TITLE - Verifications of Basic and Universal Gate integrated Circuits

OBJECTIVE - To verify the outputs of all Logic Gates along with their truth tables.

LIST OF APPARATUS - Digital Trainer Kit, wires, Integrated Circuits

We get two kinds of Logic Gates -

- Basic Gates
  - → 4071 2-input OR
  - → 4081 2-input AND
  - → 4069 1-input NOT
  - → 4070 2-input XOR
- Universal Gates
  - → 4001 2-input NOR
  - → 4011 2-input NAND

THEORY - The manipulation of binary information is done by logic circuits called "gates". Gates are blocks of hardware that produce signals of 1 and 0 when the input logic requirements are met. This input output relationship of the binary variables for each gate can be represented in a tabular format which is referred as a "truth table". These gates / logic gates can be classified into two types, based on their usage.

1) Basic Gates 2) Universal Gates

Now, these gates are also sub-divided. There are 4 kinds of Basic gates and 2 kinds of universal Gates. The basic gates are OR, AND, XOR and NOT gate. While the universal gates are NOR and NAND gate.

figure 1 contains the schematic representation of the classification of logic gates.

LOGIC GATES

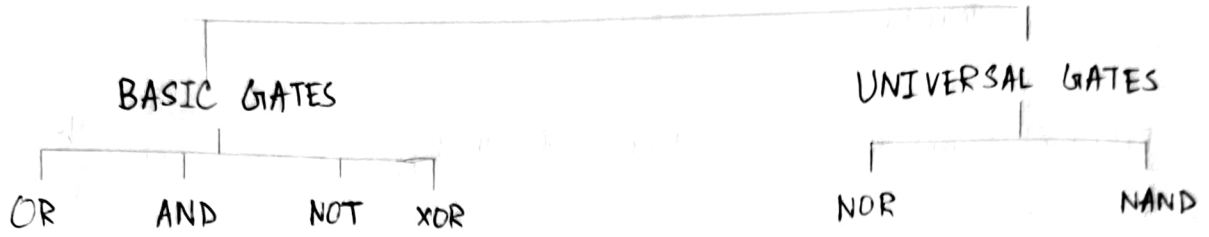


Fig 1: Schematic representation of the classification of Logic Gates

- **OR** → The OR gate produces the output 1 if the input A or input B or both inputs are 1; otherwise the output is 0.
- **AND** → The AND gate produces the output 1 if input A and input B, both equals to 1; otherwise the output is 0.
- **NOT** → The NOT gate, also known as inverter, produced the output 1 if the single input A equals to 0
- **XOR** → The output of the XOR gate is 1 if any input is 1 but excludes the combination when both inputs are 1.
- **NOR** → The NOR gate, or NOT-OR gate, produces the output 0 if input A or input B or both inputs are 1; otherwise the output is 1.
- **NAND** → The NAND gate, or NOT-AND gate, produces the output 0 if input A and input B, both equal to 1; otherwise the output is 1.

Figure 2 contains the graphical representation of each of the logic gates.

