DIGITAL LOGIC CIRCUIT

By Nouh Alkandari

Abstract

Logic gates are the fundamental components of any digital system and can be considered the "building blocks". A logic gate is a simple electric circuit consisting of two inputs and a single output. The most frequent names for logic gates are AND, OR, NOT, XOR (Exclusive or), NAND (NOT AND), and NOR. An OR logic gate begins with the provision of two electrical inputs. If one of the inputs has the value one or indicates that it is "on," then the output will also be one.

In electronics, there is a type of logic gate known as an inverter or NOT gate. The report is broken up into five distinct parts or sections. The first section of this report covers the experiment's results on logic gates. They are used in the process of performing logical operations on one or more binary inputs to produce a single binary output. This article will examine the functions of the NOT, OR, and AND gates found in a logic circuit.

The findings of the experiment are presented in the fourth section. The discussion, recommendations, and conclusions drawn from the results are in the last part. In a NOT gate, the input determines whether the output is true or false, and vice versa. ALTERNATIVELY, gates output a value of HIGH if either of the two inputs is.

HIGH and LOW if both inputs are LOW; this type of gate is also known as an inverter. A truth table was used to validate the information of each NOT, AND, and OR integrated circuit. Knowing how to use these seven fundamental logic gates makes it much simpler to comprehend Boolean algebra and simplifies the process of conducting circuit analysis. These gates are most commonly used in the manufacture of automatic machines. Learning how to design logical circuits was made possible by utilizing gates such as NOT, AND, and OR.

INTRODUCTION.

Logic gates are the fundamental components of any digital system and can be considered the "building blocks." It is a piece of electronic hardware that can accept one or more inputs but produces only a single output ("Basic logic gate and circuit theory," 2013). The things that go into it have a rational relationship to the results that come out of it. This is where the names of various logic gates come from, such as the AND gate, the OR gate, the NOT gate, and so on. [1].

Despite its intimidating name, a logic gate is nothing more than a simple electric circuit consisting of two inputs and a single output. It receives two different electric currents, analyses them by comparing them, and then generates a new currency based on the analysis results. A logic gate functions much like a door attendant or bouncer at a nightclub in that it only admits customers who have completed a series of challenges. The most frequent names for logic gates are AND, OR, NOT, XOR (Exclusive or), NAND (NOT AND), and NOR. There are many more forms of logic gates, but these are the most common ones (NOT OR). Let's start with the three that don't require mental gymnastics to comprehend: AND, OR, and NOT.

AND

Let us imagine that we went to a nightclub where the door attendant was responsible for ensuring that everyone in our party was dressed appropriately and wore a tie. One of our mutual friends extended the invitation to us. You can enter the building if both of us wear ties simultaneously. If none of us is wearing a tie, then neither of you will be allowed entry, but if only one of us is, then neither of you will. An AND logic gate operates similarly regardless of the number of electrical inputs it receives. If both of the information are active, which indicates that they both have the value 1, the output will also be one in this scenario. In that case, the response will be zero. We can demonstrate that something is an AND gate in electronics by using this little sign. The gate can operate in one of the following three ways:

OR

You should head to the club down the street where the men are wearing ties, as we do not require them here. In this situation, the person guarding the entrance is according to a different guideline: "A group of people may enter the building if at least one of them is a member." We can enter together if either of our companion or us is an established member. You will both be left out in the cold if neither of us joins the club. The operation of an OR logic gate begins with the provision of two electrical inputs. If one of the inputs has the value one or indicates that it is "on," then the output will also be 1. In that case, the response will be 0. A distinct symbol denotes the presence of an OR gate in electrical circuitry. As you can see in the following illustration, there are three possible outcomes:

NOT

We have not been successfully gaining entry to either of the clubs yet. However, there is a final effort you can make: you are aware that a friend is throwing a party a few blocks away. The only issue is that someone who is quite obstinate and argumentative is standing at the door. As each approaches him, he engages in conversation with them. He will yell at you and kick you out of the house if you are courteous and friendly to him.

On the other hand, if you treat him with contempt, for some reason, he finds that endearing and will let you enter. To put it another way, he behaves in a manner that is opposed to what you may anticipate he would. In electronics, a logic gate known as an inventor or NOT gate operates in the opposite direction. The fact that it has only one input and one output sets it apart from AND and OR gates in logic circuitry. When the input value is zero, the output value is one, and when the output value is one, the input value is zero. This is the electrical representation of the NOT operator. Here are two possible outcomes that could occur.

