Republic of the Philippines
UNIVERSITY OF RIZAL SYSTEM
MORONG CAMPUS

COLLEGE OF ENGINEERING
ELECTRONICS ENGINEERING

ECE 6

Digital Electonics 1:

Logic Circuits and Switching Theory

Boolean Algebra and Logic Gates

TRAINING MODULE

Compiled by:

Ashanty Madeline DC. Cruz
Brian Ezekiel D. Batalon
Tristan Manuel T. Sto. Tomas
Mark Jayson F. Villacura

Welcome to the SMART LoBo Training Kit Module!

Technology has transformed the way we learn, making knowledge more accessible than ever. In today's world, understanding digital logic is not just an advantage—it's essential. This training module, featuring the hands-on SMART LoBo Training Kit, is designed to make your learning journey engaging and effective, providing you with the resources and activities needed to explore digital logic circuits in depth.

The SMART LoBo Training Kit is a powerful digital logic trainer that brings theoretical concepts to life. By simulating computer circuitry with various ICs, this kit allows you to easily implement logic circuits. Like Logisim, a digital logic simulation software, it enables you to set logic level inputs, connect gates, and observe logic level outputs in real time. The trainer includes a power supply, logic blocks of various gates, a seven-segment display for output indication, logic level switches, and 3-pin JST connectors. It is enclosed in a plastic case with a latch and a handle for added convenience and safety.

As the creators of this module, we are committed to ensuring you gain valuable insights and skills from each section. If you have any questions or need further assistance, please don't hesitate to reach out using the contact options below.

Happy learning, and enjoy discovering the world of digital logic!

ASHANTY MADELINE DC. CRUZ

Email: ashantymadelinecruz@gmail.com

Facebook: https://www.facebook.com/AshantyMadelineCruz/

TRISTAN MANUEL T. STO. TOMAS

Email: tristanmanuelstotomas@gmail.com

Facebook: https://www.facebook.com/tristanmanuel.stotomas

BRIAN EZEKIEL D. BATALON

Email: batalon.brianezekiel@gmail.com

Facebook: https://www.facebook.com/brianezekiel.batalon.7

MARK JAYSON F. VILLACURA

Email: markjaysonvillacura@gmail.com

Facebook: https://www.facebook.com/jaysonvillacura0207

TABLE OF CONTENTS

COURSE CODE: ECE6

COURSE TITLE: DIGITAL ELECTRONICS 1: LOGIC CIRCUITS AND SWITCHING THEORY

TRAINING MODULE TITLE:

BOOLEAN ALGEBRA AND LOGIC GATES

GENERAL OBJECTIVE:

Upon completion this training module, students will be able to conceptualize, analyze, design, and implement fundamental digital logic circuits, applying key principles of logic gates and Boolean algebra.

SPECIFIC OBJECTIVES:

Upon the completion of this training module, the student must be able to:

1. Define and identify important logic switching circuit theories and terminologies
2. Use Boolean Algebra in simplifying logic circuits and solving related problems
3. Implement logic circuits using the Logic Board Training Kit

TOPIC OUTLINE:

1. Logic Gates Fundamentals
2. Boolean Algebra Fundamentals

LABORATORY EQUIPMENT:

SMART LoBo TRAINING KIT: HANDS-ON IOT-BASED LOGIC BOARD TRAINING KIT

Republic of the Philippines
UNIVERSITY OF RIZAL SYSTEM
MORONG CAMPUS

COLLEGE OF ENGINEERING
ELECTRONICS ENGINEERING

ECE 6

Digital Electonics 1:
Logic Circuits and Switching Theory

CHAPTER 1:
Logic Gates Fundamentals

CHAPTER 1: LOGIC GATES FUNDAMENTALS

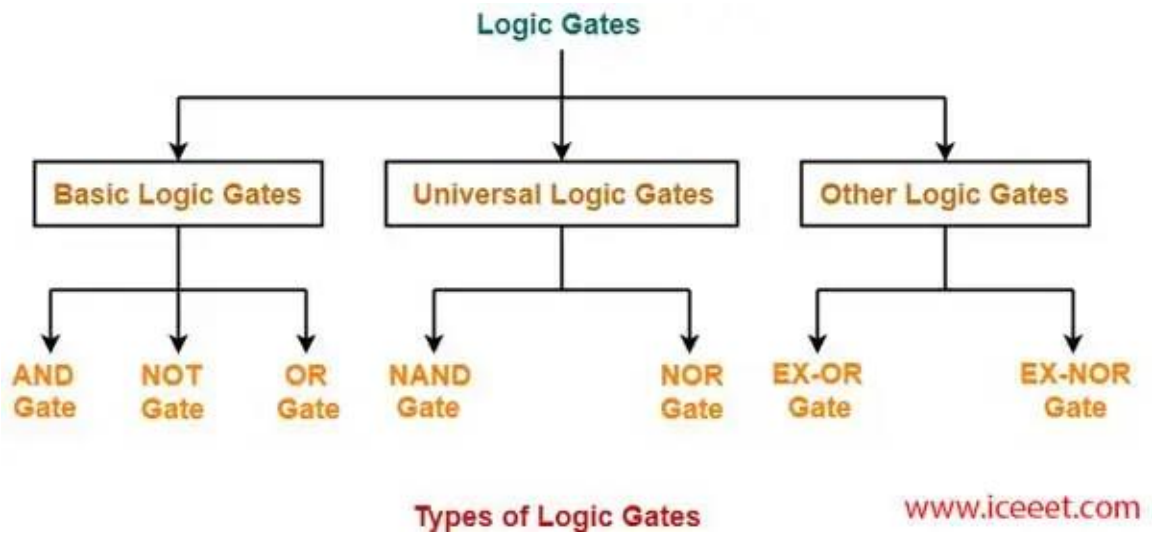
Learning Objectives:

At the end of this lesson, the students are expected to:

1. Know the Logic Gates and its Symbols
2. Know the equivalent truth table of logic gates
3. Understand Logic Gates and its Operation
4. Familiarize with Logic Gates Integrated Circuits (ICs)

INTRODUCTION:

Logic gates perform basic logical functions and are the **fundamental building blocks of digital integrated circuits**. It is an **electronic circuit** having one or more than one input and only one output. Most logic gates take an input of two binary values, and output a single value of a 1 or 0. Some circuits may have only a few logic gates, while others, such as microprocessors, may have millions of them.



Representation of Logic gates

There are three different, but equally powerful, notational methods for describing the behavior of gates and circuits.

1. **Logic Diagram**: A graphical representation of a circuit. Each type of gate is represented by a specific graphical symbol.

2. **Boolean Expression**: Expressions in this algebraic notation are elegant and powerful way to demonstrate the activity of electrical circuits.
3. **Truth Table**: A table that shows all input and output possibilities for a logic circuit.

The total number of possible combinations of binary inputs to a gate is determined by:

$$N = 2^n$$

where N is the number of possible combinations
n is the number of binary inputs

Therefore:
2 bits (n=2) = 4 combinations
3 bits = 8 combinations
4 bits = 16 combinations

1.1 OR GATE

I. Objectives

1. To examine the characteristics and operation of a TTL OR Gate.
2. To construct a logic circuit of an OR Gate with an expanded input using 2-input TTL OR Gates.

II. Equipment and Materials

- Digital Logic Trainer Board
- 4 pcs OR Gate logic blocks (7432 IC, a quad 2-input OR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The logic operation performed by the OR gate is logic addition, also known as the OR operation or function. The OR gate has two or more inputs and a single output. Table 1 shows the ANSI & IEC logic symbols for an OR gate with two input variables A and B. The output of this gate is expressed in Boolean algebra notation as A+B. This notation is read as “A or B”, not as “A plus B”.

