Experiment / Assignment / Tutorial No. 1

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of the Staff In-charge with date

Batch: A3	Roll No.: 16010122151	Experiment / assignment / tutorial No.: 1

Title: Basic Gates & Universal Gates

Objective: To study the basic gates: AND, OR, NOT and universal gates: NAND, NOR, XOR, XNOR

Expected Outcome of Experiment:

CO1: Recall basic gates and binary, octal & hexadecimal calculations and conversions.

Books/ Journals/ Websites referred:

- Vlab Link: http://vlabs.iitkgp.ernet.in/dec/#
- R. P. Jain, "Modern Digital Electronics", Tata McGraw Hill
- http://www.ee.surrey.ac.uk/Projects/Labview/gatesfunc/
- http://www.electronics-tutorials.ws/boolean/bool_6.html

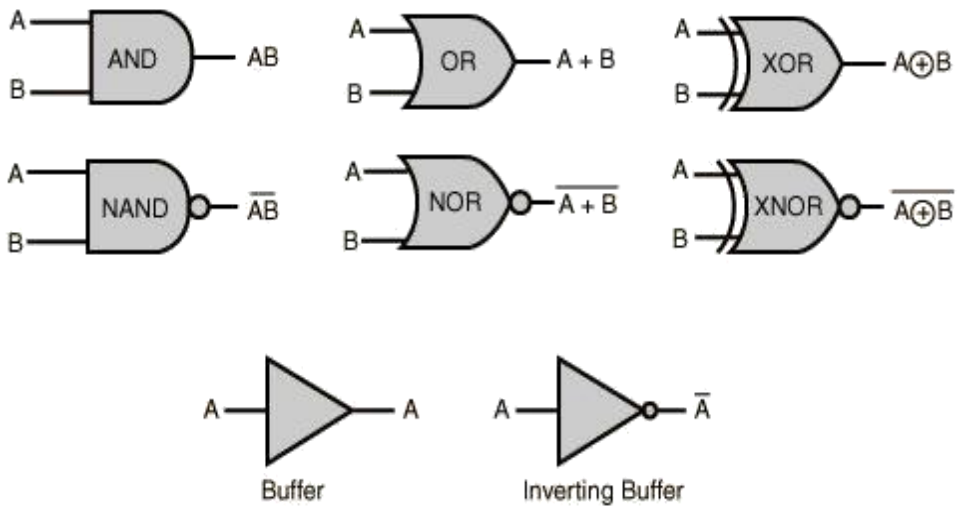
Pre Lab/ Prior Concepts:

Gate is a logic circuit with one or more inputs but only one output. Gates are digital (two state) circuit because the input & output are either low or high. Gates provide high output for certain combinations of input & for other combinations the output is low. Total number of combinations for a gate is 2^n; where n is number of input.

Classification: The two types of gate are:

1. **Basic or Fundamental Gates:**
2. **Derived Gates:**

Symbols of gates



TYPE OF IC	SPECIFICATION
IC 7408	AND: Quad 2 input AND TTL IC HEX
IC 7432	OR: Quad 2 input OR gate TTL IC
IC 7404	NOT: Inverter TTL IC
IC 7402	NOR: Quad 2 input NOR gate TTL IC
IC 7400	NAND: Quad 2 input NAND gate TTL IC
IC 7486	EX-OR: Quad 2 input XOR gate TTL IC

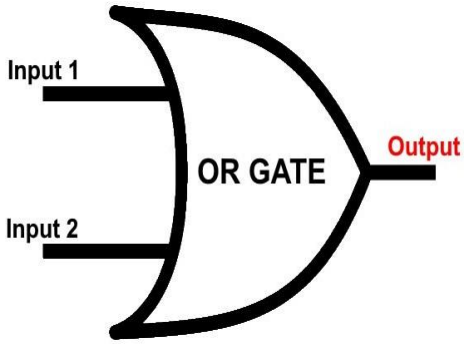
Implementation Details:

Basic Gates

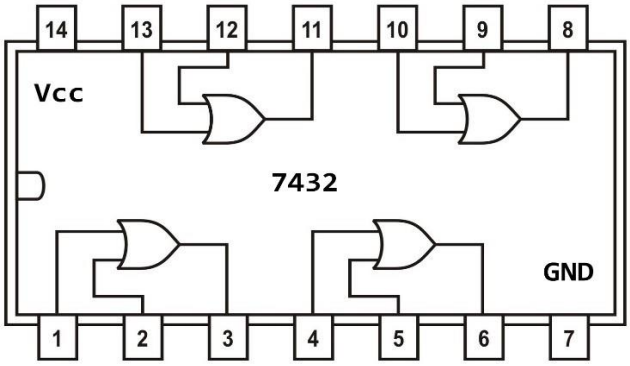
1. **OR gate:** The OR gate has two or more inputs but only 1 output. If any or all the inputs are high, the output is high. If all the inputs are low, the output is low.