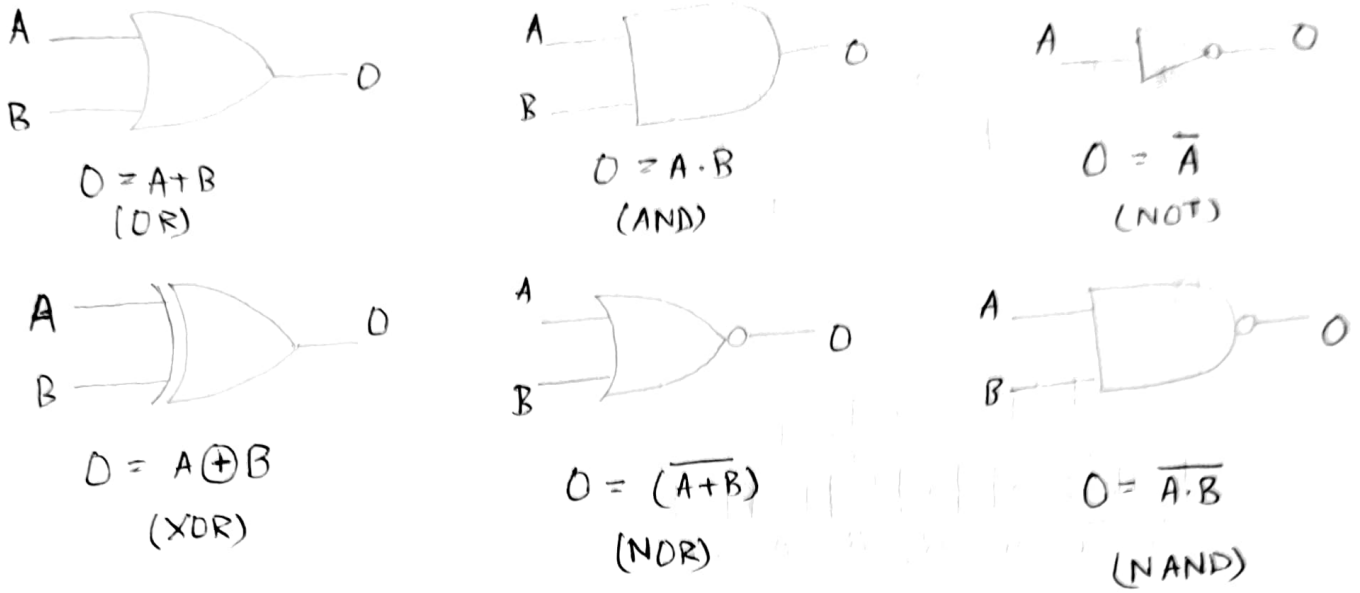
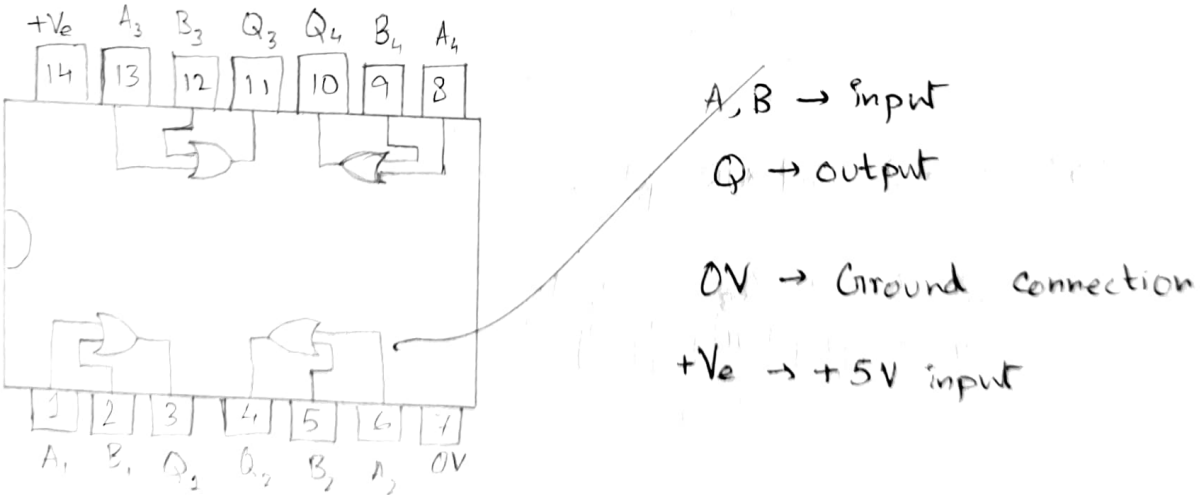


