



# UNIVERSITÀ DEGLI STUDI DI MILANO

## FACOLTÀ DI SCIENZE E TECNOLOGIE

### Lab Report: Logic Gates

Lorenzo LIUZZO

Bachelor's Degree: Physics

Course: Electronics Laboratory

Academic Year: 2022/2023

Instructor: Professor Valentino LIBERALI

Partners: Jiahao MIAO

Riccardo SALTO

Date Performed: May 24, 2023

## Abstract

In the realm of modern electronics, logic gates serve as the basic building blocks that underpin the entire digital landscape. Composed of transistors and various electronic components, logic gates enable the manipulation and processing of binary information, the driving force behind computers, smartphones, and numerous other digital systems.

This report embarks on a journey to unveil the principles governing logic gates, their operations, and practical applications. The various types of logic gates — NOT, AND, OR, NAND, NOR, and XOR — will be explored, each characterized by its distinct behavior and role within digital circuits. The goal of the experience was to build and test logic gates.

# Contents

|     |   |    |
|-----|---|----|
| 1   | Theoretical background . . . . .              | 2  |
| 1.1 | Resistor-Transistor Logic . . . . .           | 2  |
| 1.2 | Bipolar Junction Transistor . . . . .         | 2  |
| 1.3 | Boolean Algebra . . . . .                     | 3  |
| 2   | Basic Logic Gates . . . . .                   | 4  |
| 2.1 | NOT . . . . .                                 | 4  |
| 2.2 | AND . . . . .                                 | 4  |
| 2.3 | OR . . . . .                                  | 6  |
| 3   | Universal Logic Gates . . . . .               | 8  |
| 3.1 | NAND . . . . .                                | 8  |
| 3.2 | NOR . . . . .                                 | 8  |
| 3.3 | Basic Logic Gates form NAND and NOR . . . . . | 9  |
| 4   | Other Logic Gates . . . . .                   | 10 |
| 4.1 | XOR . . . . .                                 | 10 |

# 1 Theoretical background

## 1.1 Resistor-Transistor Logic

Resistor-Transistor Logic (RTL) is a fundamental digital logic family extensively used in electronic circuit design. It leverages transistors for logic gates and integrates resistors for biasing and signal conditioning. This straightforward and cost-effective approach to building digital circuits lays the groundwork for advanced logic families and integrated circuits.

## 1.2 Bipolar Junction Transistor

The Bipolar Junction Transistor (BJT) consists of three layers: the emitter, the base, and the collector. N-type and P-type materials, each doped with different impurities, compose these layers. The emitter and collector are doped oppositely, either with excess electrons or holes, while the base remains lightly doped. This setup creates two types of BJTs: NPN and PNP, defined by the arrangement of the layers.

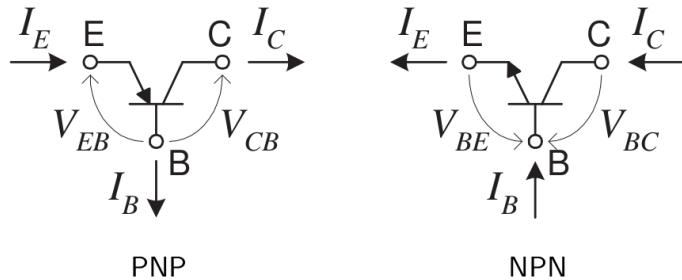


Figure 1: PNP and NPN Bipolar Junction Transistors

In its off state, the BJT functions as a switch, effectively isolating the collector from the emitter. This behavior is consistent for both NPN and PNP BJTs. During this state, the collector-emitter junction acts as a barrier, preventing any significant current flow, much like a dam restraining water. Additionally, the base-emitter junction remains non-conductive, blocking the flow of current from the emitter to the collector.

Transitioning to the active state, the BJT takes on the role of a signal amplifier. A small current flows from the emitter to the base, giving rise to a more substantial current that courses from the collector to the emitter. This relationship between the base and collector currents forms the cornerstone of the BJT's amplification capability.