$$Y = A + B$$

Symbol for OR gate



Pin Diagram For IC 7432



The truth table for OR operations are:

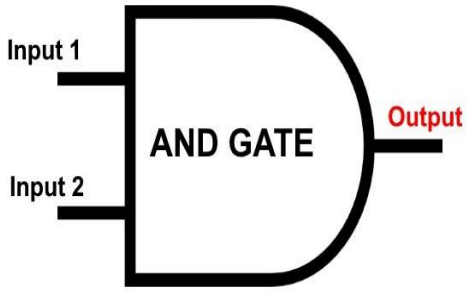
OR Truth Table

Inputs		Output
A	B	Y = A+B
0	0	0
0	1	1
1	0	1
1	1	1

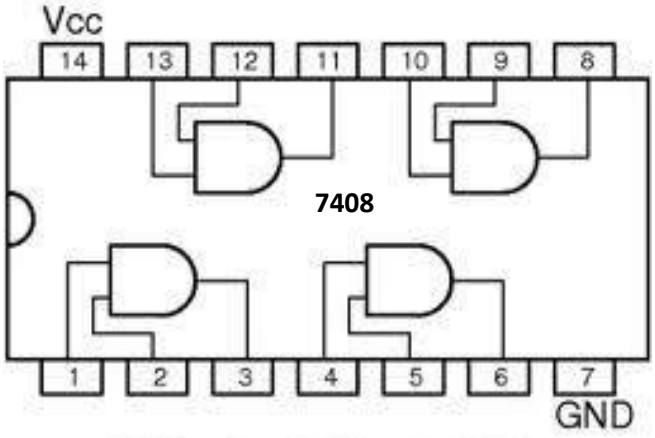
2. **AND gate:** The AND gate has two or more inputs but only one output. If any or all inputs are high then output is also high

$$Y = A.B$$

Symbol for AND gate



Pin Diagram For IC 7408



The truth table for AND operations are:

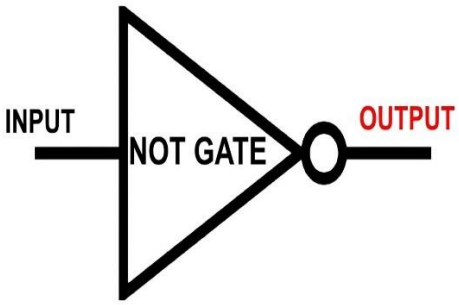
AND Truth Table

Inputs		Output
A	B	Y = A.B
0	0	0
0	1	0
1	0	0
1	1	1

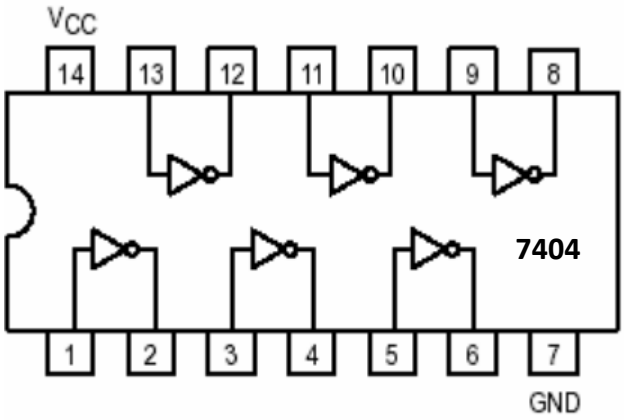
3. **NOT gate:** The Not gate is a gate with only one input and one output. The output is always in opposite state of an input. A NOT gate is also called as Inverter because it performs inversion.

$$Y = \overline{A}$$

Symbol for NOT gate



Pin Diagram For IC 7404



The truth table for NOT operations is:

NOT gate Truth Table

Input A	Output $X=\overline{A}$
0	1
1	0

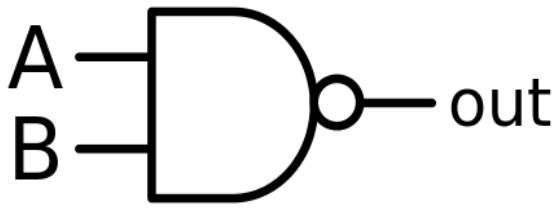
Derived Gates/Universal Gates

- NAND gate
- NOR gate
- EX-OR gate
- EX-NOR gate

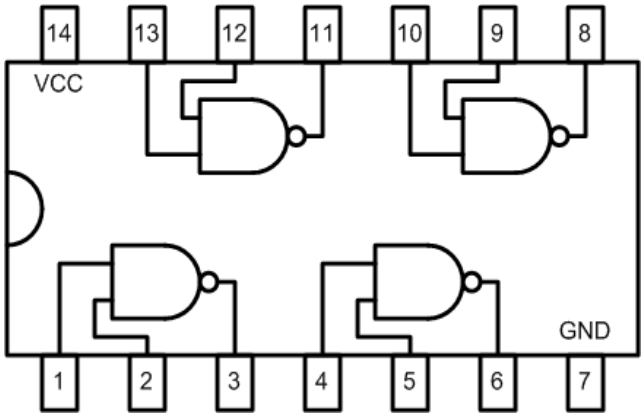
1. **NAND gate:** This is a NOT-AND gate which is equal to an AND gate followed by a NOT gate. The outputs of all NAND gates are high if any of the inputs are low. The symbol is an AND gate with a small circle on the output. The small circle represents inversion.

$$Y= \overline{A.B}$$

Symbol



Pin Diagram for IC 7400



The truth table for NAND operations is:

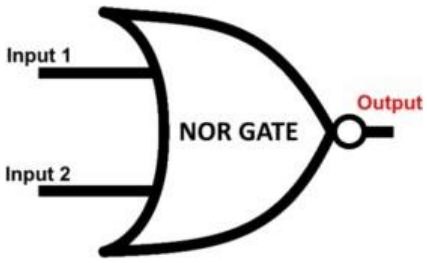
Truth Table

Input A	Input B		Output Q
0	0		1
0	1		1
1	0		1
1	1		0

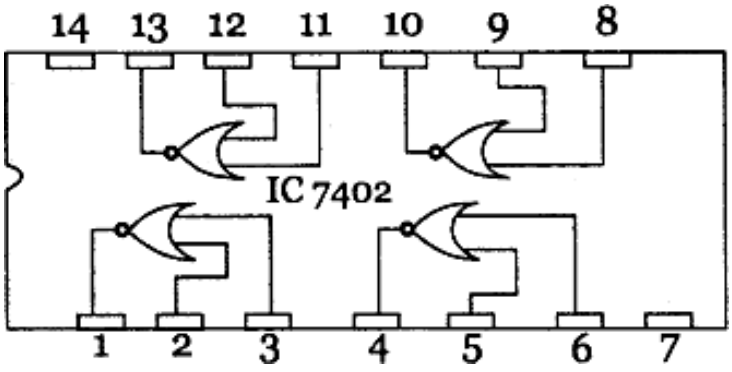
2. **NOR gate:** This is a NOT-OR gate which is equal to an OR gate followed by a NOT gate. The outputs of all NOR gates are low if any of the inputs are high. The symbol is an OR gate with a small circle on the output. The small circle represents inversion.

$$Y = \overline{A + B}$$

Symbol for NOR gate



Pin Diagram For IC 7402



The truth table for NOR operations are:

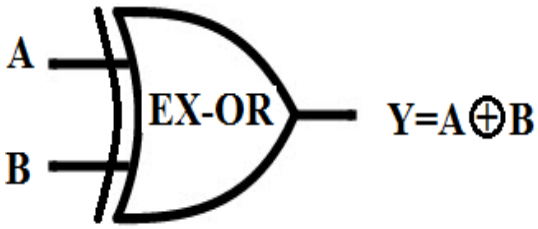
TRUTH TABLE

INPUT		OUTPUT
A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

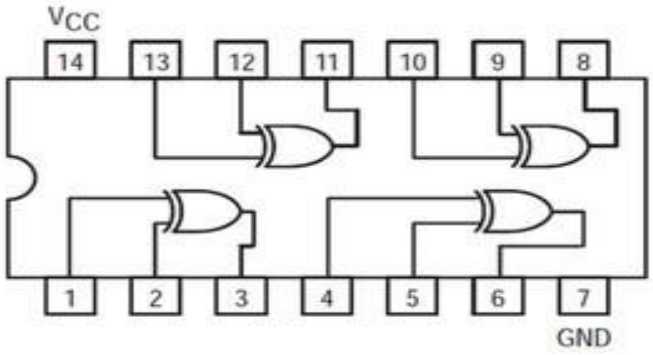
3. **EX-OR gate**: The 'Exclusive-OR' gate is a circuit which will give a high output if either, but not both, of its two inputs are high. An encircled plus sign (⊕) is used to show the EX-OR operation