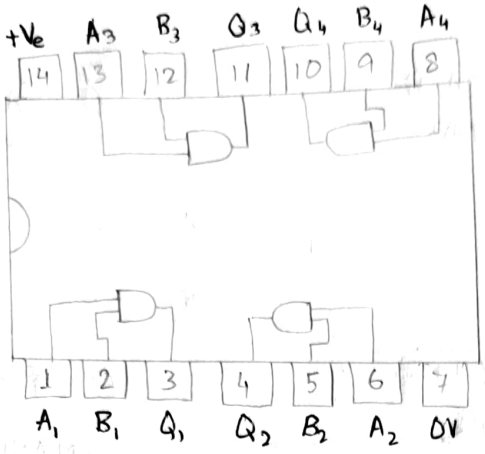
Fig 2: Graphical representation of the logic Gates

PIN CONFIGURATONS - The following diagrams are the pin Configurations of the given Integrated circuits :-

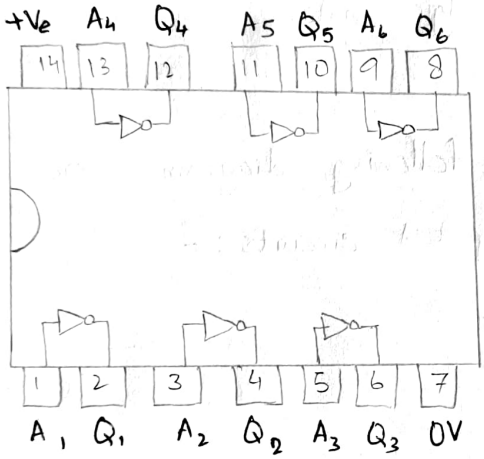
IC4071 → 2-input OR Gate



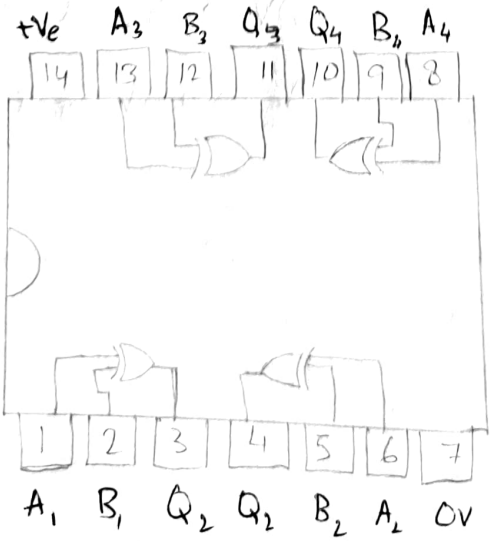
IC 4081 → 2-input AND Gate



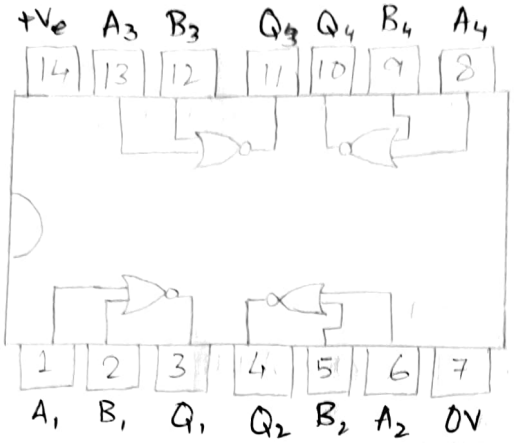
IC 4069 → 1 input NOT Gate



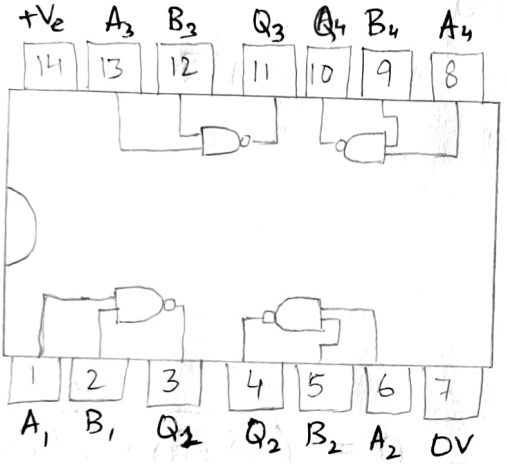
IC4070 → 2 input-XOR gate



IC 4001 → 2-input NOR Gate



IC 4011 → 2-input NAND Gate



CIRCUIT DIAGRAM: Circuit diagrams for the verification of each logic gates are as follows