These three logic gates are the basis for all three of the other common logic gates, which are only variants of these three. Exclusive OR, sometimes known as XOR, functions similarly to OR but turns off when both inputs are active. The only difference between AND and NAND is in the outcome, which is reversed (so where AND produces an output of 1, NAND produces an output of 0). NOR is equivalent to OR, except that the output is reversed.

Calculators can perform all of their necessary functions because of logic gates that can be assembled in various ways. Logic gates are responsible for the operation of the display on a calculator, while other logic gates compute the calculation results. Let us take a closer look at it.

NAND gate

The NOT-AND gate is functionally equivalent to an AND gate followed by a NOT gate. The output of all NAND gates will be high if there is a low value at any of the inputs. A tiny circle is produced as the output of the symbol. This is a representation of an AND gate. The smaller circle represents the inverted symbol.

NOR gate

This is a NOT-OR gate, which functions similarly to an OR gate placed after a NOT gate. The outputs of all NOR gates will be low if there is a high value at any of the inputs. A tiny circle is produced as the output of the symbol. This is an illustration of an OR gate. The smaller circle represents the inverted symbol.

The Boolean function is the primary conceptual foundation for logic gates, which are virtual digital devices. They are used in the process of performing logical operations on one or more binary inputs to produce a single binary output. In more non-technical language, the electronic circuits that make up a digital system are called logic gates. [2]

After the initial introduction, this report's section covering the experiment's results on logic gates will be presented. The information is broken up into five distinct parts or sections. The first section of this report is the introduction, which consists of a brief synopsis of the subject matter and is presented at the beginning of the document. The second section will discuss the experiment's objectives in detail. The third section will find a survey of the relevant theory and literature. The experiment's findings are presented in the fourth part of the article. The discussion, recommendations, and conclusions that are drawn from the results are shown in the last detail. The references page also includes the sources (material) that were used in the experiment.

OBJECTIVES

Investigate the operations of the NOT, OR, and AND gates found in a logic circuit.

Understand how logic gates, such as AND, OR, and NOT, are designed and how they are wired and used .

Develop your understanding of logic gates to construct basic circuits by following a schematic.

THEORY AND LITERATURE REVIEW

Logic gates are the fundamental building blocks of digital electronics. An electronic device known as a gate is utilized to determine what action should be taken concerning a signal that possesses just two possible states at any given time ("Basic logic gate and circuit theory," 2013). Logic gates, the fundamental constituents of digital circuitry, are the circuit's primary constituents. Because every logic gate has one output and two inputs, the way it operates is the same across the board. Some logic gates, such as the NOT gate and the inverter, only have one input and one output each, respectively. The inputs of the logic gates are set up to receive only binary data, which consists of either a low 0 or a high 1, which is accomplished by taking in voltage. A high logic level indicates a positive supply voltage of 3 or 5 volts, while a low logic level indicates that the supply voltage is 0. There is no limit to the number of logic gates that can be connected to complete the digital circuit that we require. We store many logic gates in integrated circuits (I.C.sI.C.s). This results in savings in the amount of physical space needed for the many logic gates. Integrated circuits also allow us to do intricate jobs rapidly and at high speeds (I.C.I.C.). We can construct various courses by connecting logic gates, including shift registers, multiplexers, flip-flops, and latches("Circuit and logic styles," 2009).

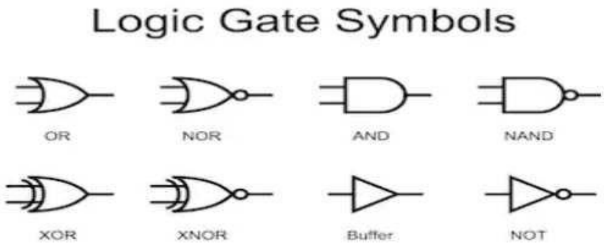
A signal's status or voltage at a particular point is called its logic level. It is common knowledge that logic gates can exist in either the 0 or 1 state. LOW stands for logic level 0, and HIGH stands for logic level 1. These binary logic levels are utilized in digital electronics for information storage and transmission. Generally, you may consider these logic levels to be either ON or OFF. As we have discussed in the past, the logic levels of the logic gate are determined by the supply voltage. If no voltage comes into the logic gate, the logic level is low, or the entrance is in an "off" state. There is no logic level if the voltage goes into the logic gate.

AND gate: The output will be high if both inputs have a high value. The output will be low if either one or both inputs are low. OR gates output a value of HIGH if either of the two inputs is HIGH, while they output a value of LOW if both inputs are LOW.

In a NOT gate, the input determines whether the output is true or false, and vice versa. This type of gate is also known as an inverter.

If only one of the two inputs has a high value, the output will be increased.