This amplification phenomenon hinges on the controlled movement of charge carriers across the emitter-base junction. As electrons (or holes) traverse this junction, they engage with the majority carriers in the adjacent layer, resulting in the creation of localized charge regions. These regions facilitate the flow of current from the collector to the emitter.

In the NPN BJT, electrons cross from the emitter to the base. The thinness of the base region and the attractive force exerted by the positively doped collector allow some electrons to overcome the barrier and reach the collector. The modest base current exerts a substantial influence, enabling small variations in its magnitude to produce significant changes in the collector current. Similarly, the PNP BJT witnesses the flow of holes from the emitter to the base, and the base current governs the larger collector current. In this configuration, the base-emitter junction, now comprised of N-type emitter material and P-type base material, emulates the behavior exhibited by the NPN counterpart. Holes traverse the base region, ultimately converging at the collector, thus amplifying the overall current.

As we increase the base current, a point is reached where the collector current can't increase further, regardless of the base current's rise. This state is called saturation. It's as if the BJT switch is turned on fully, allowing maximal current flow through the collector-emitter path.

On the opposite side, as the base current drops, the collector current also decreases until it nearly vanishes. This state is known as cutoff, where the BJT operates as a near-perfect insulator, stopping any significant current from flowing through the collector-emitter junction.

### 1.3 Boolean Algebra

Boolean algebra is a fundamental branch of mathematics that deals with binary variables and logical operations, providing a formal framework for analyzing and manipulating logical expressions. Binary variables can take on one of two values: 0 or 1, representing, respectively, false and true. These binary values correspond to the states of a switch, a circuit element, or a logical proposition.

Boolean algebra uses logical operators such as conjunction (and) denoted as  $\wedge$ , disjunction (or) denoted as  $\vee$ , and the negation (not) denoted as  $\neg$ . Boolean algebra is characterized by several key properties that enable the manipulation and simplification of logical expressions. Some of the fundamental properties include:

#### Commutative Property:

$$\begin{aligned} \text{AND: } A \wedge B &= B \wedge A \\ \text{OR: } A \vee B &= B \vee A \end{aligned}$$

#### Associative Property:

$$\begin{aligned} \text{AND: } (A \wedge B) \wedge C &= A \wedge (B \wedge C) \\ \text{OR: } (A \vee B) \vee C &= A \vee (B \vee C) \end{aligned}$$

#### Distributive Property:

$$\begin{aligned} \text{AND over OR: } A \wedge (B \vee C) &= (A \wedge B) \vee (A \wedge C) \\ \text{OR over AND: } A \vee (B \wedge C) &= (A \vee B) \wedge (A \vee C) \end{aligned}$$

#### Identity Property:

$$\begin{aligned} \text{AND: } A \wedge 1 &= A \\ \text{OR: } A \vee 0 &= A \end{aligned}$$

#### Complement (Inverse) Property:

$$\begin{aligned} \text{AND: } A \wedge \neg A &= 0 \\ \text{OR: } A \vee \neg A &= 1 \end{aligned}$$

#### Double Negation:

$$\neg(\neg A) = A$$

## 2 Basic Logic Gates

There are three basic logic gates from which any Boolean circuit may be built up. Any function in binary mathematics may be implemented using only the logic gates NOT, AND, and OR.

### 2.1 NOT

The NOT gate, also known as an inverter, is the simplest type of logic gate. The circuitry consists of a single transistor along with some associated resistors, as shown in Figure 2.

It is equivalent to the logical negation operator ( $\neg$ ) in mathematical logic, as its primary function is to output the logical complement of the input. Its operation can be algebraically represented as  $\text{NOT}(A) = \bar{A}$ , while its analytical representation is  $f(A) = 1 - A$ , where  $A$  is the input signal. When the input is high (1), the transistor is turned on, creating a low output (0). Conversely, when the input is low (0), the transistor is off, resulting in a high output (1). This behavior is consistent with the NOT gate's truth table shown in Table 1.

The symbol for the NOT gate, in Figure 3, consists of a triangle with a circle at its output.