$$Y = A \oplus B$$

Symbol for Ex-OR gate



Pin Diagram For IC 7486



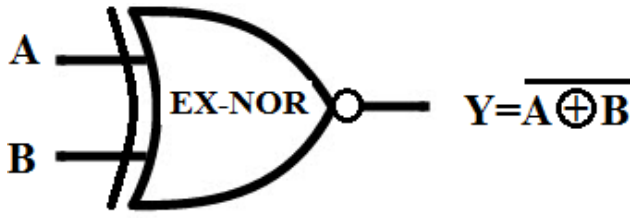
The truth table for XOR operations is:

Inputs		Output
A	B	Y=A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

4. **EX-NOR gate**: The 'Exclusive-NOR' gate circuit does the opposite to the EOR gate. It will give a low output if either, but not both, of its two inputs are high. The symbol is an EXOR gate with a small circle on the output. The small circle represents inversion

Y= $\overline{A \oplus B}$

Symbol for Ex-NOR gate



The truth table for XNOR operations is:

Inputs		Output
A	B	$Y=\overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

Implementation Using NAND Gate

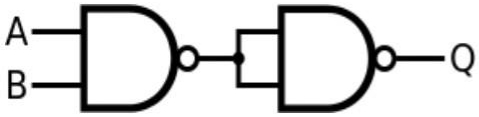
NOT GATE

A NOT gate is made by joining the inputs of a NAND gate together. Since a NAND gate is equivalent to an AND gate followed by a NOT gate, joining the inputs of a NAND gate leaves only the NOT gate.

A	Q
0	1
1	0

AND GATE

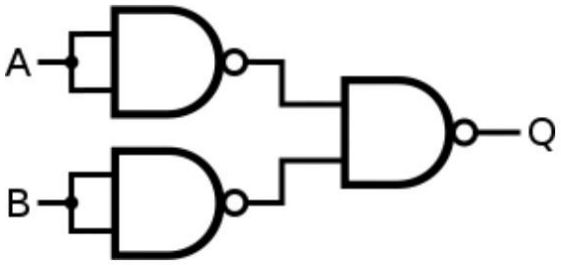
An AND gate is made by inverting the output of a NAND gate as shown below.



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

OR GATE

If the truth table for a NAND gate is examined or by applying De Morgan's Laws, it can be seen that if any of the inputs are 0, then the output will be 1. To be an OR gate, however, the output must be 1 if any input is 1. Therefore, if the inputs are inverted, any high input will trigger a high output.



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

IMPLEMENTATION USING NOR GATE

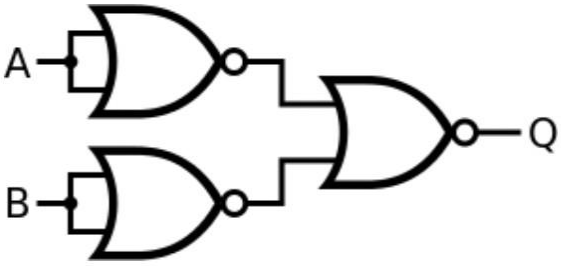
NOT GATE

This is made by joining the inputs of a NOR gate. As a NOR gate is equivalent to an OR gate leading to NOT gate, this automatically sees to the "OR" part of the NOR gate, eliminating it from consideration and leaving only the NOT part.

A	Q
0	1
1	0

AND GATE

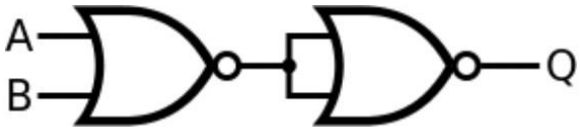
An AND gate gives a 1 output when both inputs are 1. Therefore, an AND gate is made by inverting the inputs of a NOR gate. Again, note that a NOT gate is equivalent to a NOR with its inputs joined.