The output is low when both inputs are high, and vice versa. This is the behaviour of a NAND gate. If both inputs are standard, the output will be increased, whereas it will be low if both are high.



APPARATUS AND PROCEDURES

.1 x 74LS04 (NOT)

1 x 74LS08 (AND)

1 x 74HC32 (OR)

1 x LED

1 x 330 resistor

1 x Breadboard

PROCEDURES.

On the breadboard, the NOT, AND, and OR gates of the three I.C.sI.C.s were arranged, and a connection was made to a power supply that was 5V D.CD.C. After making the vertical lines ground and the horizontal lines +5v, two express buses emerged from the process. A truth table was utilized to validate the inputs of each NOT, AND, and OR integrated circuit. After connecting the information of each gate to every possible combination of 0V (logical 0) and 5V (logical 1), the output voltages of the gates were measured. The instructor was given a truth table for two and three inputs of I.C.sI.C.s, and each work was presented to them using that table.

RESULTS ANALYSIS.

For part 1, the following results were obtained after all connections respective to each gate.

Two input

Input A	Input B	Output for AND gate (A.B.A.B.)	Output OR gates (A+B)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

c) NOT gate.

Input	output
0	1
1	0

Three input.

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

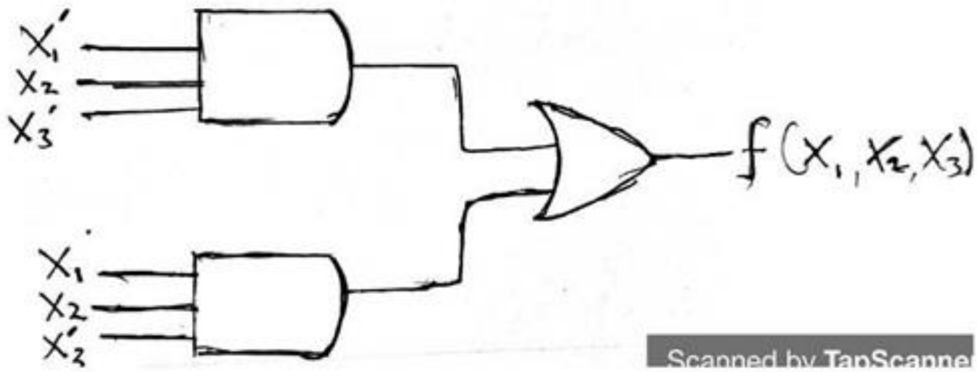
Logic circuit for f (X_1, X_2, X_3) =

Part 2

A	B	C	f
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$f(X_1, X_2, X_3) = X'_1.X_2.X'_3 + X_1.X_2.X'_3$

logical circuit diagram.



DISCUSSION.

The logic gates and the truth table are utilized in the first part of the experiment to determine what steps to take. In an AND gate that had two inputs, the L.E.D.L.E.D. would only light up (or "on") when both of those inputs were high, and it would go dark (or "off") when either of those inputs was low. The same was true for an AND gate that had three inputs; the L.E.D.L.E.D. would only light up (1) when both of the inputs were high, and it would go dark (0) when either one or both of the information was low (0).

Same way. When it comes to an OR gate with two inputs, the L.E.D.L.E.D. will light up if either one is high (1), but it will go dark if both inputs are low (0). When it comes to an OR gate with three inputs, the L.E.D.L.E.D. will only light up if one of the inputs is high (1), but it will go dark if both inputs are low (0). The outcome was as anticipated concerning the NOT gate. Each infusion was toggled whenever either a one or a 0 was entered into the terminal that served as the input.

As seen in figure 1, the AND and OR gates were combined for use in part 2, and the L.E.D.L.E.D. was only activated in the ways specified in the truth table.

RECOMMENDATION.

Before connecting the components and obtaining results, it is essential to verify the functionality of all of the parts, especially the L.E.D.s and the microchips, so that the results obtained from testing the output of each I.C.I.C. are reliable. Additionally, the wires used to connect them ought to be as short as possible to prevent or minimize current loss, which results in heat ("Dynamic logic circuit concepts,").

For example, advancing integrated circuits based on transistors and their configuration has resulted in faster computers. For a long time, logic gates were thought to be capable of running a computer system very quickly. Information is transferred to memory before electrically processing to avoid data loss in conventional microprocessors. This enables the logic gate to store output, improving computation performance.

It is also known that the circuit is noise-proof using modern circuit design techniques. The most advanced design can reduce noise by 1.5 times while saving 81 per cent of the power. A computer circuit can also assist the computer in conserving energy. The computer's circuit comprises a power supply (20) with an output voltage of (P.S.P.S. +5VSB). The computer system consumes less power by utilizing a low-power switch circuit ("Dynamic logic circuit concepts,").

