LABORATORY REPORT - DIGITAL ELECTRONIC LABORATORY N° 1

Students: Toctaguano Doménica, Torres Salomé, Troya Alejandro
Teacher: Msc. Daniel Jaramillo
Laboratory Teacher: Msc. Alejandra Pinto Erazo

November 5, 2024

1 Introduction

In this practice, have been explored the function and the applications of logic gates, with the use of electronic circuits. The gates that have been explored were the three more fundamentals: AND, OR and NOT and as of this, can build complex digital systems, as that of computers. The principal purpose of this practice have been to understand the function of each mentioned gate and how this can combine for specific tasks and operations as sum, multiplication and inversion of result, for this, have been used integrated circuits with gates, boolean expressions and diagrams of circuits, have been built basic circuits based on simulations with TinkerCAD and Proteus. With the proposals activities, developed a big comprehension of logic operations and its influence in complex logic operations, the which are base of the modern technology.

2 Objectives

2.1 General Objective

- Develop skills for analysis and implementation of basic logic circuits from study of logic gates, with the purpose of have comprehension of fundamentals concepts of digital systems.

2.2 Specific Objectives

- Analyze the function of each logic gate: AND, OR and NOT from the creation of truth tables for comprehension the logic operations.
- Build combinatorial circuits using integrated circuits for implement specific boolean functions.

3 Theoretical Foundation

3.1 Configurations Pull-Up and Pull-Down

3.1.1 What are pull-up resistors?

Pull -up resistors are resistors used in logic circuits to ensure a well-defined logic level on a pin under all conditions. As a reminder, digital logic circuits have three logic states: high, low, and floating (or high impedance). The high impedance state occurs when the pin is not pulled to a high or low logic level, but is instead left "floating." A good example of this is an unconnected input pin on a microcontroller. It is not in a high or low logic state, and the microcontroller might unpredictably interpret the input value as a high or low logic level. Pull -up resistors are used to solve the microcontroller's dilemma by pulling the value to a high logic state, as seen in the following figure. EEPOWER (n.d.) Pull -up resistors are resistors used to ensure that a wire is pulled to a high logic level in the absence of an input signal.

3.1.2 What are pull-down resistors?

Pull-down resistors work in the same way as pull -up resistors , except that they pull the pin to a logic low value. They are connected between ground and the corresponding pin on a device. An example of a pull-down resistor in a digital circuit can be seen in the figure below . EEPOWER (n.d.)

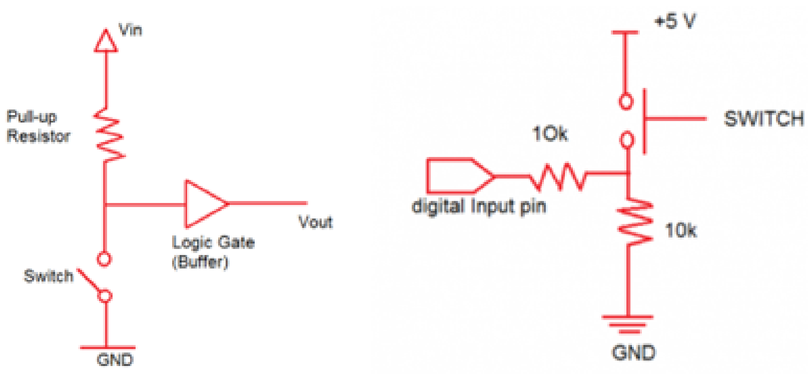


Figure 1: Configurations Pull-Up and Pull-Down. Source: EEPOWER (n.d)

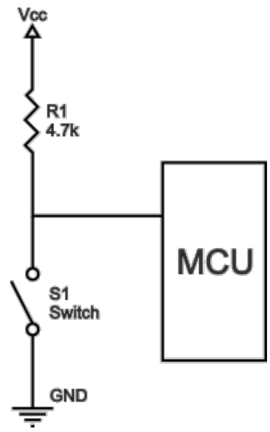


Figure 2: Pull-Up. Source: EEPOWER (n.d)

3.1.3 Typical Applications of Pull -Up and Pull-Down Resistors

Pull -up and pull-down resistors are often used when interfacing a switch or some other input to a microcontroller or other digital gates. Pull -up resistors are generally more frequently used than pull-down resistors , although some microcontroller families have both pull -up and pull-down resistors available . They are often used to provide a controlled current flow to a resistive sensor prior to analog-to-digital conversion of the sensor's output voltage signal. When not connected to a bus, the pin floats in a high-impedance state. EEPOWER (n.d.)

3.2 Gate AND

The AND gate is also known as "all or nothing" gate. In Boolean Algebra it is represented by a multiplication, therefore to have the output in active state it is necessary that its inputs have a binary state 1, when having an inactive input "0" its output will be 0. MecatrónicaLATAM (2021a)

- Q = $A \cdot B$

3.2.1 Representative circuit of the AND gate

It can be represented by a circuit that has its switches in series, having all the switches active allows the circuit to be closed and therefore the flow of current that allows the focus to be activated (representation).

3.2.2 AND logic gate integrated circuits

74 series integrated circuits

- 7408 - 2-input AND gate.
- 7409 - 2-input, open collector output AND gate.
- 7411 - 3 3-input AND gates.

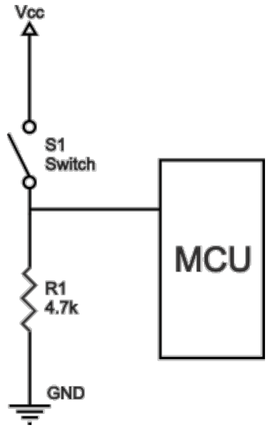


Figure 3: Pull-Down. Source: EEPOWER (n.d)

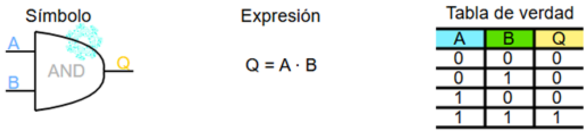


Figure 4: GATE AND. Source: MecatrónicaLATAM (2021)

- 7415 - 3 AND gates with 3 inputs and open collector output.
- 7421 - 2 4-input AND gates.

4000 Series Integrated Circuits

- 4019.
- 4073.
- 4085.
- 4086

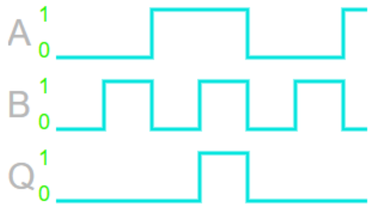
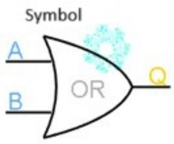


Figure 5: Representation AND. Source: MecatrónicaLATAM (2021)

3.3 Gate OR

The logic gate OR is known as the gate of "everything or anything", because it's expression of boolean algebra, its symbol is an add. This logic component it's located in active state while of one of its ports of input have an active binary state "1", and for getting an inactive state "0". In the output, it requires what all its points of outputs were heal inactive state "0". MecatrónicaLATAM (2021b)



Expression

$Q = A + B$

Truth table

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

Figure 6: Symbol, expression and truth table of Gate OR. Source: MecatrónicaLATAM (2021)

Types of gates OR:
Usually, can find with integrated circuits which contain gates of two, three and even of four ports of input. Even exists also diverse structures for create a gate with more points for inputs, just how indicates Ingeniería-Mecafenix (2023):

- Gate OR 78LS32, which contain two inputs.
- Gate OR 74HC4075, with three inputs (Technology CMOS).
- Gate OR CD4072, which have four inpunts (Technology CMOS).

