4)

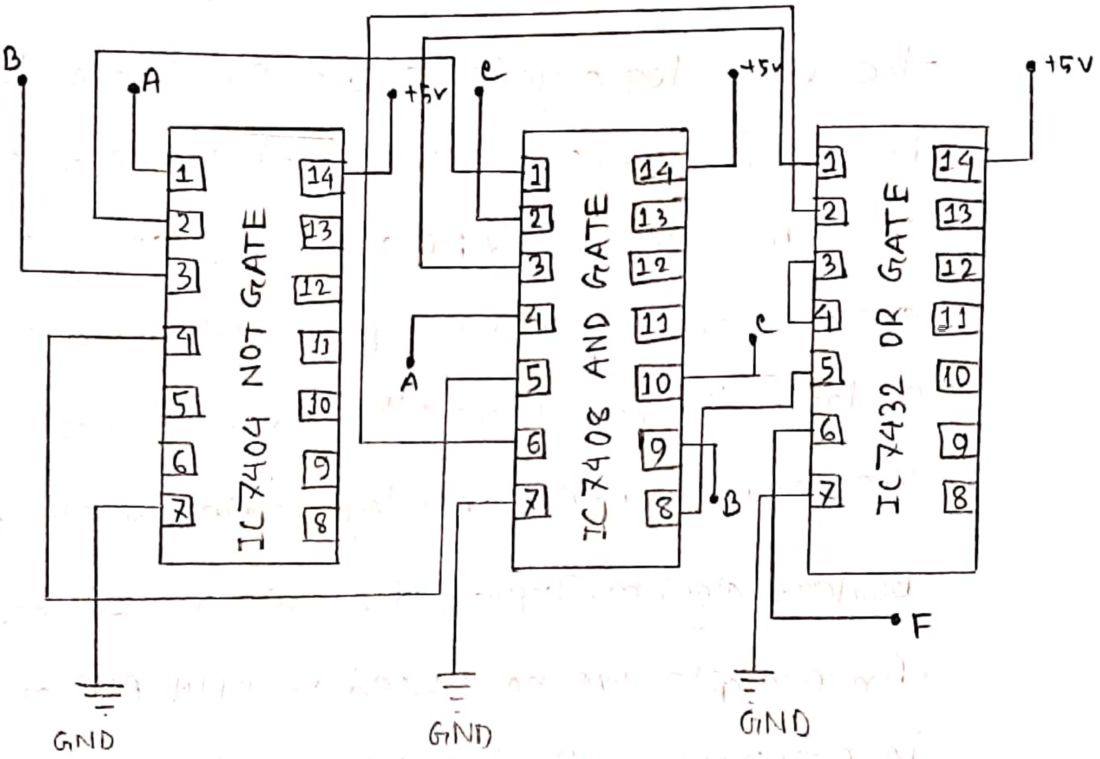


The final output will represent AND Gate.

The name of the Boolean Algebra theorem that can be used to find the answer is De Morgan's Law. According to this theorem:

$$(A+B)' = A'B'$$
$$(AB)' = A'+B'$$

5)



$F = A'C + AB' + BC$

Figure: IC diagram for the logic circuit.

Discussion

Because of human error and equipment error, we didn't get our expected results. In the lab, we observed that completing the truth table from the given inputs we can make a logic circuit of it. In experiment -1, we used ~~to~~ the NOT, OR, AND, NAND, XOR and NOR gates. We used their IC's number and connected the wires based on the truth table and logic circuit. Thus our experiment-1 verifies the basic logic gates and their truth tables. Again, in experiment-2, we have made 3-input AND and OR gates from 2-input AND and OR gates by giving proper connection and from experiment-3 we have implemented a logic diagram from the given boolean function. So we have studied how logic gates work by using Boolean algebra. Input of a gate can be more than 2 for example we have used 3-input AND and OR gates in experiment-2. We also faced some errors while doing the experiment-3. At first, we could not implement the logic diagram for not giving a proper connection and we found the AND gate to be heated as we did not off the power button after completing the work. Thus, our experiment is verified.

CSE231L – Lab 1 – Digital Logic Gates and Boolean Functions

Data Sheet:

Instructor's Signature: ....