As a result, the internet and computer systems consume approximately 8% of the total electricity generated by computer logic gates. Modern computer logic gates may save power and improve communication because they are specifically designed. Computers must consume the least amount of energy possible, whether portable or stationary. Because of the rapid growth of computer information systems, there are now billions of logic gates capable of storing 64 G.I.B.I.T.S. of memory. This increases the storage capacity of the computer ("Dynamic logic circuit concepts,").

For many years, logic gates have been used to improve computer memory. The computer system's 400 GBIT memory has been enhanced due to logic gates in modern computers. It is also possible to reconstruct the computer system's logic circuits to log any errors that may have occurred. The computer can keep track of more precise operations thanks to logic gates. Because logic circuits do not detect noise, data can be easily stored in a computer's memory. The logic gates are also said to be capable of logically changing the order of computer instructions. The gate can convert computer code into instructions that can be executed promptly.

Logic gates can also increase hardware capacity by strengthening high-speed bus connections, speeding up the hardware processor, and shrinking the size of the logic gates. This increases the number of bits that can be processed simultaneously. More gates can be crammed into a smaller space, altering the processor and computer design. Because logic gates are becoming smaller, multiple stages of instruction can be executed at the same time. In other words, a single processor can handle multiple tasks at once.

Because of improved methods for designing logic circuits, the physical gate has shrunk in size, which is a critical development in increasing the speed of the 10-gigahertz processor module. Because of this significant change, the Pentium M computer was created, which has a speed function of more than 2.2 gigahertz and parts of its chip that can reach 4.4 gigahertz.

It is possible to reduce the effects of designing microchips using the device in gates, significantly reducing the size of electronic computers and their cooling products and simplifying computer systems used to be a difficult task. Insufficient memory and slow processing speed are two problems that plague modern computers. However, these issues have been resolved thanks to the invention of logic gates. Small, portable computers capable of storing large amounts of data and processing it quickly are now widely available. CPUs benefit significantly from our high-speed logic circuit devices regarding data processing. This enables the computer to carry out more complex mathematical calculations.

CONCLUSION.

Logic gates exist in a wide variety of shapes and sizes. Whether used in conjunction with one another or independently, they provide the user with various alternatives and ways to solve problems that may appear to be difficult at first glance. In addition, knowing how to use these seven fundamental logic gates makes it much simpler to comprehend Boolean algebra and simplifies the process of conducting circuit analysis ("Fabrication and layout of CMOS integrated circuits,"). When comparing frequencies to produce filters for communication, for example, or when employing choppers and inverters, which compare input and output currents to discover modulating indices, these gates can be used practically any place. One example is when comparing frequencies to make filters.

All expectations were satisfied, considering the overall purpose of this experiment. Learning how to design logical circuits was made possible through utilizing gates known as NOT, AND, and OR ("Fabrication and layout of CMOS integrated circuits,"). Additionally, the use of digital logic circuit design was investigated and shown to be effective. This demonstrated that logic gates are most commonly used in manufacturing automatic machines, in which the designer can set the conditions according to the user's requirements. All of the gates were successfully connected with the assistance of the truth table as well as the logic circuit diagram.

REFERENCES.

"Logic Gates - Tutorialspoint." https://www.tutorialspoint.com/computer_logical_organization/logic_gates.htm (accessed July. 8, 2022).

"Basic Logic Gates - Types, Functions, Truth Table, Boolean Expressions," B.Y.J.U.S. https://byjus.com/jee/basic-logic-gates/ (accessed July. 8, 2022).

"Introduction to Logic Gates | NOT, AND, NAND, OR, NOR" Electronics Hub, Jun. 19, 2019. https://www.electronicshub.org/introduction-to-logic-gates/ (accessed July. 8, 2022).

"logic-gates-min.png (334×202)." https://physicsabout.com/wp-content/uploads/2018/02/logic-gates-min.png (accessed July. 8, 2022).

Basic logic gate and circuit theory. (2013). CMOS Integrated Digital Electronics: A First Course, 1-37. https://doi.org/10.1049/sbcs501e_ch1

Circuit and logic styles. (2009). Low-Power CMOS Design. https://doi.org/10.1109/9780470545058.part4

Dynamic logic circuit concepts. (n.d.). CMOS Logic Circuit Design, 287-347. https://doi.org/10.1007/0-306-47529-4_7

Fabrication and layout of CMOS integrated circuits. (n.d.). CMOS Logic Circuit Design, 61-102. https://doi.org/10.1007/0-306-47529-4_2