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

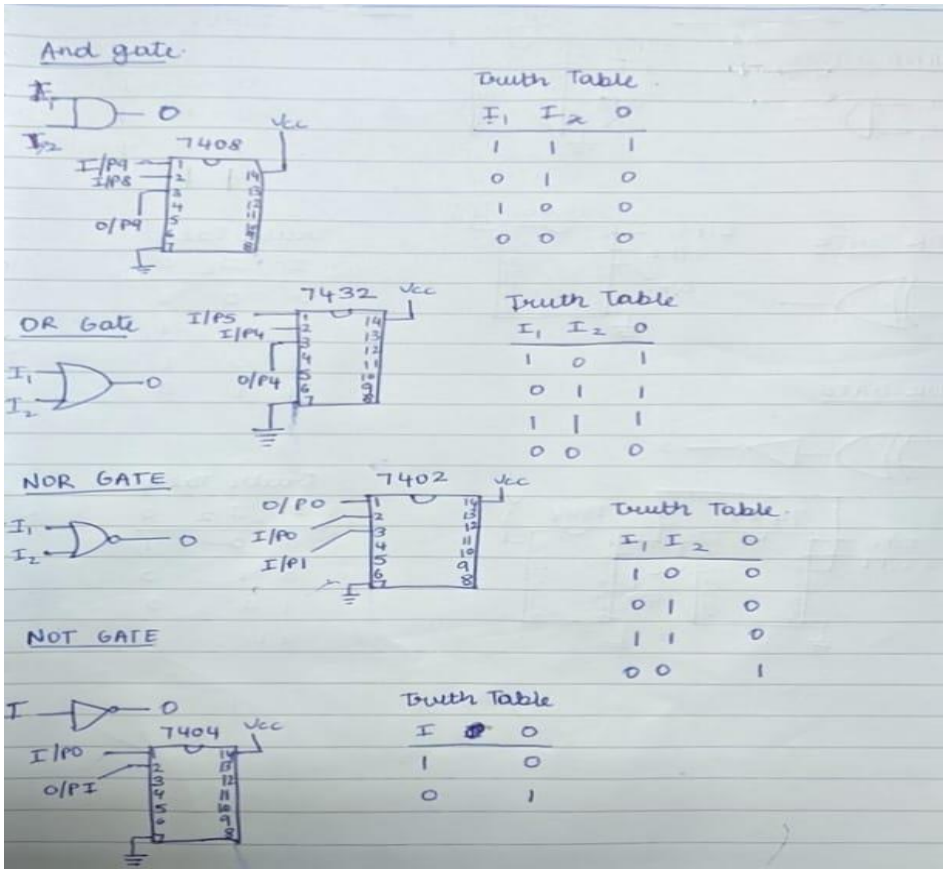
OR GATE

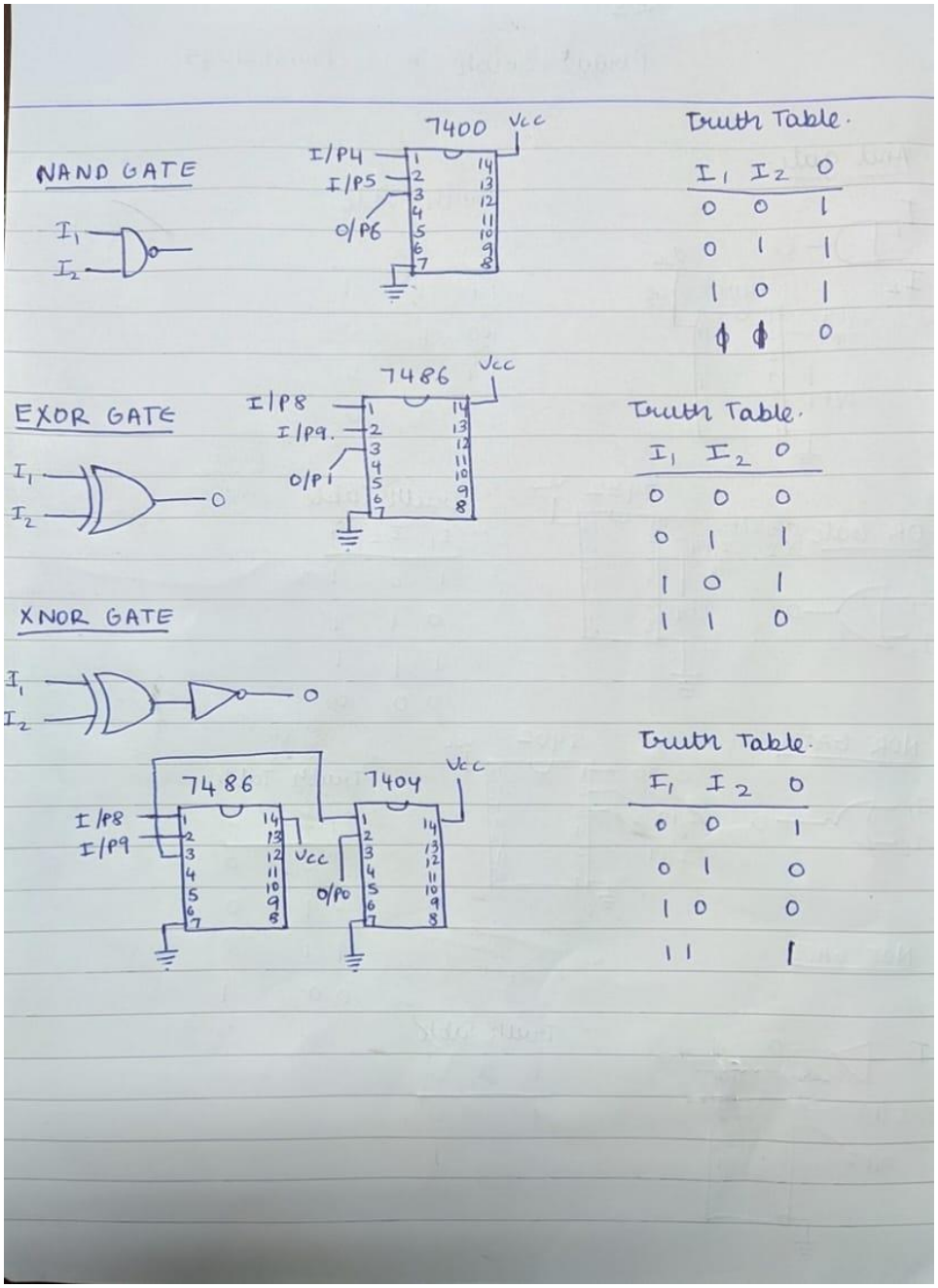
An OR gate is made by inverting the output of a NOR gate. Note that we already know that a NOT gate isequivalent to a NOR gate with its inputs joined.

