# Study of Boolean Logic Operations using Digital ICs

Gayatri P
*2nd year, Integrated M.Sc. Physics*
*Roll No.: 2211185*
(Dated: February 14, 2024)

In this experiment, we study various logic gates and verify boolean logic operations. We also get familiar with ICs and construction of circuit using ICs. We also demonstrate De Morgan's Law using NAND and NOR gates, discuss its importance and verify the output.

## I. OBJECTIVE

To study and verify various Boolean logic operations and the De Morgan's laws using digital ICs.

## II. THEORY

### Introduction

In digital electronics, we deal with circuits in which there are only two states possible at any point, The `HIGH` and `LOW` voltage states represent the `TRUE` and `FALSE` states of Boolean logic.

When input signal is naturally discrete in nature, e.g., pulses from a particle detector, or *bits* of data from a computer, the use of digital electronics is convenient. Furthermore, it is often desirable to convert analog data to digital form, and vice versa in order to perform computations on the data or to store large quantities of data as numbers.

Another advantage of digital techniques is the transmission of analog signals without degradation by noise. An analog signal, picks up noise while being transmitted by cable or wireless that cannot be removed. If, instead, the signal is converted to a series of numbers representing its amplitude at successive instants of time, and these numbers are transmitted as digital signals, the analog signal reconstruction at the receiving end will minimize error. This technique is known as pulse-code modulation.

### Logic Gates

All digital electronic circuits and microprocessor based systems contain hardware elements called Digital Logic Gates that perform the logical operations of AND, OR and NOT on binary numbers.

Standard commercially available Digital Logic Gates are available in two basic forms,

- Transistor-Transistor Logic (TTL): 7400 series
- Complementary Metal-Oxide-Silicon (CMOS) 4000 series

These logic gates are embedded on Integrated Circuits (IC) or *chips*. TTL IC's use NPN type Bipolar Junction Transistors while CMOS IC's use Field Effect Transistors or FET's for both their circuitry.

In digital circuitry, the voltage levels corresponding to `HIGH` and `LOW` are allowed to fall in some range, according to the particular logic family. For example, in a standard +5 volt supply, any TTL voltage input between 2V and 5V is considered to be `HIGH` while any voltage input below 0.8V is considered as `LOW`. TTL outputs are recognized as `LOW` between 0V and 0.4V and `HIGH` between 2.7 V and 5 V.

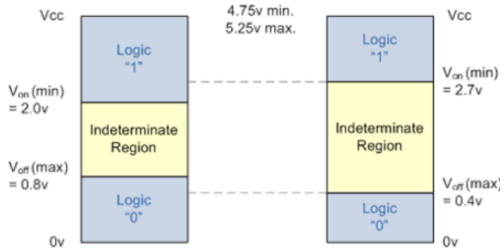
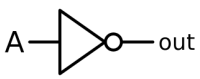



FIG. 1: TTL Input & Output Voltage Levels

There are several types of simple logic gates.

- **NOT Gate**: Also called an inverter, it takes one bit as input and produces output as its opposite. $Q = \overline{A}$




| A | Out |
|---|---|
| 1 | 0 |
| 0 | 1 |

FIG. 2: The circuit symbol and logic table for NOT gate

- **OR Gate**: performs a logical OR operation on two inputs, A and B: $Q = A + B$.


| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

FIG. 3: The circuit symbol and logic table for OR gate

- **AND Gate**: performs a logical AND operation on two inputs, A and B: $Q = A \cdot B$.


| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

FIG. 4: The circuit symbol and logic table for AND gate

There are also two **Universal Gates**. These are combinations of an AND or an OR gate with a NOT gate.

- **NAND Gate**: Inverts AND operation: $Q = \overline{A \cdot B}$.


| A | B | Q |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

FIG. 5: The circuit symbol and logic table for NAND gate

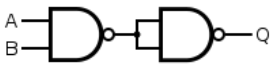
- **NOR Gate**: Inverts OR operation: $Q = \overline{A + B}$.


| A | B | Q |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

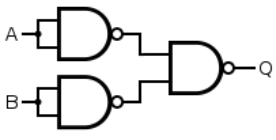
FIG. 6: The circuit symbol and logic table for NOR gate

Thse gates can implement any Boolean function without need to use any other gate type. In practice, this is advantageous since NAND and NOR gates are economical and easier to fabricate and are the basic gates used in all IC digital logic families.