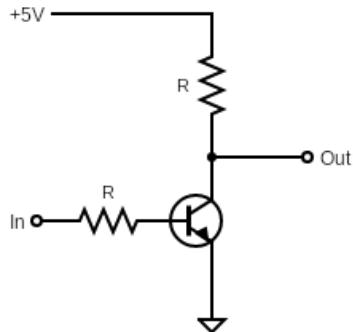


Table 1: NOT truth table.

| Input | Output |
|-------|--------|
| 0     | 1      |
| 1     | 0      |

Figure 2: NOT schematic circuit.

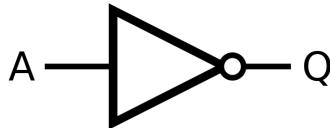


Figure 3: NOT symbol.

### 2.2 AND

The AND gate implements logical conjunction ( $\wedge$ ) from mathematical logic. Its operation can be represented as  $\text{AND}(A, B) = A \cdot B$ , where  $A$  and  $B$  are the input signals.

The circuitry of an AND gate involves multiple transistors arranged in series. For a two-input AND gate, two transistors are connected in series, as shown in Figure 4.

If both inputs are high (1), both transistors conduct, creating a low-resistance path from the power supply to the output, resulting in a high output (1). If either or both inputs are low (0), at least one of the transistors will be off, and the output will be pulled to a low state (0). This behavior is consistent with the AND gate's truth table shown in Table 2.

The symbol for the AND gate is shown in Figure 5.

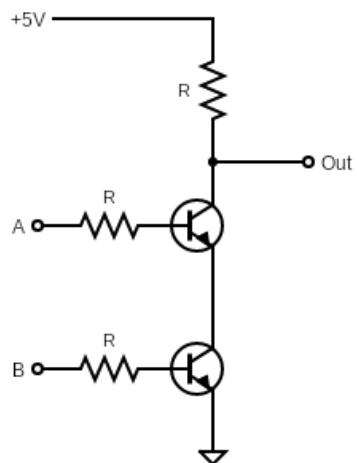


Table 2: AND truth table.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 0      |
| 0       | 1       | 0      |
| 1       | 0       | 0      |
| 1       | 1       | 1      |

Figure 4: AND schematic circuit.

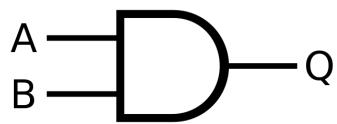


Figure 5: AND symbol.

The following photographs show the AND gate built in the laboratory and its behavior.