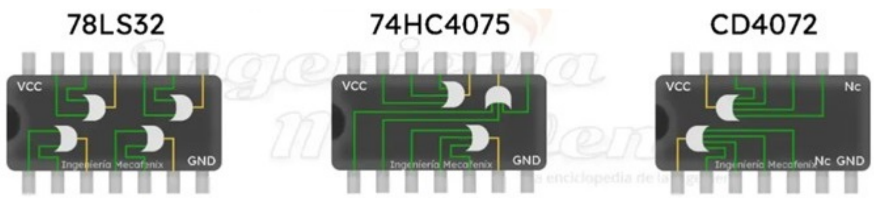


Figure 7: Types of Gate OR. Source: Ingeniería Mecafenix (2023)

3.4 Gate NOT

The gate NOT or INV (Inverse), contain a port of input and a port of output, its objective it's to produce an output inverse or opposite to point of input. Even if the terminal of input locates in activate state "1" will count to the output as an inactive state "0" and for the opposite case, if the port of input is in a inactive state "0" the output terminal will be activate state "1". MecatrónicaLATAM (2021b) The gates NOT can be unified like a waterfall. By this way, it obtains, after two gates, a port of output equal to the original input port. Electrónica-Unicrom (n.d.)
Its symbols in the diagrams has the shape of triangle, pointing to the right with a circle in one of its sides.
Types of gates NOT:
For multiple motives, this gate only form a port it has now existent variations.

- Logic gate NOT 74Ls04.

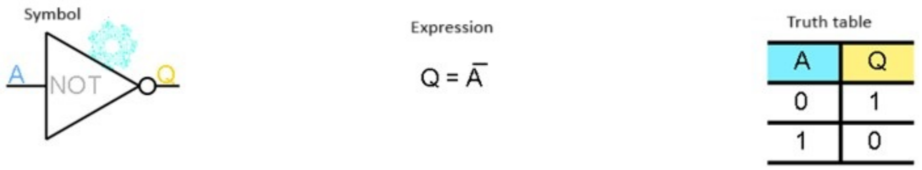


Figure 8: Symbol, expression and truth table of Gate NOT. Source: MecatrónicaLATAM (2021)

- NOT 74LS05, it can be an open collector.
- NOT 74LS14, schmitt trigger.

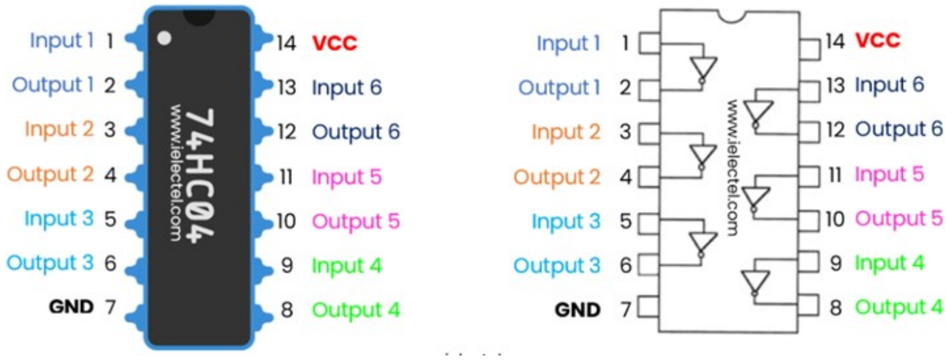


Figure 9: Principal Gate NOT. Source: Electronica Unicrom (n.d)

4 Materials and Equipments

- Personal computer with Proteus and TinkerCAD: Proteus and Tinkercad are two Software of simulations for circuits. TinkerCAD has more tools for create basic and complex circuits, with interactive interface and web page.

TinkerCAD has several tools for create basic and complex circuits, with interactive interface and web page. Proteus has several tools for create basic and complex circuits, but it has more tools that TinkerCAD and its interface is less interactive, is more formal and should install in the computer.

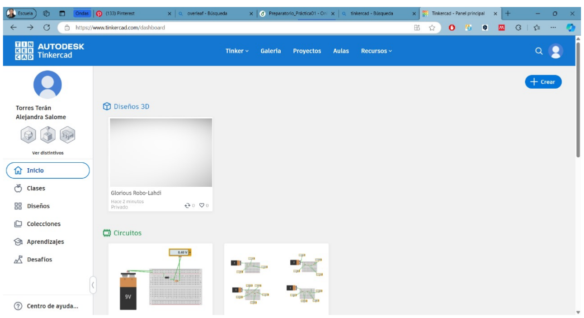


Figure 10: TinkerCAD. Source: Torres (2024)

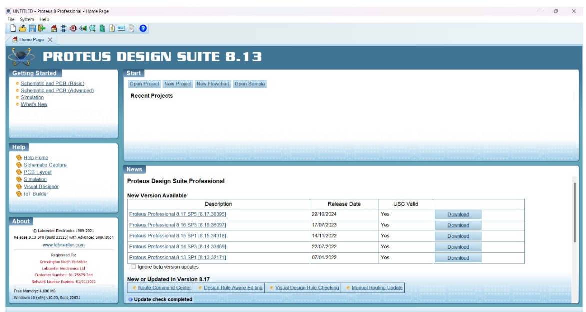


Figure 11: Proteus. Source: Torres (2024)

- Protoboard

It's a tablet used for experimentation, it's made of plastic case, rails of copper and a bottom of foam. It's used for prototype circuits and simple circuits.

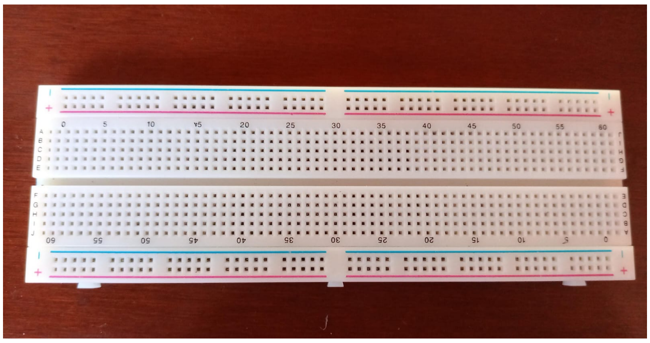


Figure 12: Protoboard. Source: Torres (2024)

- Integrated Circuits: 7432, 7404 and 7408 Are electronic components, which posses a specific number of gates, depending of them, will be a different output.