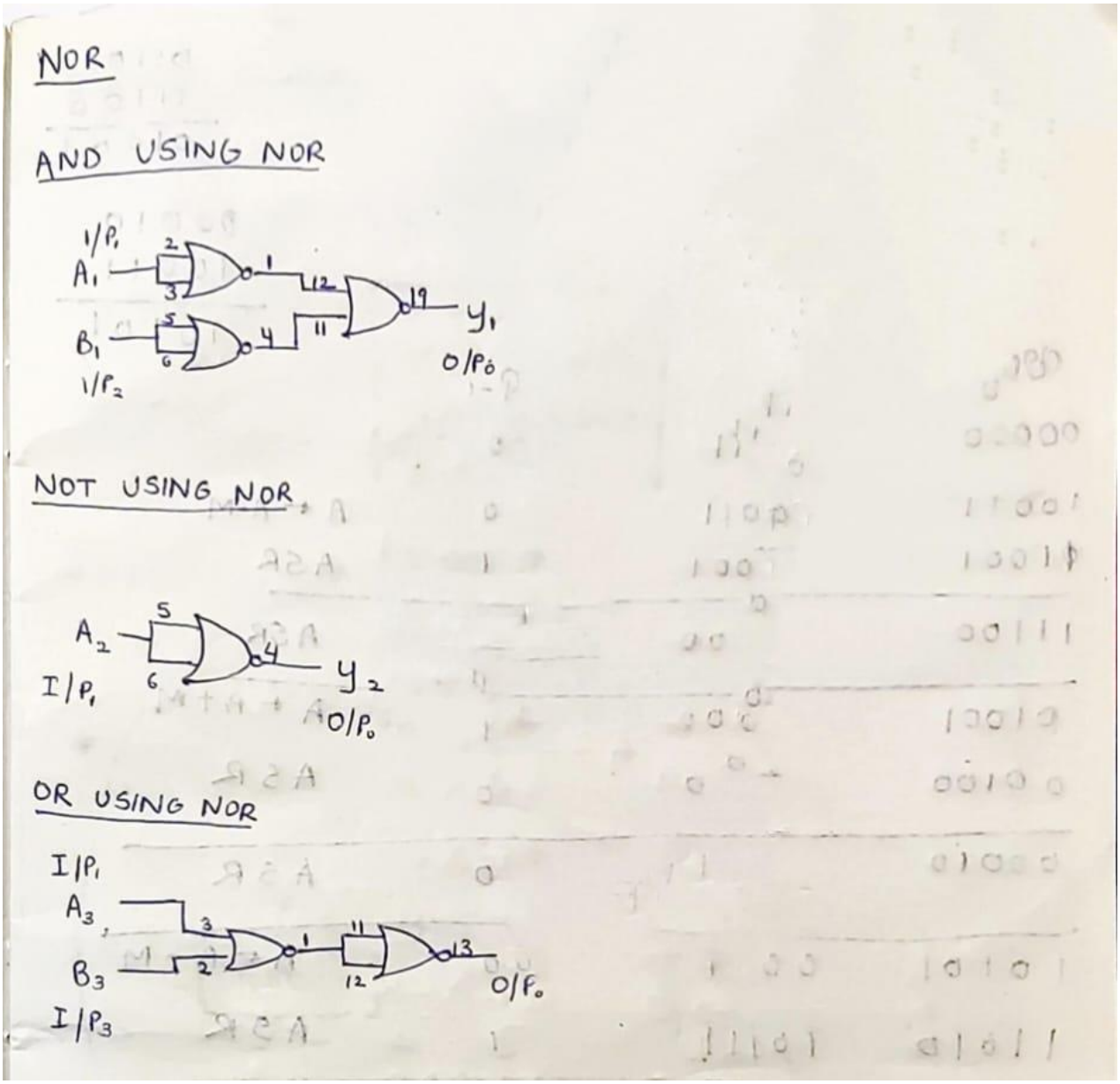


A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Lab Work:




NAND GATE
7400
Vcc
I/P4
I/P5
O/P6
Truth Table.
I1 I2 O
0 0 1
0 1 1
1 0 1
1 1 0
EXOR GATE
7486
I/P8
I/P9.
O/P1
Truth Table.
I1 I2 O
0 0 0
0 1 1
1 0 1
1 1 0
XNOR GATE
7486
7404
I/P8
I/P9
Vcc
O/P0
Truth Table.
I1 I2 O
0 0 1
0 1 0
1 0 0
1 1 1



Conclusion:

Studied about logic gates and how they work in combinations. The basic gates: AND, OR, NOT and universal gates: NAND, NOR, XOR, XNOR