(a) NOT operation using NAND



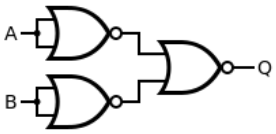

(b) AND operation using NAND



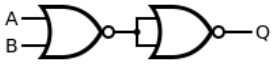

(c) OR operation using NAND

FIG. 7: Other gates constructed using NAND


(a) NOT operation using NOR




(b) AND operation using NOR




(c) OR operation using NOR

FIG. 8: Other gates constructed using NOR

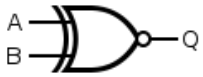
Additionally there are two more gates used.

- **XOR Gate**: Exclusive OR, it outputs true if exclusively one of the inputs is true. Mathematically, $Q = A \cdot \overline{B} + B \cdot \overline{A} = A \oplus B$.


| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

FIG. 9: The circuit symbol and logic table for XOR gate

- **XNOR Gate**: Inverts XOR gate. Mathematically, $Q = A \cdot B + \overline{A} \cdot \overline{B} = A \odot B$.




| A | B | Q |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

FIG. 10: The circuit symbol and logic table for XNOR gate

### De Morgan's Laws

De Morgan's laws, state that NAND gate is equivalent to an OR gate with negated inputs, and a NOR gate is equivalent to an AND gate with negated inputs. This is most commonly used to implement logic gates as combinations of only NAND gates, or as combinations of only NOR gates as shown in Fig. 7 & 8.

There are two basic theorems which can be proved using set theory. (Note: Here we use UNION and OR operators equivalently, as well as the INTERSECTION and AND operators).

***Theorem 1:*** $\overline{A+B} = \overline{A} \cdot \overline{B}$
***Proof:*** Using set theory,

$$\begin{aligned} &\text{Let } x \in \overline{A \cup B} \\ &\implies x \notin (A \cup B) \\ &\implies x \notin A \text{ and } x \notin B \\ &\implies x \in \overline{A} \text{ and } x \in \overline{B} \\ &\implies x \in (\overline{A} \cap \overline{B}) \ \forall x \in \overline{A \cup B} \\ &\text{or, } \overline{A} + \overline{B} = \overline{A \cdot B} \end{aligned}$$

***Theorem 2:*** $\overline{A \cdot B} = \overline{A} + \overline{B}$
***Proof:*** Using set theory,

$$\begin{aligned} &\text{Let } y \in \overline{A \cap B} \\ &\implies y \notin (A \cap B) \\ &\implies y \notin A \text{ or } y \notin B \\ &\implies y \in \overline{A} \text{ or } y \in \overline{B} \\ &\implies y \in (\overline{A} \cup \overline{B}) \ \forall y \in \overline{A \cap B} \\ &\text{or, } \overline{A} \cdot \overline{B} = \overline{A + B} \end{aligned}$$

## III. EXPERIMENTAL SETUP

### Apparatus

1. Digital ICs
2. Resistors
3. DC Power Supply (5V)
4. Breadboard
5. Connecting Wires
6. LEDs

### Circuit Diagrams

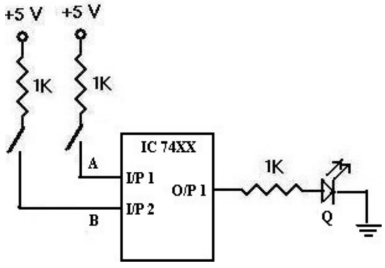



FIG. 11: General circuit diagram used for all the ICs

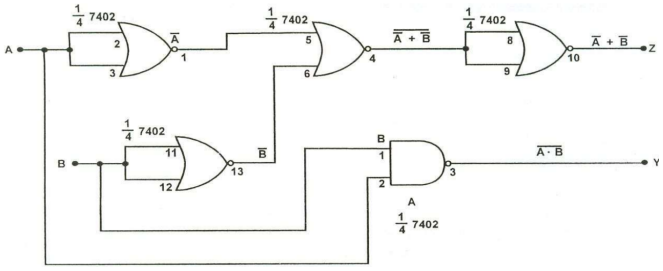



FIG. 12: Circuit diagram used for proving De Morgan's Law: $\overline{A \cdot B} = \overline{A} + \overline{B}$