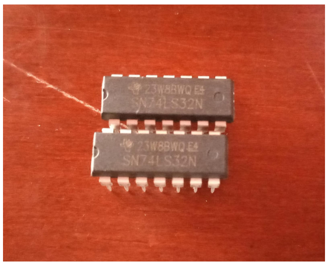


Figure 13: Integrated Circuit 7432. Source: Torres (2024)

Figure 14: Integrated Circuit 7404. Source: Torres (2024)

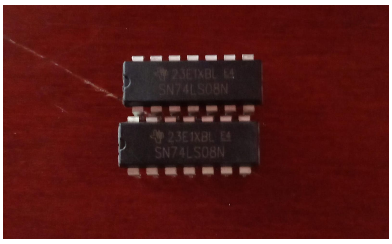


Figure 15: Integrated Circuit 7408. Source: Torres (2024)

- Dipswitch of two or four positions

It's a electronic component, which have two, four of eight switches. Can activate or block inputs.

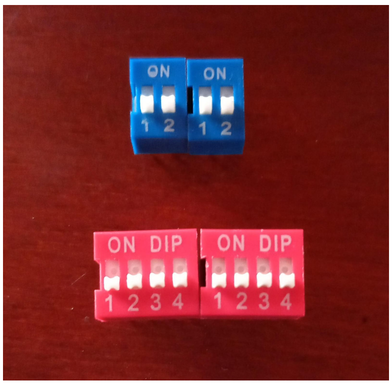


Figure 16: Dipswitches of two and four positions. Source: Torres (2024)

- Resistances of 10 K ohms, 1 k ohms, 220 ohms and 330 ohms

The resistances are electronic components, which control the flow of energy.

Figure 17: Resistances of 10 K ohms. Source: Torres (2024)

Figure 18: Resistances of 1 k ohm. Source: Torres (2024)

Figure 19: Resistances of 330 ohms. Source: Torres (2024)

Figure 20: Resistances of 220 ohms. Source: Torres (2024)

- Leds

The led diode is an electronic component, which emits light.

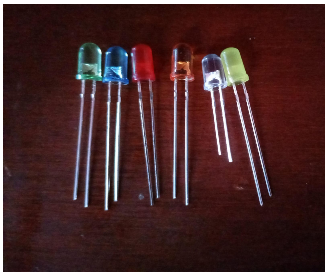


Figure 21: Leds of different colors. Source: Torres (2024)

- Charger of 5 Volts

This component is very important, because the integrated circuits works in less 7 Volts. **Costs Table:**

Figure 22: Charger of 5 Volts. Source: Torres (2024)

Material	**Quantity**	**Unit Price**	**Total Price**
Protoboard	4	$ 5	$20
Integrated Circuit 7432	6	$0,70	$4,20
Integrated Circuit 7404	6	$0,70	$4,20
Integrated Circuit 7408	6	$0,80	$4,80
Dipswitch	6	$0,70	$4,20
Resistances	40	$0,10	$3,00
Leds	18	$0,10	$1,80
Charger of 5 Volts	1	$4,50	$4,50
Jumpers	26	$0,80	$1,60
Total			$49,80

Table 1: Results of materials. Source: Torres (2024)

5 Development

In this practice, have been realized six exercises about several logic operations with use of gates in different configurations:

5.1 Exercise 1

Design a circuit which realizes a logic function $A \cdot B + \overline{C}$. Use the gate AND (7408) for $A \cdot B$, a gate NOT (7404) to inverse C, and a gate OR (7432) to summate both results.

- Connect A and B to a gate AND (7408).
- Connect C to a gate NOT to obtain C (7404).
- Connect the output of AND and NOT to a gate OR (7432).

For this exercise, were made the next simulations:

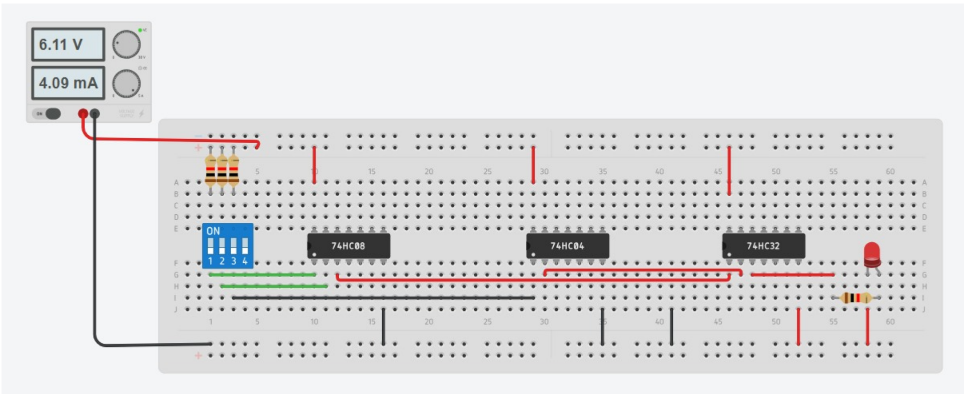


Figure 23: Exercise 1 in TinkerCAD. Source: Torres (2024)

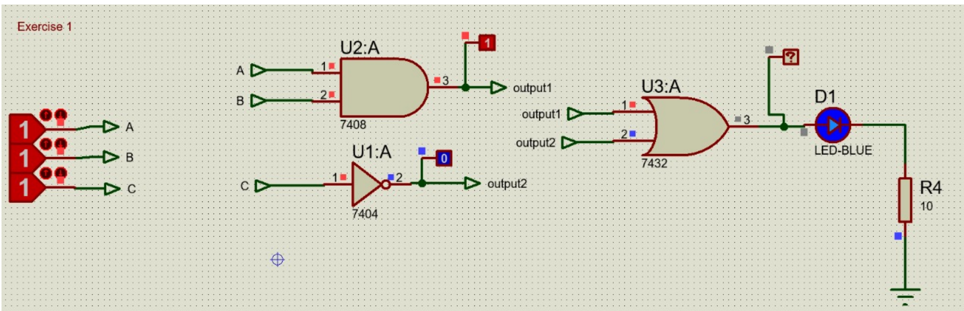


Figure 24: Exercise 1 in Proteus. Source: Torres (2024)

In this simulation, can view the asked function. The simulation has components as: resistances, dipswitch, integrated circuits, led diode and a voltage source.
Truth table:

A	B	C	$A \cdot B$	$\overline{C}$	AB + $\overline{C}$
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	0	1	1
1	0	1	0	0	0
1	1	0	1	1	1
1	1	1	1	0	1

Table 2: Results of exercise 1 - Theoric. Source: Torres (2024)

In base of this table, verified each result:
Assembled circuit:

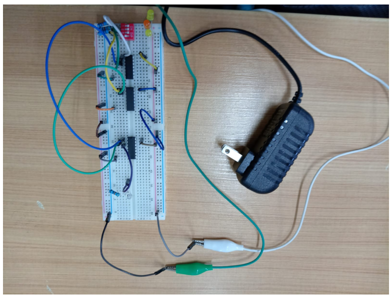


Figure 25: Assembled. Source: Torres (2024)

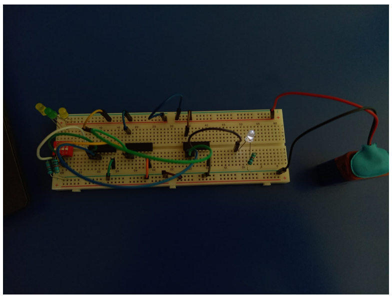


Figure 26: Result 1. Source: Torres (2024)