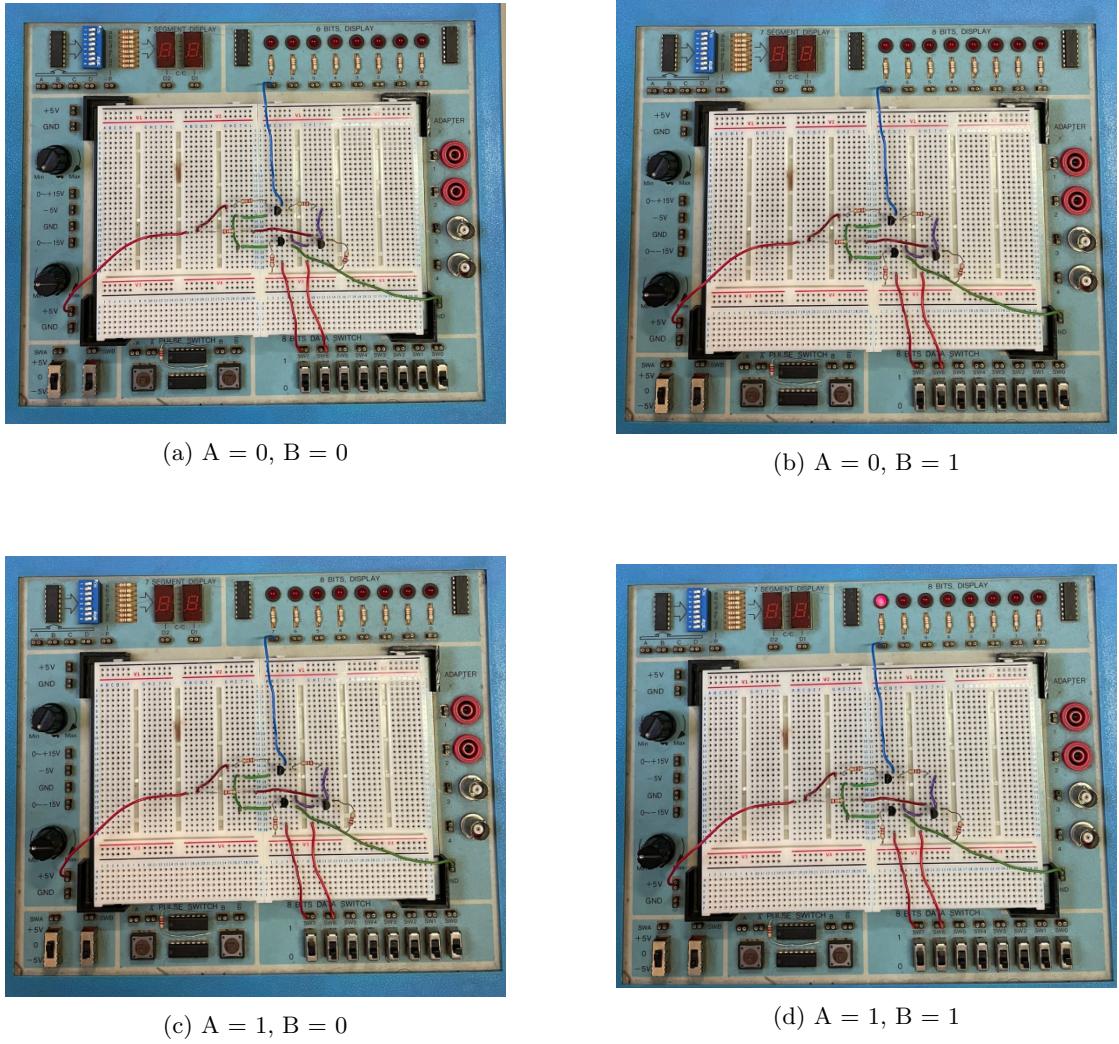


Figure 6: AND gate constructed in the laboratory.

### 2.3 OR

The OR gate represents the logical disjunction ( $\vee$ ) from mathematical logic, and its operation can be represented as  $OR(A, B) = A + B$ , where  $A$  and  $B$  are the input signals.

The circuitry of an OR gate involves multiple transistors arranged in parallel. For a two-input OR gate, two transistors are connected in parallel, as shown in Figure 8.

If either or both inputs are high (1), at least one of the transistors conducts, providing a low-resistance path from the power supply to the output, resulting in a high output (1). Only when both inputs are low (0) will both transistors be off, causing the output to be pulled to a low state (0). This behavior is consistent with the OR gate's truth table shown in Table 3.

The symbol for the OR gate is shown in Figure 7.

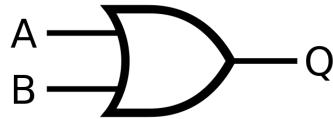


Figure 7: OR symbol.

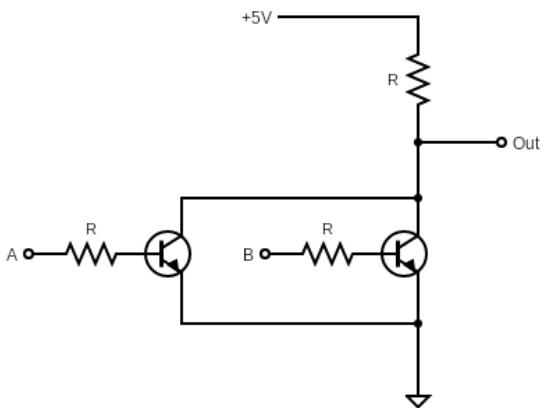
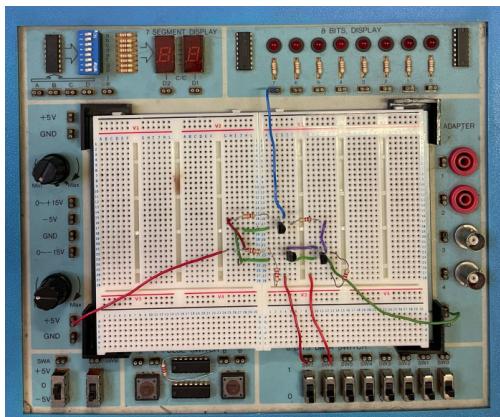


Table 3: OR truth table.