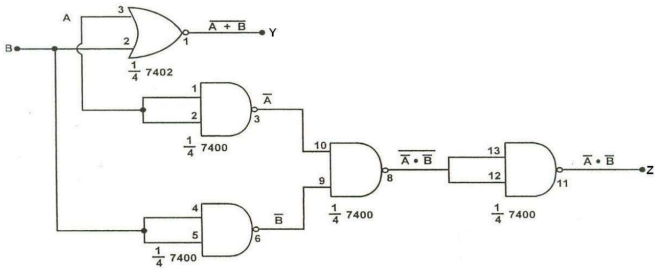



FIG. 13: Circuit diagram used for proving De Morgan's Law: $\overline{A+B} = \overline{A} \cdot \overline{B}$

### IC Pinout Diagrams

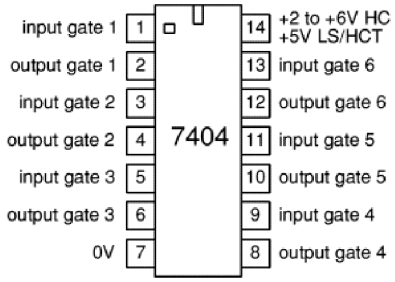



FIG. 14: Pinout Diagram for NOT (IC 7404) gate. It contains 6 gates per IC.

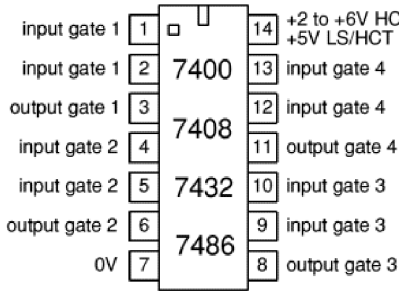



FIG. 15: Pinout Diagram for AND (IC 7408), OR (IC 7432), XOR (IC 7486) and NAND (IC 7400) gates. Note how it contains only 4 gates per IC

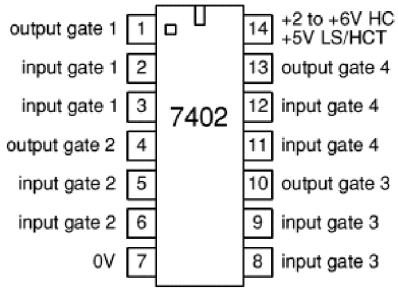



FIG. 16: Pinout Diagram for NOR (IC 7402) gate

## IV. OBSERVATIONS

### Verification of Boolean Logic Operations

As shown in Fig 11, we connected the IC to the 5V power suppy from the DC power power supply source. The `HIGH` or '1' inputs were provided from the 5V line while the `LOW` or '0' inputs were provided from the ground line. The outputs were observed by connecting an LED across the output pin and ground (output '1' means the LED glows and vice-versa).

1. **NOT Gate** (IC 7404LS)
   Operation: $Q = \overline{A}$

| **Gate 1: Pin 1 & 2** | | **Gate 2: Pin 12 & 13** | |
|---|---|---|---|
| I/P (A) | O/P (Q) | I/P (A) | O/P (Q) |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |

TABLE I: I/P and O/P values observed for NOT gate

2. **OR Gate** (IC 7432LS)
   Operation: $Q = A + B$

| **Gate 1: Pins 1, 2 & 3** | | | **Gate 2: Pins 11, 12 & 13** | | |
|---|---|---|---|---|---|
| I/P (A) | I/P (B) | O/P (Q) | I/P (A) | I/P (B) | O/P (Q) |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 |