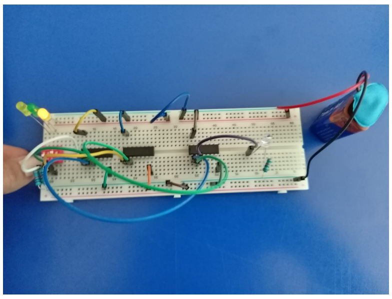


Figure 27: Result 2. Source: Torres (2024)

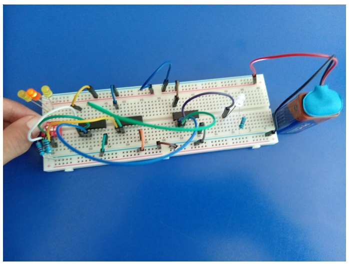


Figure 28: Result 3. Source: Torres (2024)

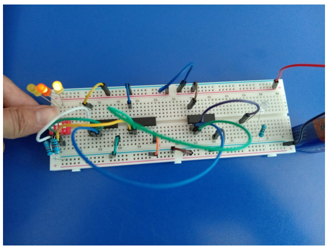


Figure 29: Result 4. Source: Torres (2024)

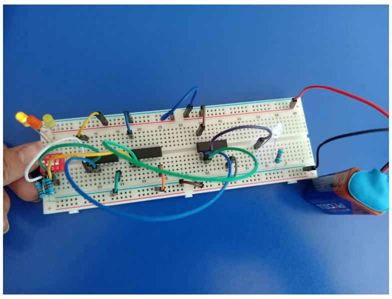


Figure 30: Result 5. Source: Torres (2024)

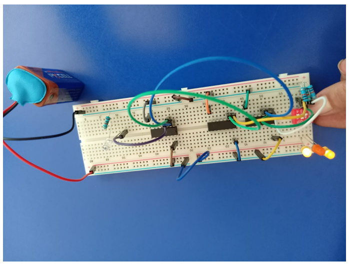


Figure 31: Result 6. Source: Torres (2024)

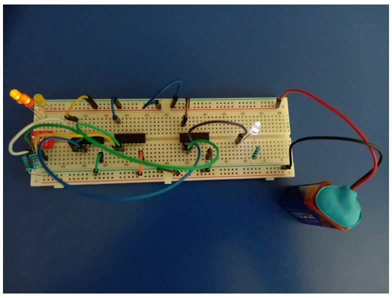


Figure 32: Result 7. Source: Torres (2024)

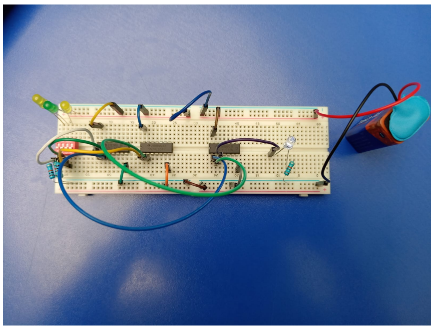


Figure 33: Result 8. Source: Torres (2024)

5.2 Excercise 2

Create a circuit with the logical function A + B. Use the 7432 to perform A + B and the 7404 to invert the result.

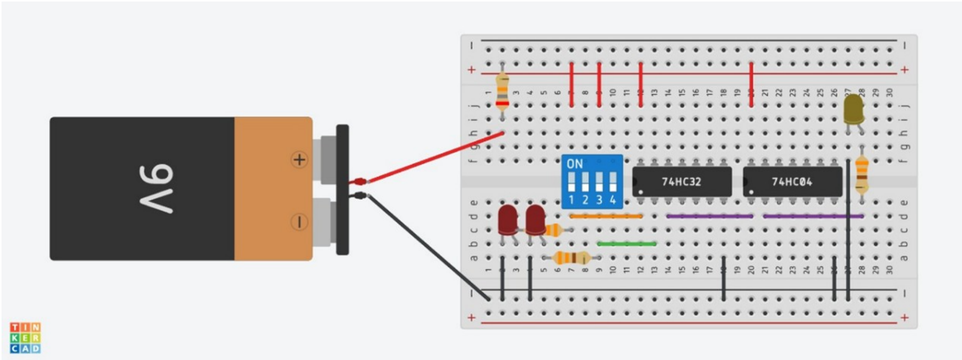


Figure 34: Exercise 2 in Tinkercad. Source: Toctaguano (2024)

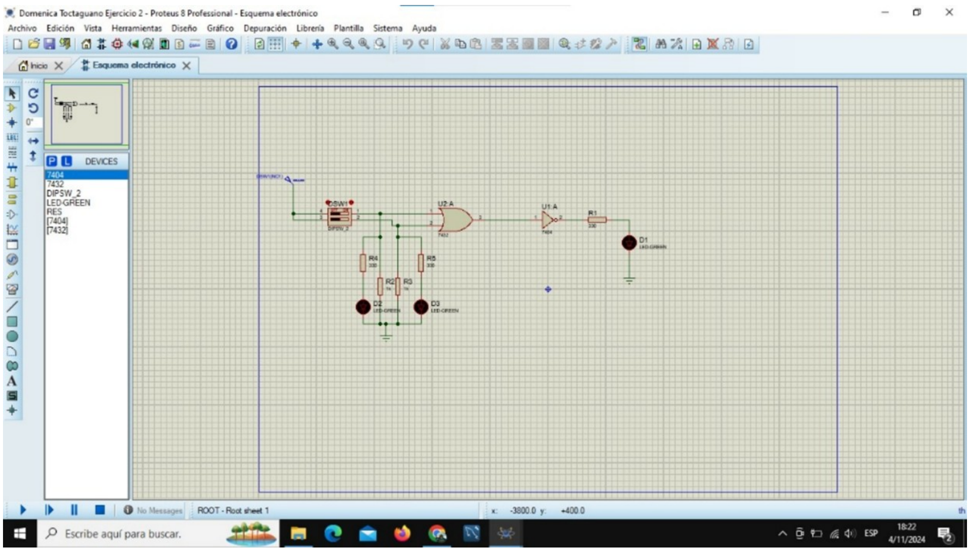


Figure 35: Exercise 2 in Proteus. Source: Toctaguano (2024)

- **Resistors:** These components are used to limit the flow of current and control the voltage levels in the circuit. They play an essential role in protecting other components and ensuring proper functioning.
- **Dipswitch:** A manual switch that allows you to set binary input values for the circuit. By toggling the switches, you can change the inputs and observe how the circuit responds.
- **Integrated Circuits (ICs):** These pre-assembled components contain multiple logic gates or functional blocks. In this simulation, they perform specific tasks such as AND, OR, or NOT operations, depending on the logic you need to simulate.
- **LED Diode:** The LED (Light Emitting Diode) lights up to indicate the output of the circuit. Typically, an LED will be used to show whether the output is high (ON) or low (OFF), making it easier to visualize the circuit's behavior.
- **Voltage Source:** This component provides the necessary voltage to power the circuit. The voltage source ensures that all components receive the power they need to function properly.