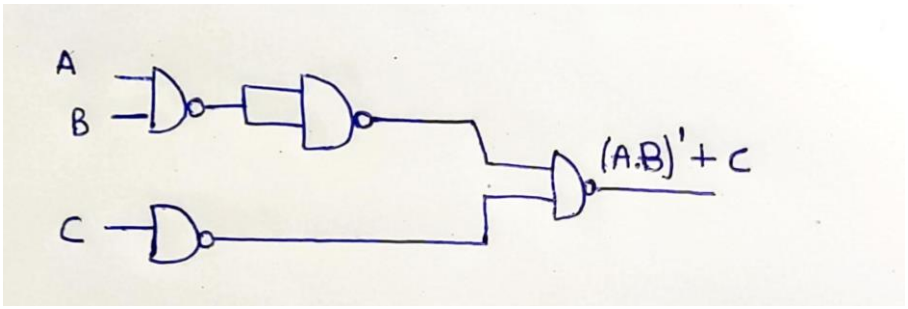
Post Lab Descriptive Questions

1. Verify the expression (A·B)' + C by:
   a) Using NAND Gate directly.
   b) Using AND & NOT gate consecutively.

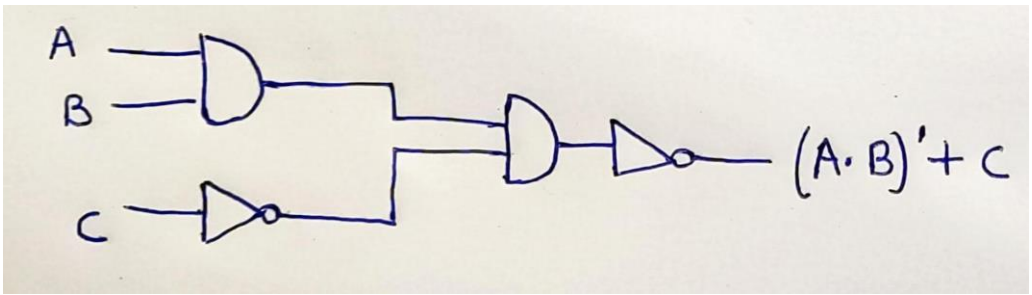
Truth Table

A	B	C	A.B	(A.B)’	(A.B)’ + C
0	0	0	0	1	1
0	0	1	0	1	1
0	1	0	0	1	1
0	1	1	0	1	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	0	1

a) Using NAND Gate directly



b) Using AND & NOT gate consecutively.