O → Output
A, B → Inputs

OR Gate →

O = A + B

AND Gate →

O = A · B

NOT Gate → 

$O = \bar{A}$

XOR Gate → 

$O = A \cdot B' + A' \cdot B = A \oplus B$

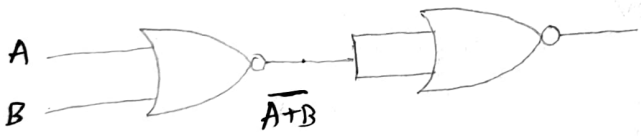
NOR Gate → 

$O = \overline{A+B}$

NAND Gate → 

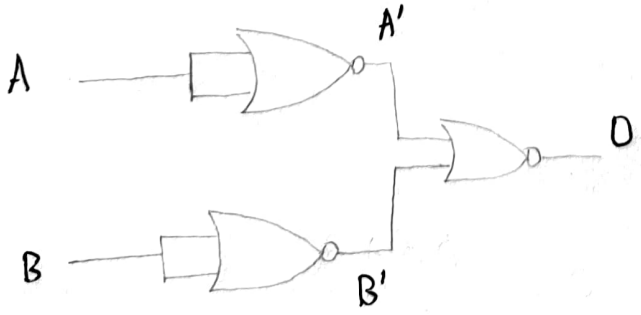
$O = \overline{A \cdot B}$

OR Using NOR Gate →



$$O = \overline{(\overline{(A+B)} + \overline{(A+B)})}$$
$$= (\overline{(\overline{A+B})})$$
$$= A+B$$