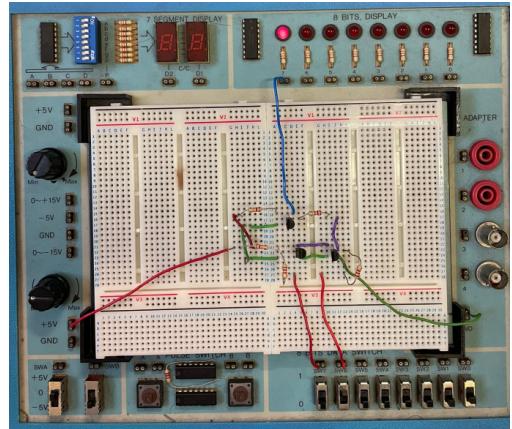
| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 0      |
| 0       | 1       | 1      |
| 1       | 0       | 1      |
| 1       | 1       | 1      |

Figure 8: OR schematic circuit.

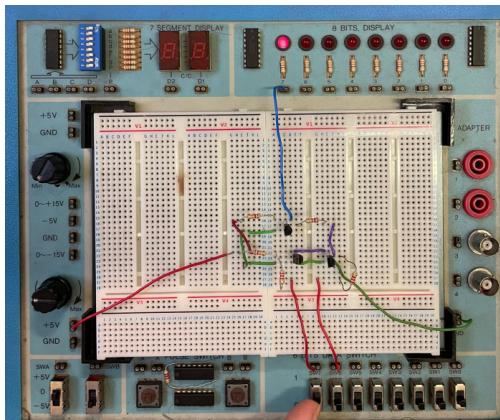
The following photographs show the OR gate built in the laboratory and its behavior.



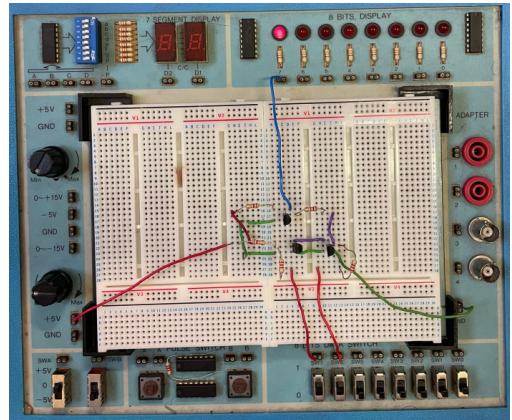
(a)  $A = 0, B = 0$



(b)  $A = 0, B = 1$



(c)  $A = 1, B = 0$



(d)  $A = 1, B = 1$

Figure 9: OR gate constructed in the laboratory.

### 3 Universal Logic Gates

#### 3.1 NAND

The NAND gate, short for NOT-AND, is a versatile logic gate that can be obtained by connecting an AND gate and a NOT gate in series, as shown in Figure 10.

The NAND gate's operation can be algebraically represented as  $\text{NAND}(A, B) = \overline{A \cdot B}$ , where  $A$  and  $B$  are the input signals. When all inputs are high (1), all the transistors in the AND part conduct, resulting in a high output (1) for the AND operation. The NOT gate then inverts this high output to a low final output (0). This behavior is consistent with the NAND gate's truth table shown in Table 4.

The symbol for the NAND gate is shown in Figure 11.

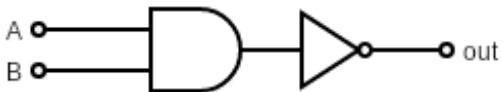


Table 4: NAND truth table.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 1      |
| 0       | 1       | 1      |
| 1       | 0       | 1      |
| 1       | 1       | 0      |

Figure 10: NAND gate.