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

Assembled circuit:

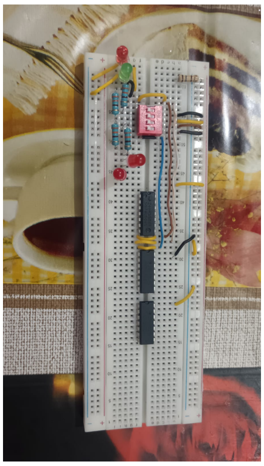


Figure 36: Assembled of exercise 2. Source: Toctaguano (2024)

5.3 Excercise 5

Make a circuit for A + B + C $\cdot$ $\overline{D}$ using 7408 for C $\cdot$ D, 7404 for negation and 7432 for the sum of the three terms. For the following exercise, some simulations were carried out:

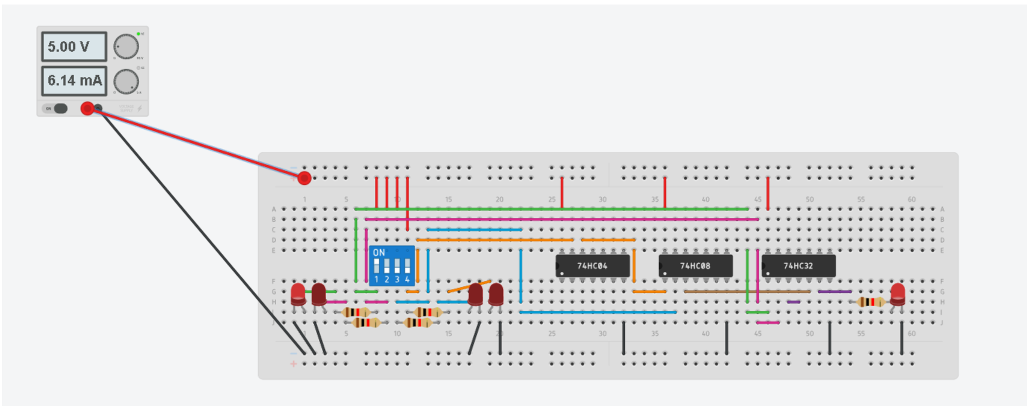


Figure 37: Excercise 5 in TinkerCAD. Source: Troya (2024)

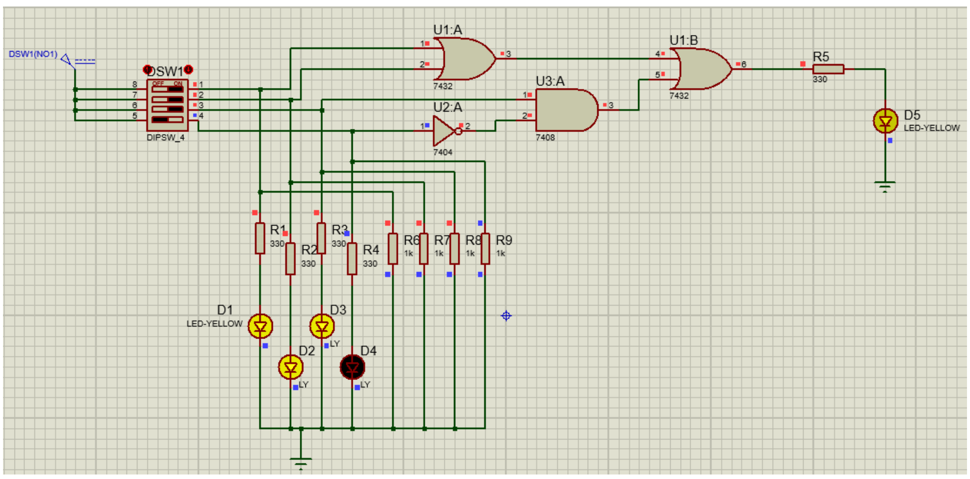


Figure 38: Exercise 5 in Proteus. Source: Troya (2024)

In this simulation, you can see the requested function. The simulation has components such as: resistors,

Dipswitch, integrated circuits, LED diode and a voltage source.
Truth table:

A	B	C	D	A + B	$\overline{D}$	C $\cdot$ $\overline{D}$	f
0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	0	0	0
0	1	0	0	1	1	0	1
0	1	0	1	1	0	0	1
0	1	1	0	1	1	0	1
0	1	1	1	1	0	0	1
1	0	0	0	1	1	1	1
1	0	0	1	1	0	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	0	0	1
1	1	0	0	1	1	0	1
1	1	0	1	1	0	0	1
1	1	1	0	1	1	0	1
1	1	1	1	1	0	0	1

Table 3: Results of exercise 5 - Theoric. Source: Troya (2024)

Assembled circuit:

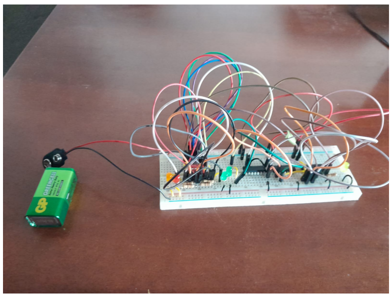


Figure 39: Assembled of exercise 5. Source: Troya (2024)

5.4 Excercise 6

Implement a circuit which complies $\overline{A} \cdot (B + C)$. Use the 7404 to inverse A, the 7432 for B + C, and the 7408 to multiply the results. For this excercise, were made the next simulations:

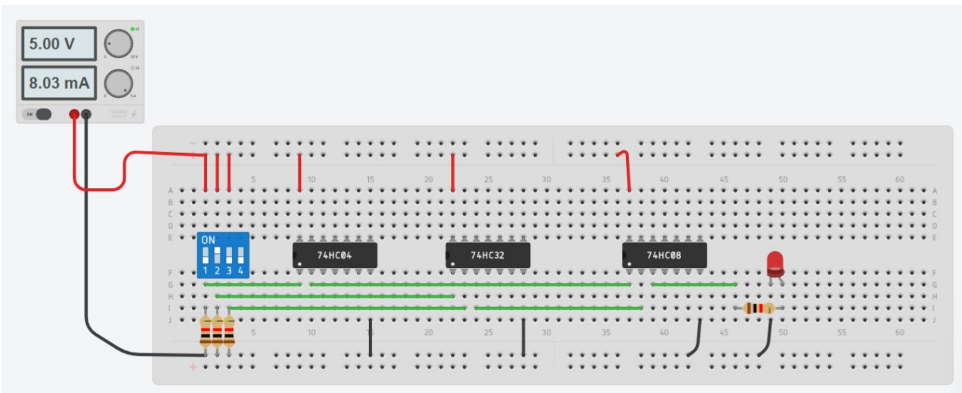


Figure 40: Excercise 6 in TinkerCAD. Source: Torres (2024)

In this simulation, can view the asked function. The simulation has components as: resistances, dipswitch,

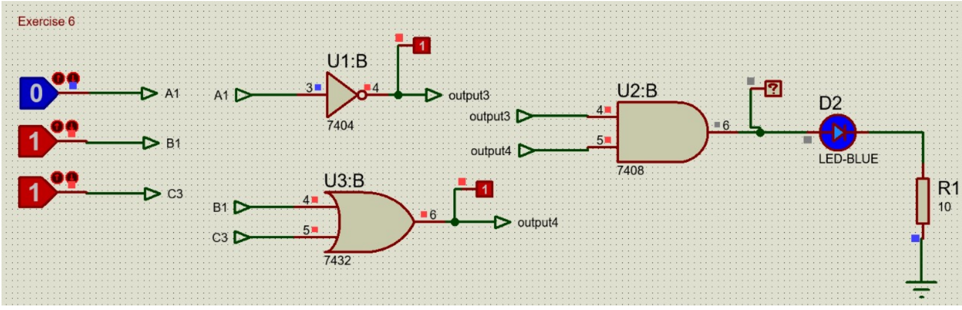


Figure 41: Exercise 6 in Proteus. Source: Torres (2024)

integrated circuits, led diode and a voltage source.