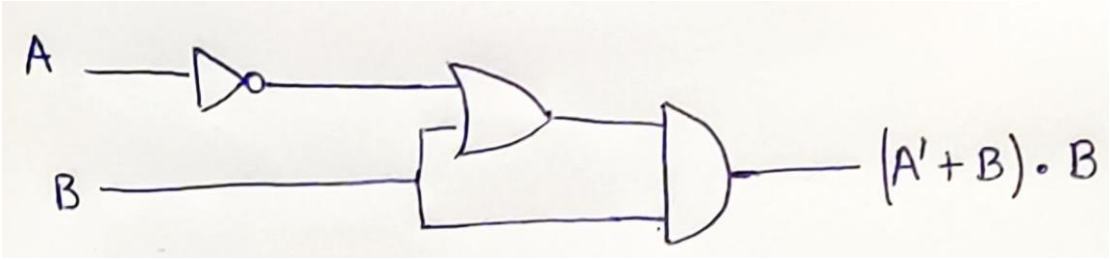


2. Implement the following expressions using combination of gates:

a) (A'+B)·B

Truth Table

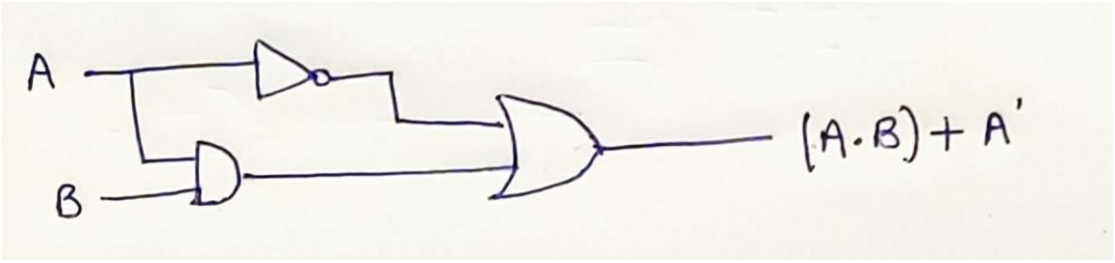
A	B	A’	A’+B	(A’+B).B
0	0	1	1	0
0	1	1	1	1
1	0	0	0	0
1	1	0	1	1



b) (A·B)+A'

Truth Table

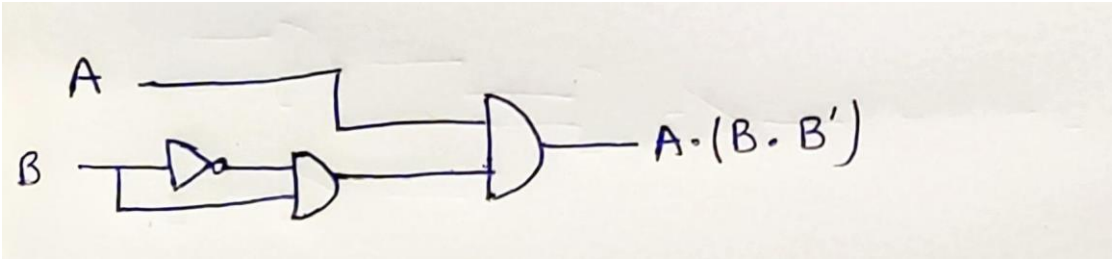
A	B	A’	A.B	(A.B) + A’
0	0	1	0	1
0	1	1	0	1
1	0	0	0	0
1	1	0	1	1



c) A· (B·B')

Truth Table

A	B	B’	B.B’	A.(B.B’)
0	0	1	0	0
0	1	0	0	0
1	0	1	0	0
1	1	0	0	0



d) (A'⊕B)·A

Truth Table

A	B	A’	A’ ⊕ B	(A’ ⊕ B).A
0	0	1	1	0
0	1	1	0	0
1	0	0	0	0
1	1	0	1	1