Figure 11: NAND symbol.

#### 3.2 NOR

The NOR gate, short for NOT-OR, can be obtained by connecting an OR gate and a NOT gate in series, as shown in Figure 12.

Its operation can be algebraically represented as  $\text{NOR}(A, B) = \overline{A + B}$ , where  $A$  and  $B$  are the input signals. When all inputs are low (0), all the transistors in the OR part are off, resulting in a low output (0) for the OR operation. The NOT gate then inverts this low output to a high final output (1). This behavior is consistent with the NOR gate's truth table shown in Table 5.

The symbol for the NOR gate is shown in Figure 13.

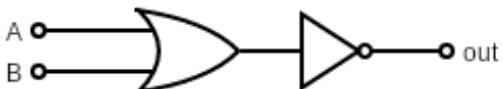


Table 5: NOR truth table.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 1      |
| 0       | 1       | 0      |
| 1       | 0       | 0      |
| 1       | 1       | 0      |

Figure 12: NOR gate.



Figure 13: NOR symbol.

### 3.3 Basic Logic Gates form NAND and NOR

Both NAND and NOR gates are considered “universal gates”. This means that any logical function can be constructed using either NAND logic or NOR logic alone. In other words, complex circuits and logical operations can be built using only one type of gate.

In this subsection, it is shown how to construct the NOT (Figures 14 and 15), AND (Figures 16 and 17), and OR (Figures 18 and 19) gates using only NAND gates and only NOR gates.



Figure 14: NOT gate from a NAND gate.



Figure 15: NOT gate from a NOR gate.

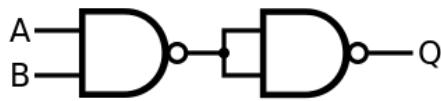


Figure 16: AND gate from NAND gates.

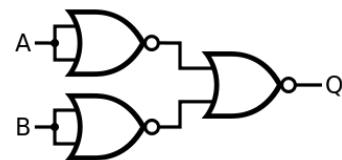


Figure 17: AND gate from NOR gates.

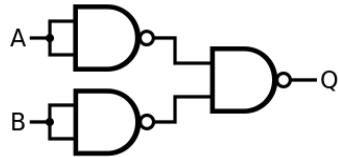


Figure 18: OR gate from NAND Gates.

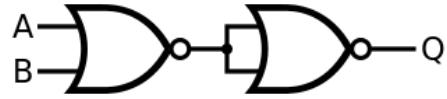


Figure 19: OR gate from NOR Gates.

## 4 Other Logic Gates

### 4.1 XOR

The XOR gate, short for Exclusive OR, represents the expression  $(A \cdot \bar{B}) + (\bar{A} \cdot B)$ . It can be constructed using AND, OR and NOT gates, however, this approach would require five gates of three different kinds. As an alternative, the XOR gate can be made of just universal gates: from four NAND gates, as shown in Figure 20, or from five NOR gates, as shown in Figure 21.

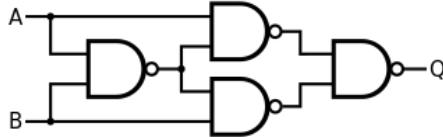


Figure 20: XOR gate from NAND gates.

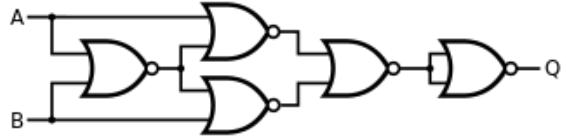


Figure 21: XOR gate from NOR gates.

An analytical representation of XOR gate could be  $f(A, B) = A + B - 2AB$ . The XOR gate gives a high output if one, and only one, of the inputs to the gate is high. This behavior is consistent with the XOR gate's truth table shown in Table 6.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 0      |
| 0       | 1       | 1      |
| 1       | 0       | 1      |
| 1       | 1       | 0      |

Table 6: XOR truth table.



Figure 22: XOR symbol.