Truth table:

A	B	C	$\overline{A}$	B + C	$\overline{A} \cdot (B + C)$
0	0	0	1	0	0
0	0	1	1	1	1
0	1	0	1	1	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	0	0	0

Table 4: Results of exercise 6 - Theoric. Source: Torres (2024)

Assembled circuit:

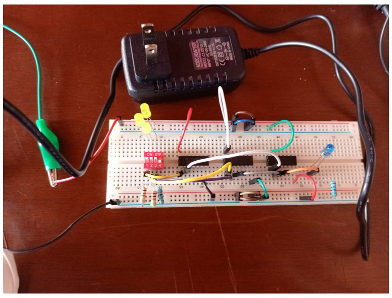


Figure 42: Assembled. Source: Torres (2024)

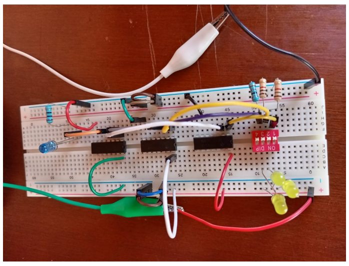


Figure 43: Result 1. Source: Torres (2024)

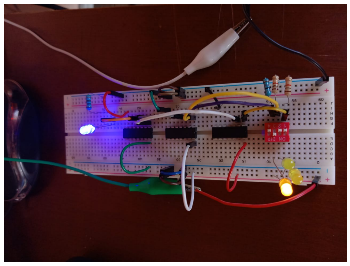


Figure 44: Result 2. Source: Torres (2024)

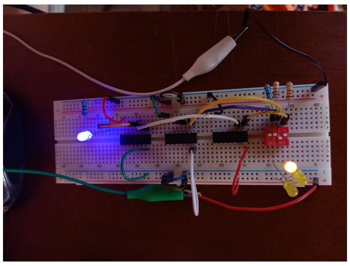


Figure 45: Result 3. Source: Torres (2024)

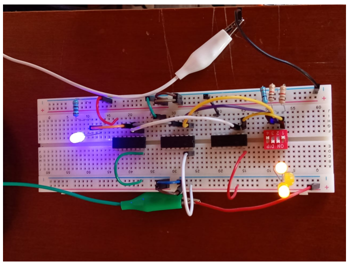


Figure 46: Result 4. Source: Torres (2024)

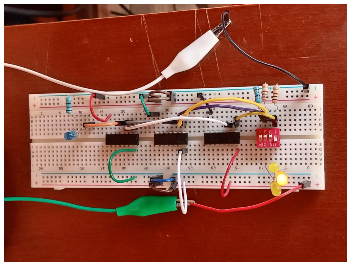


Figure 47: Result 5. Source: Torres (2024)

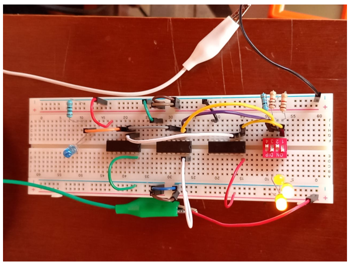


Figure 48: Result 6. Source: Torres (2024)

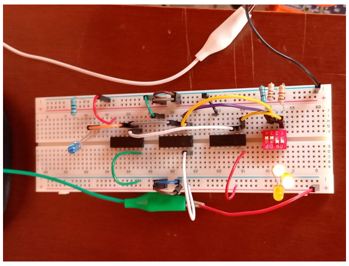


Figure 49: Result 7. Source: Torres (2024)

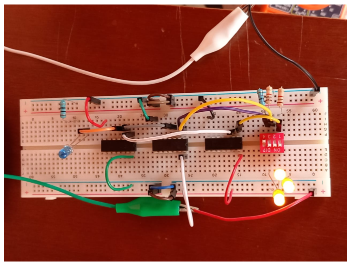


Figure 50: Result 8. Source: Torres (2024)

5.5 Excercise 7

Create the logical function $(A \cdot B) + (C \cdot D)$. Use two AND gates (7408), one for $A \cdot B$ and another for $C \cdot D$ with the 7404, and the 7432 to combine both terms.

5.5.1 Exercise 7: $(A \cdot B) + (C \cdot D)$

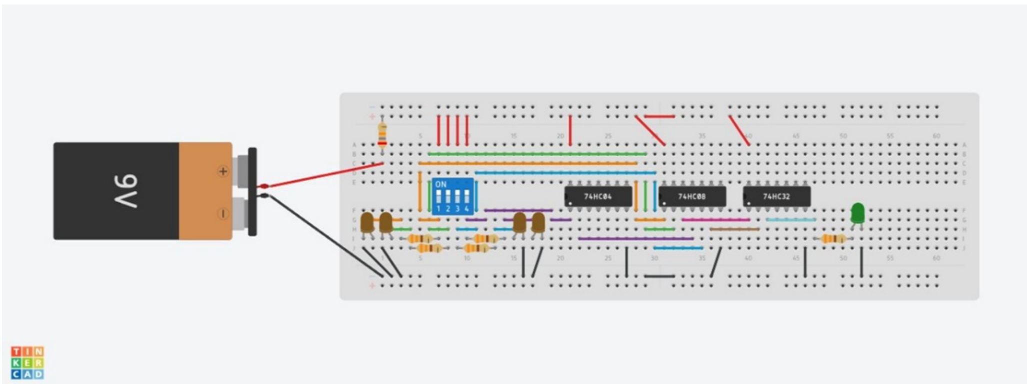


Figure 51: Excercise 7 in TinkerCAD. Source: Toctaguano (2024)

In this simulation, as in the previous one, you can see the required function. The simulation includes components such as resistors, a dipswitch, integrated circuits, an LED diode and a voltage source.