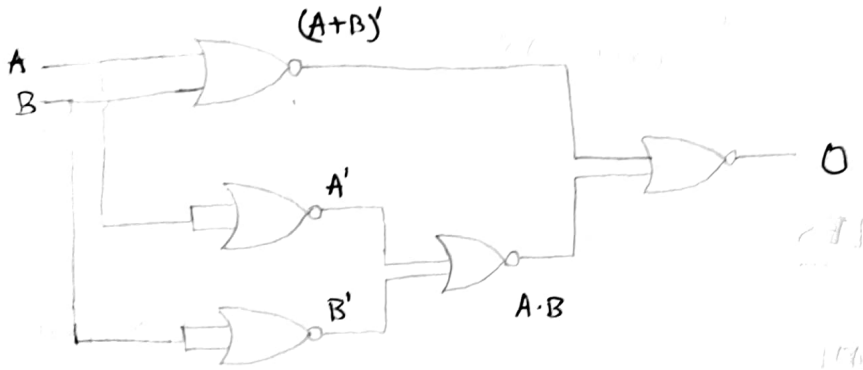
AND Using NOR Gate →



$O = (A' + B') = A \cdot B$

NOT Using NOR Gate

XOR Using NOR Gate

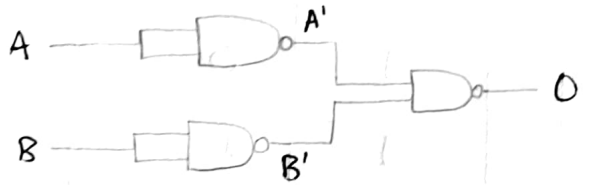


$O = ((A+B)' + A \cdot B)'$

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

$= AB' + A'B$

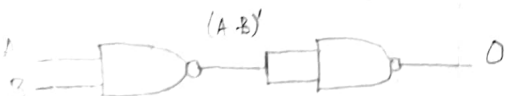
OR Using NAND Gate



$O = (A' \cdot B')'$

$= A + B$

AND Using NAND Gate



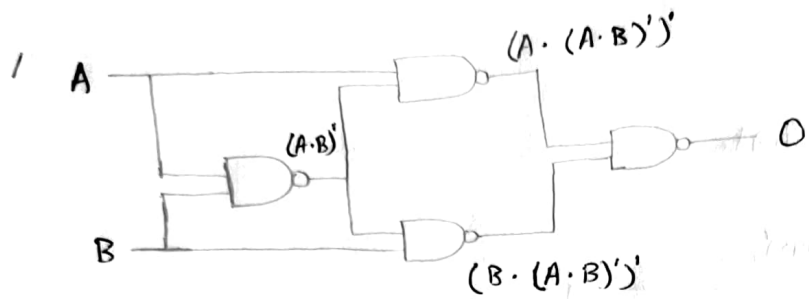
$O = ((A \cdot B)' \cdot (A \cdot B)')'$

$= ((A \cdot B)')' = A \cdot B$

NOT Using NAND Gate

$O = (A \cdot A)' = A'$

XOR Using NAND Gate



$O = ((A\cdot(A\cdot B)')'\cdot(B\cdot(A\cdot B)')')'$

$= AB' + A'B$

TRUTH TABLES

OR GATE		
A	B	O = A+B
0	0	0
0	1	1
1	0	1
1	1	1

AND Gate		
A	B	O = A·B
0	0	0
0	1	0
1	0	0
1	1	1

NOT Gate	
A	O = A'
0	1
1	0

XOR Gate		
A	B	O = A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

NOR Gate

A	B	Q = A + B	O = (A + B)'
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

NAND Gate

A	B	Q = A · B	Q = (A · B)'
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

OR Using NOR Gate →

A	B	A + B	(A+B)'	O = ((A+B)' + (A+B)')'
0	0	0	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	0	1

AND Using NOR Gate-

A	B	A+A	B+B	A'	B'	(A'+B')'	(A'+B')'	O = AB
0	0	0	0	1	1	1	0	0
0	1	0	1	1	0	1	0	0
1	0	1	0	0	1	1	0	0
1	1	1	1	0	0	0	1	1

NOT Using NOR Gate:

A	A+A	O = (A+A)' = A'
0	0	1
1	1	0

XOR Using NOR Gate:

A	B	$(A+B)'$	A'	B'	$(A'+B')' = A \cdot B$	$O = ((A+B)' + A \cdot B)'$
0	0	1	1	1	0	0
0	1	0	1	0	0	1
1	0	0	0	1	0	1
1	1	0	0	0	1	0

~~NOT Using NOR Gate~~:

OR Using NAND Gate:

A	B	$(A \cdot A)' = A'$	$(B \cdot B)' = B'$	$O = (A' \cdot B')' = A+B$
0	0	1	1	0
0	1	1	0	1
1	0	0	1	1
1	1	0	0	1

AND Using NAND Gate:

A	B	$(A \cdot B)'$	$O = ((A \cdot B)' \cdot (A \cdot B)')' = A \cdot B$
0	0	1	0
0	1	1	0
1	0	1	0
1	1	0	1

NOT using NAND Gate:

A	A·A	0 = (A·A)'
0	0	1
1	1	0

XOR using NAND Gate:

A	B	(A·B)'	(A·(A·B)')'	(B·(A·B)')'	0 = ((A·(A·B)')'·(B·(A·B)')')' = $A\overline{B} + \overline{A}B = A \oplus B$
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	0

CONCLUSION: In the lab, I explored the function of the basic logic gates. I learned how to implement them on a breadboard with integrated circuits. I tested the output voltage of the circuits and ensured the result corresponded with the truth table of the logic gate tested.