Section: 13	Group No.: 03	Date: 2 oct, 2019

F.1 Introduction to Basic Logic Gates

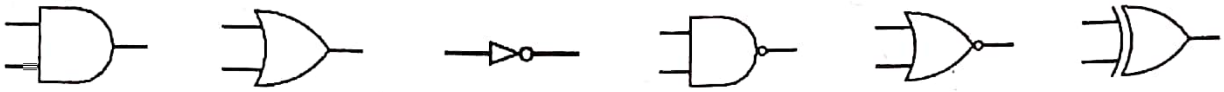


Figure F.1.1: Pin configurations of gates in ICs

Input A B	AND $F = A \cdot B$	OR $F = A + B$	NAND $F = \overline{A \cdot B}$	XOR $F = A \oplus B$	NOR $F = \overline{A + B}$
0 0	0	0	1	0	1
0 1	0	1	1	1	0
1 0	0	1	1	1	0
1 1	1	1	0	0	0

Input A	NOT $F = \overline{A}$
0	1
1	0

Table F.1.1: Truth Table of Logic Gates

F.2 Constructing 3-input AND & OR gates from 2-input AND & OR gates

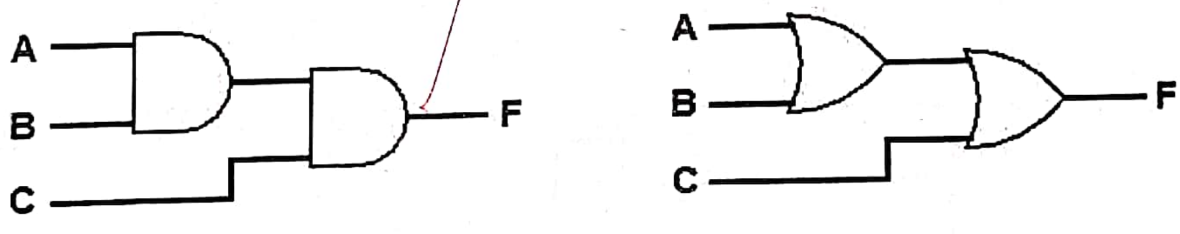


Figure F.2.1: Extension of inputs of AND and OR gates

$A\ B\ C$	$F = ABC$	$F = A + B + C$
0 0 0	0	0
0 0 1	0	1
0 1 0	0	1
0 1 1	0	1
1 0 0	0	1
1 0 1	0	1
1 1 0	0	1
1 1 1	1	1

Table F.2.1: Truth Tables for 3-input AND and OR

F.3 Implementation of Boolean Functions

Results.

$$F = A'C + AB' + BC$$

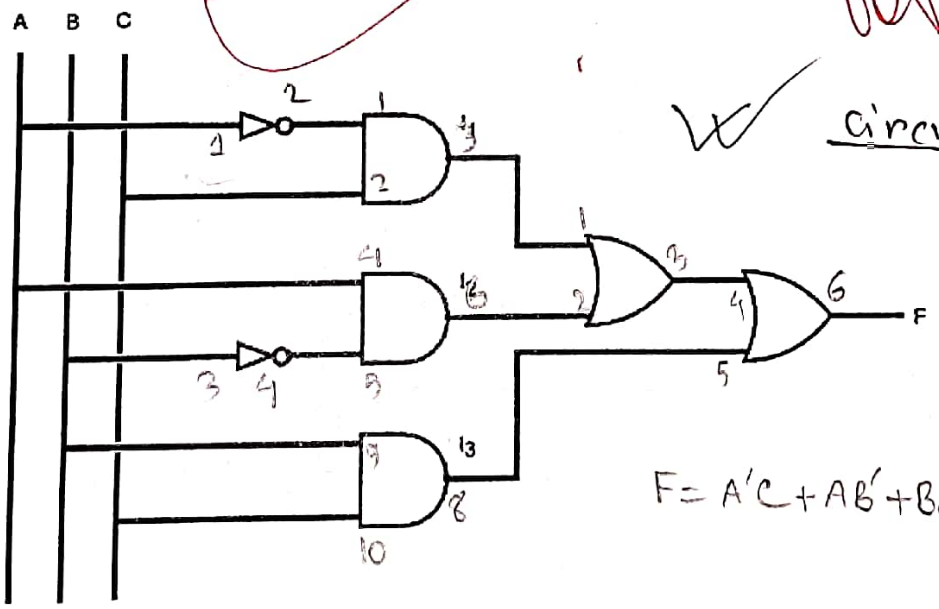
$A\,B\,C$	$I_1 = A'C$	$I_2 = AB'$	$I_3 = BC$	$F = I_1 + I_2 + I_3$
0 0 0	0	0	0	0
0 0 1	1	0	0	1
0 1 0	0	0	0	0
0 1 1	1	0	1	1
1 0 0	0	1	0	1
1 0 1	0	1	0	1
1 1 0	0	0	0	0
1 1 1	0	0	1	1

A - 6
B - 5
C - 4

Table F.3.1: Truth Table for the given Boolean Function

2.10.19

circuit diagram



F= A'C+AB'+BC.

Figure F.3.1: Logic Diagram for the given Boolean Function