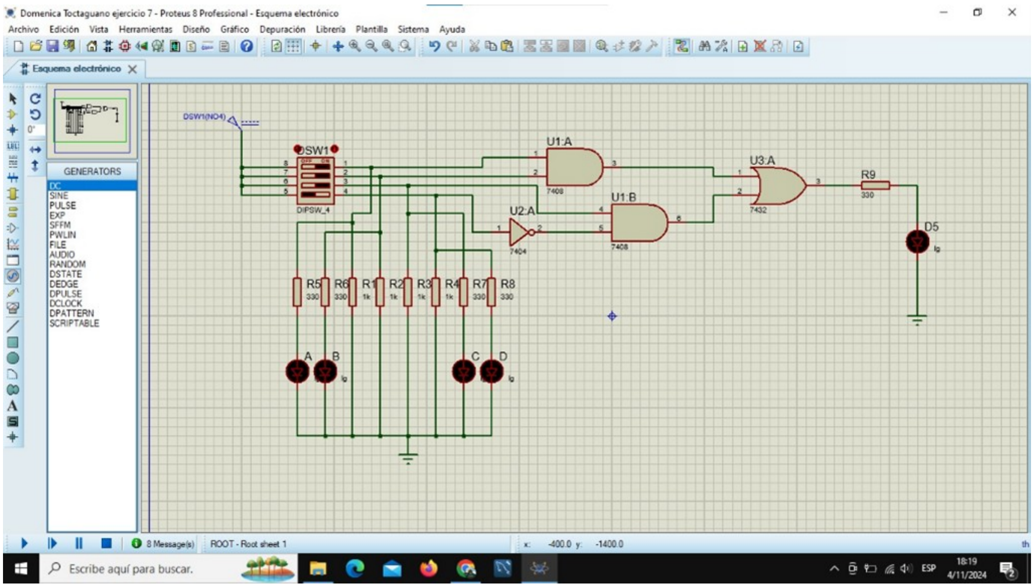


Figure 52: Excercise 7 in Proteus. Source: Toctaguano (2024)

A	B	C	D	$A \cdot B$	$\overline{D}$	$C \cdot D$	$\overline{f}$
0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	1	0	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	1	0	0
0	1	1	1	0	0	1	1
1	0	0	0	0	1	0	0
1	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0
1	0	1	1	0	0	1	1
1	1	0	0	1	1	0	1
1	1	0	1	1	0	0	1
1	1	1	0	1	1	0	1
1	1	1	1	1	0	1	1

Table 5: Truth table of exercise 7. Source: Toctaguano (2024)

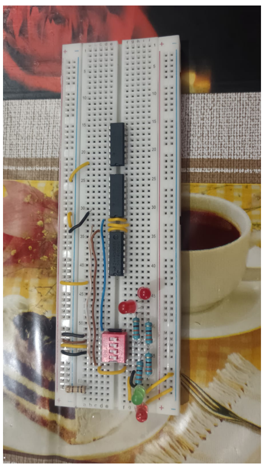


Figure 53: Assembled of exercise 7. Source: Toctaguano (2024)

5.6 Excercise 8

Design the circuit for $(A+B)\cdot\overline{C}$. Use 7432 for $A+B$, 7408 to multiply with C, and 7404 to invert the result.

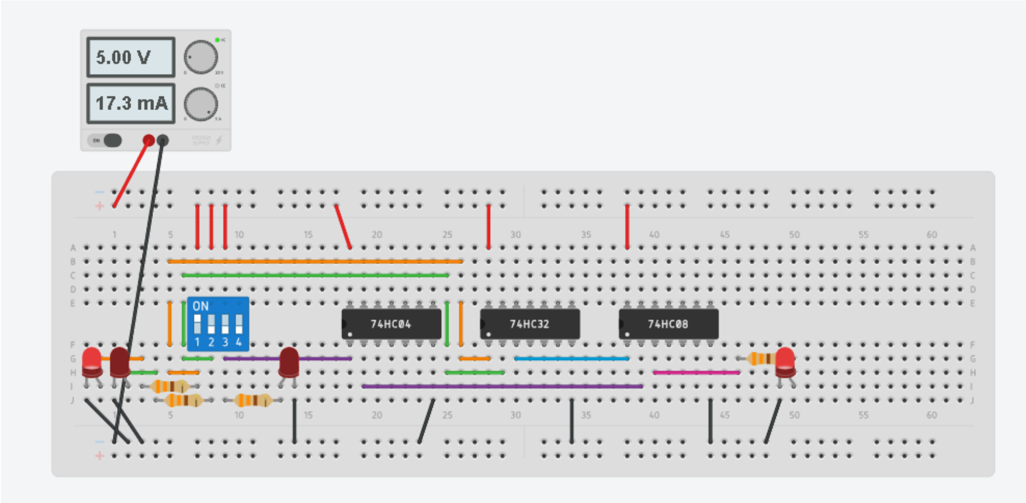


Figure 54: Excercise 8 in TinkerCAD. Source: Troya (2024)

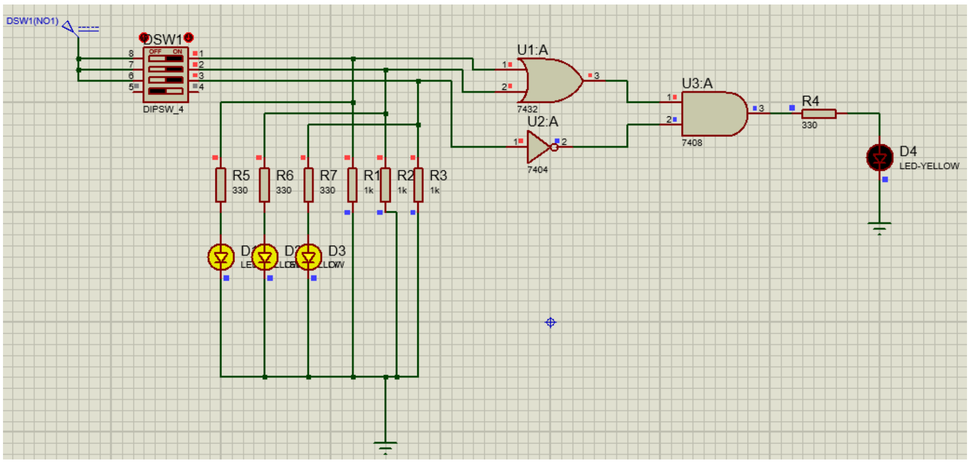


Figure 55: Exercise 8 in Proteus. Source: Troya (2024)

In this simulation, like the previous one, you can see the requested function. The simulation has components such as: resistors, Dipswitch, integrated circuits, LED diode and a voltage source. **Truth table:**

A	B	C	A + B	$(A+B)\cdot\overline{C}$
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	1
1	0	0	1	0
1	0	1	1	1
1	1	0	1	0
1	1	1	1	1

Table 6: Results of exercise 8 - Theoric. Source: Troya (2024)

Assembled circuit:

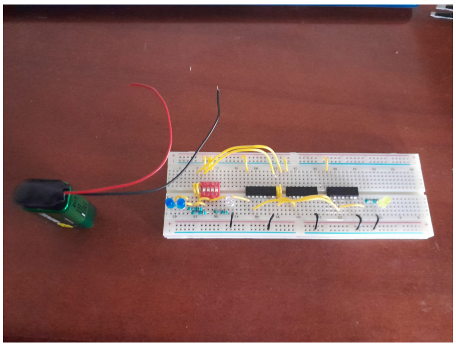


Figure 56: Assembled of exercise 8. Source: Troya (2024)

6 Interpretation of results / Discussion

6.1 Exercise 1

In this exercise, the results are correct with the verification realized with the assembled circuit, because each gate has its own operation. For this case, used the gates 7408, 7404 and 7432. For operation $A \cdot B$ the first two inputs were connected to the two inputs of gate 7408 and for $\overline{C}$, the third input was connected in the first input of gate 7404. The output of gate 7408 was connected in the first input of gate 7432 and the output of gate 7404 was connected in the second input of the same gate, so, both outputs will be summed. The operations are:

Inputs: A = 0, B = 0, C=0
First operation:
$\mathbf{A \cdot B = 0 \cdot 0 = 0}$
Second operation:
$\mathbf{\overline{C} = 1}$
Result:
$\mathbf{A \cdot B + \overline{C} = 0 + 1 = 1}$

Figure 57: Demonstration of operations in exercise 1. Source: Torres (2024)

Will be with the possible inputs of the exercise.