In the laboratory, it was used a different approach to build the XOR gate, using only four transistors, as shown in Figure 23.

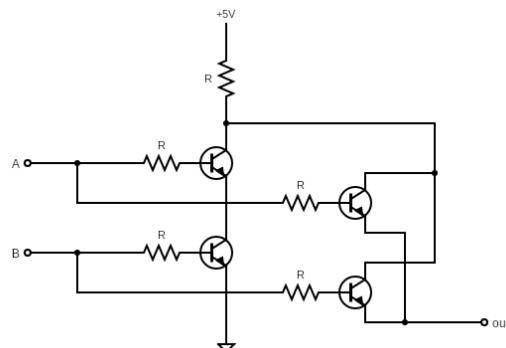
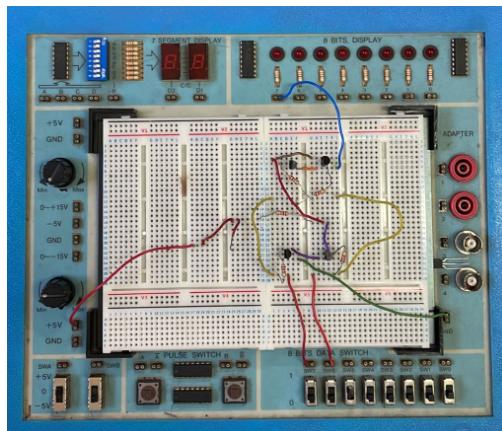
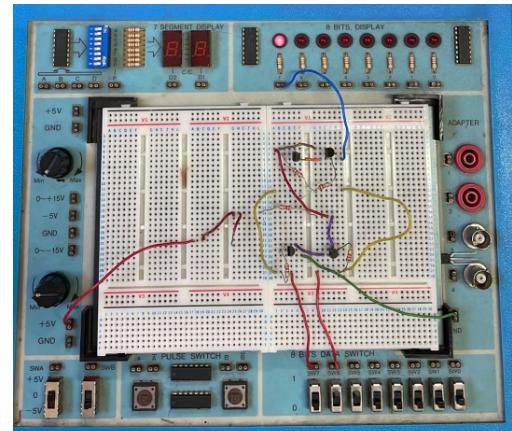


Figure 23: XOR gate.

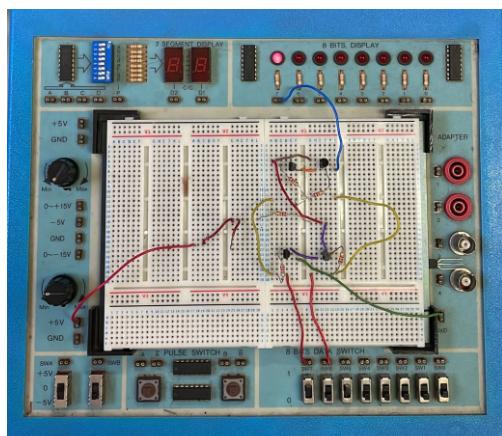
The following photographs show the XOR gate built in the laboratory and its behavior.



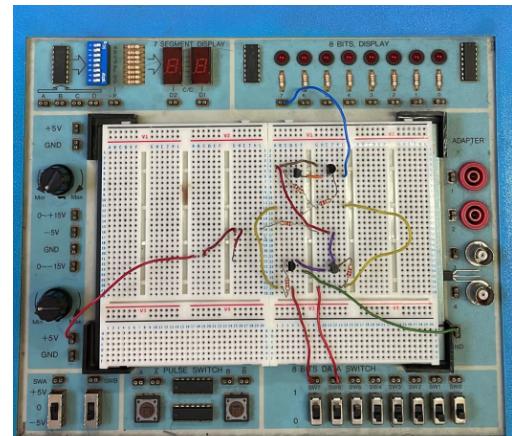
(a)  $A = 0, B = 0$



(b)  $A = 0, B = 1$



(c)  $A = 1, B = 0$



(d)  $A = 1, B = 1$

Figure 24: Four BJTs XOR gate constructed in the laboratory.