TABLE II: I/P and O/P values observed for OR gate

3. **AND Gate** (IC 7408LS)
   Operation: $Q = A \cdot B$

| **Gate 1: Pins 1, 2 & 3** | | | **Gate 2: Pins 11, 12 & 13** | | |
|---|---|---|---|---|---|
| I/P (A) | I/P (B) | O/P (Q) | I/P (A) | I/P (B) | O/P (Q) |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 |

TABLE III: I/P and O/P values observed for AND gate

4. **NAND Gate** (IC 7400LS)
   Operation: $Q = \overline{A \cdot B}$

| **Gate 1: Pins 1, 2 & 3** | | | **Gate 2: Pins 11, 12 & 13** | | |
|---|---|---|---|---|---|
| I/P (A) | I/P (B) | O/P (Q) | I/P (A) | I/P (B) | O/P (Q) |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 |

TABLE IV: I/P and O/P values observed for NAND gate

5. **NOR Gate** (IC 7402LS)
   Operation: $Q = \overline{A + B}$

| **Gate 1: Pins 1, 2 & 3** | | | **Gate 2: Pins 11, 12 & 13** | | |
|---|---|---|---|---|---|
| I/P (A) | I/P (B) | O/P (Q) | I/P (A) | I/P (B) | O/P (Q) |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 |

TABLE V: I/P and O/P values observed for NOR gate

6. **XOR Gate** (IC 7486LS)
   Operation: $Q = A \cdot \overline{B} + B \cdot \overline{A}$

| **Gate 1: Pins 1, 2 & 3** | | | **Gate 2: Pins 11, 12 & 13** | | |
|---|---|---|---|---|---|
| I/P (A) | I/P (B) | O/P (Q) | I/P (A) | I/P (B) | O/P (Q) |
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 |

TABLE VI: I/P and O/P values observed for XOR gate

Here, all observations aligh with and thus verify our predictions.

### Verification of De Morgan's laws

1. $\overline{A + B} = \overline{A} \cdot \overline{B}$

| I/P (A) | I/P (B) | $\overline{A + B}$ | Y (Obs) | $\overline{A}$ | $\overline{B}$ | $\overline{A} \cdot \overline{B}$ | Z (Obs) |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |

TABLE VII: Observation table for proving De Morgan's Law based on the circuit diagram given in Fig. 13

2. $\overline{A \cdot B} = \overline{A} \cdot \overline{B}$

| I/P (A) | I/P (B) | $\overline{A \cdot B}$ | Y (Obs) | $\overline{A}$ | $\overline{B}$ | $\overline{A} + \overline{B}$ | Z (Obs) |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |

TABLE VIII: Observation table for proving De Morgan's Law based on the circuit diagram given in Fig. 12

Hence, the observed and predicted values match, thus verifying De Morgan's Laws.

## V. DISCUSSION & CONCLUSION

In this experiment, we have studied various logic gates and have verified boolean logic operations using ICs. We have also demonstrated De Morgan's Law using NAND and NOR gates and verified the output. We can here observe the importance of De Morgan's Law as it can be used to implement any logic gate as combinations of only NAND gates or NOR gates. Hence we can manufacture ICs easily in a cost-efficient way.

Hence we are now familiar with the working and functioning of Digital logic gates and Integrated Chips.

## VI. PRECAUTIONS

1. Verify the connections before switching on the circuit.
2. Make sure take readings only when all the the connections are closed.
3. Make sure the IC is properly grounded.

## VII. APPLICATIONS

Logic gates are extensively used in the modern world. They can be used in digital measuring instruments, calculators, automation control machines, digital memory devices for storage etc. A combination of a few to thousands of logic gates are used in manufacturing ICs used in microprocessors, CPUs and GPUs, which make up our computers, mobile phones and laptops.

[1] SPS, *Study of Boolean Logic Operations using Digital ICs*, NISER (2023).
[2] P. Horowitz and W. Hill, *The art of electronics* (Cambridge University Press, 2015).