6.2 Exercise 2

- When $A = 0$ and $B = 0$:
  - The OR gate $(A + B)$ produces 0.
  - The NOT gate inverts the output, resulting in $\overline{0} = 1$.
  - The LED will light up, indicating a high output (1).
- When $A = 0$ and $B = 1$:
  - The OR gate $(A + B)$ produces 1.
  - The NOT gate inverts the output, resulting in $\overline{1} = 0$.
  - The LED will be off, indicating a low output (0).
- When $A = 1$ and $B = 0$:
  - The OR gate $(A + B)$ produces 1.
  - The NOT gate inverts the output, resulting in $\overline{1} = 0$.
  - The LED will be off, indicating a low output (0).
- When $A = 1$ and $B = 1$:
  - The OR gate $(A + B)$ produces 1.

- The NOT gate inverts the output, resulting in $\overline{1} = 0$.
- The LED will be off, indicating a low output (0).

6.3 Exercise 5

In this circuit assembly, gates such as 7408 (AND), 7404 (NOT) and 7432 (OR) were used to perform the corresponding logical operations. First, the input ports A and B were connected to the two inputs of the AND gate to achieve the result of the A B operation. Next, the C input was connected to a NOT gate, inverting its signal. The output of the AND gate (A · B) was connected to the input of the OR gate, while the output of the NOT gate (inversion of C) was connected to the other input port of the same OR. In this way, the circuit assembly combines both outputs, achieving a final result of 1 if A · B is 1 or if C is 0, checking the proper functioning of the gates.

6.4 Exercise 6

In this exercise, the results are correct with the verification realized with the assembled circuit, because each gate has its own operation. For this case, for $\overline{A}$ the first input was connected in the input of gate 7404, for the operation (B + C), the second input was connected in the first input of gate 7432 and the third input was connected in the second input of gate 7432. The output of gate 7404 was connected in the first input of gate 7408, the output of gate 7432 was connected in the second input of gate 7408, so, both outputs will be multiplied. The operations are: Will be with the possible inputs of the exercise.

Inputs: A = 0, B = 0, C=0
First operation:
$\overline{A} = 1$
Second operation:
B + C = 0 + 0 = 0
Result:
$\overline{A} \cdot (B + C) = 1 \cdot 0 = 0$

Figure 58: Demonstration of operations in exercise 6. Source: Torres (2024)

6.5 Exercise 7

In this exercise, the results in the table were verified against the assembled circuit, and all values matched as expected, demonstrating the correct operation of each logic gate. For this circuit, the following connections were made:

- **Inversion of A:** The input for A was connected to the input of the 7404 gate (NOT gate) to obtain $\overline{A}$, generating the inverted value of A as needed.
- **OR Operation for $B + C$:** The second and third inputs, representing B and C respectively, were connected to the first and second inputs of gate 7432 (OR gate) to perform the $B + C$ operation. The output of this gate provided the logical OR of B and C.
- **AND Operation Combining $\overline{A}$ and $B + C$:** The output from the 7404 gate (representing $\overline{A}$) was connected to the first input of gate 7408 (AND gate). Simultaneously, the output from the 7432 gate (representing $B + C$) was connected to the second input of the same AND gate (7408). This connection performed the logical AND operation, effectively multiplying the outputs of $\overline{A}$ and $B + C$ to yield the desired output.

6.6 Exercise 8

Within this assembled circuit, the results obtained were validated, since each gate performs its precise logical function. Integrated circuits 7408 (AND), 7404 (NOT) and 7432 (OR) were used. The first for the A - B operation, the first two input ports were connected to the AND gate on the 7408. Input C was destined to the NOT gate on the 7404 to invert its signal. The output of the AND gate was connected to one of its input ports of the OR gate on the 7432, while the output of the NOT was connected to the other input point of the same OR. This helps that both outputs can be combined, making the result true if A - B is satisfied or if the inversion of C is 1.

7 Conclusions and recommendations

Conclusions

- The use of the logic gates permit to obtain great knowledge about the functions which realize much electronic devices and how that are internally composed to obey different orders in diverse fields. The laboratory practice showed us that with the help of the truth tables which favored the comprobation of the correct operation of the circuit which permits to detect in which part of an incorrect way.
- Practicing with simulations and real circuits enhances the understanding of theoretical concepts, such as logical operations and the use of specific components. Seeing how design decisions affect the final outcomes helps bridge the gap between theory and practical application.
- Laboratory work helps build essential technical skills, such as interpreting truth tables, using simulation tools, and troubleshooting in real-time. Encountering and resolving errors in circuits provides valuable problem-solving experience and enhances diagnostic and adjustment abilities.
- The laboratory practice demonstrated that with the help of the use of truth tables it favored the verification of the correct functioning of the circuit since it allows to detect in which part of the assembly it is functioning incorrectly.
- During the lab session, it's crucial to document each step, including the connections made and the outcomes observed. Keeping a detailed record aids in troubleshooting and provides a reflective approach to the design process. Additionally, analyzing the results strengthens the learning experience.

Recommendations

- We recommend to check the function of the integrated circuits with its gates, with the objective to build a circuit without risks.
- In a laboratory setting, it's essential to perform several tests on the circuit by varying the inputs and ensuring the results remain consistent. This helps confirm that the circuit functions correctly in all possible scenarios and deepens the understanding of how it works.
- Before implementing the assembly of the circuit, it is very important to carry out the respective simulations in different software in order to avoid risks at the time of assembly.

References

EEPOWER. (n.d.). Pull-up and pull-down resistors — resistor applications — resistor guide. https://eepower.com/resistor-guide/resistor-applications/pull-up-resistor-pull-down-resistor/#

Electrónica-Unicrom. (n.d.). Compuerta not - compuerta lógica not - electrónica unicrom. https://unicrom.com/compuerta-logica-not/

Ingeniería-Mecafenix. (2023, April). Compuerta lógica or: Símbolos, datasheet y tabla de verdad. https://www.ingmecafenix.com/electronica/digital/compuerta-logica-or/

MecatrónicaLATAM. (2021a, May). Compuerta and — mecatrónicalatam. https://www.mecatronicalatam.com/es/tutoriales/electronica/compuertas-logicas/compuerta-and/

MecatrónicaLATAM. (2021b, May). Compuerta or — mecatrónicalatam. https://www.mecatronicalatam.com/es/tutoriales/electronica/compuertas-logicas/compuerta-or/