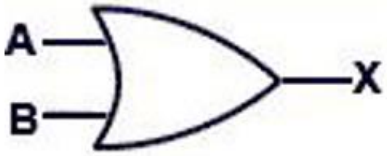
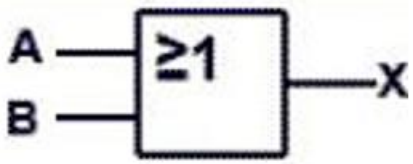
ANSI Symbol	IEC Symbol
	

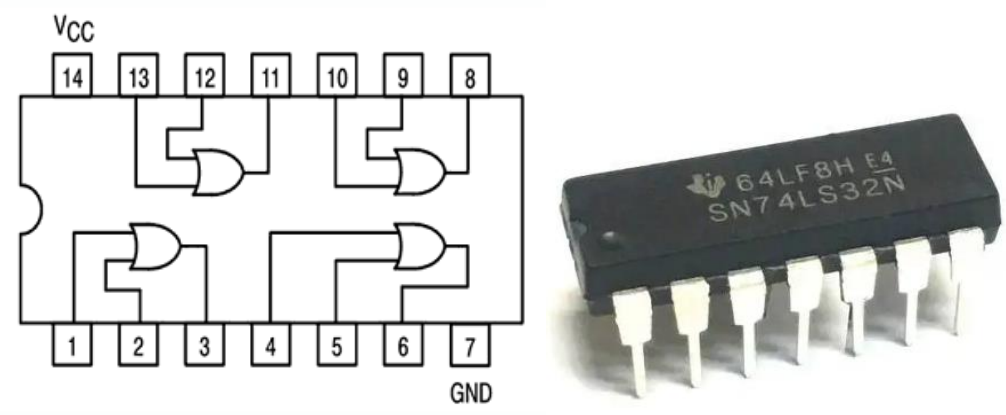
Table 1. Logic symbols for a two-input OR gate

For a 2-input OR gate:

- Output (X) is HIGH (1) when either A or B is HIGH, or when both A and B are HIGH.
- Output (X) is LOW (0) only when both A and B are LOW.

The OR gate determines when certain conditions are met simultaneously. The OR gate is an any-or-all gate because a 1 at any input gives rise to a 1 at the output. The HIGH (1) level is the prime or active output level for the OR Gate.

7432 – OR (four gates per chip)



IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the OR gate logic block (7432 IC) as shown in Figure A.

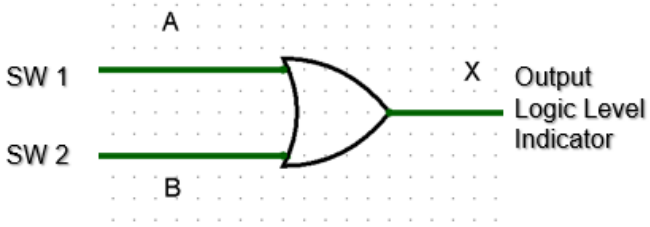


Figure A

3. Set the OR gate to each input combination shown in truth table A.1, observe its output, and record each in the truth table.
4. Using the same circuit shown in Figure A, complete the truth tables A.2 and A.3.
5. Connect the circuit as shown in Figure B.
6. Set the OR gate to the required input level as shown in truth table B, observe its output, and record it in the truth table.

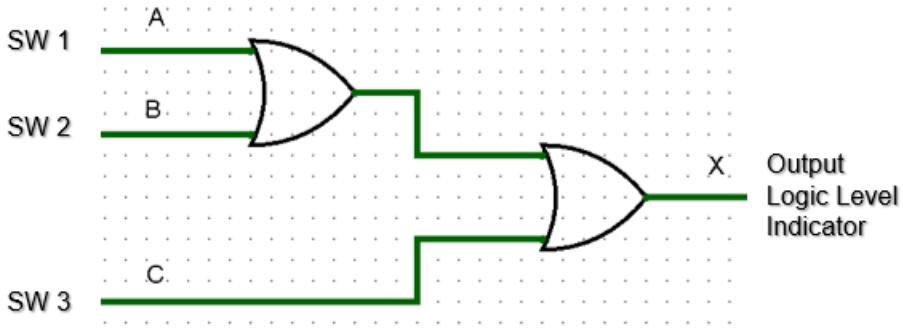


Figure B

ACTIVITY # 1

Name: _____

Title of the Experiment: _____

Truth Table A.1

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table A.2

A	B	X
0	Hang	
1	Hang	

Truth Table A.3

A	B	X
Hang	0	
Hang	1	

Truth Table B

A	B	X
0	0	
0	0	
0	1	
0	1	
1	0	
1	0	
1	1	
1	1	

Questions:

1. What is the equivalent of a floating input of a TTL OR Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

3. What are the two ways that must be done on a three-input OR Gate for it to function as a two-input OR Gate?

   A. _____

   _____

   _____

   B. _____

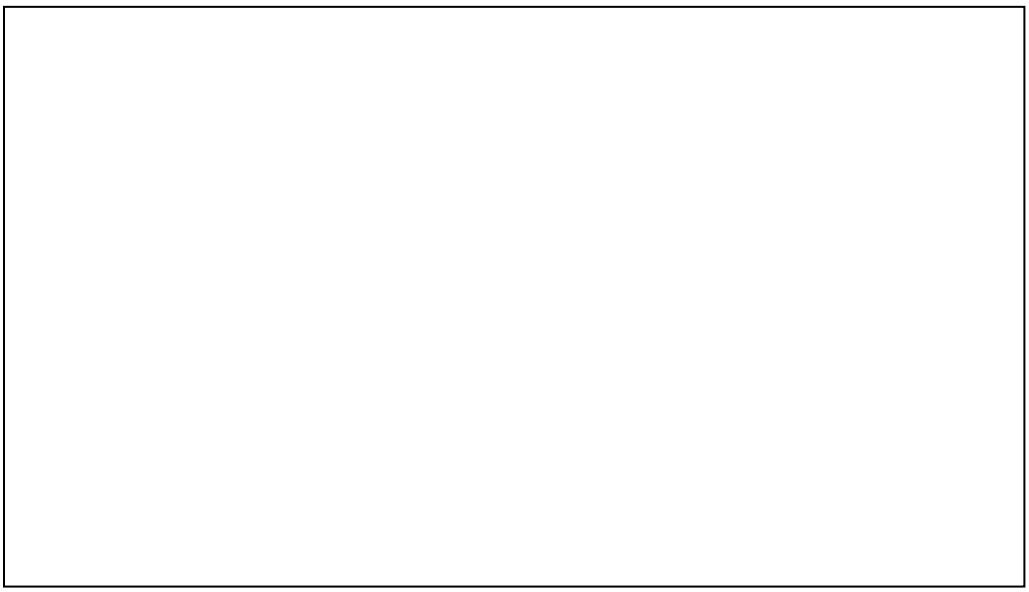
   _____

   _____

4. Draw the two possible connections on how the three two-input OR Gates can be wired to function as a four-input OR Gate. Write also the Boolean expression for each output of the OR Gate.

   Circuit 1

Circuit 2



5. What are the possible applications of OR Gates?

_____

_____

Conclusion:

_____

_____

_____

_____

_____

_____

_____

_____

_____

1.2 AND GATE

I. Objectives

1. To examine the characteristics and operation of a TTL AND Gate.
2. To construct a logic circuit of an AND Gate with an expanded input using 2-input TTL AND Gates.

II. Equipment and Materials

- Digital Logic Trainer Board
- 4 pcs AND Gate logic blocks (7408 IC, a quad 2-input AND Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The basic operation of logic multiplication, commonly known as the AND operation or function, is performed by the AND gate. This gate is composed of two or ore inputs and a single output. Table 1 shows the ANSI & IEC logic symbols for an AND gate with two input variables A and B. The output of this gate is expressed in Boolean algebra notation as AB or A · B. This notation is read as “A and B”, not as “A times B”.

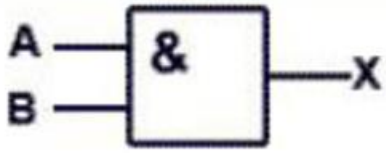
ANSI Symbol	IEC Symbol
	

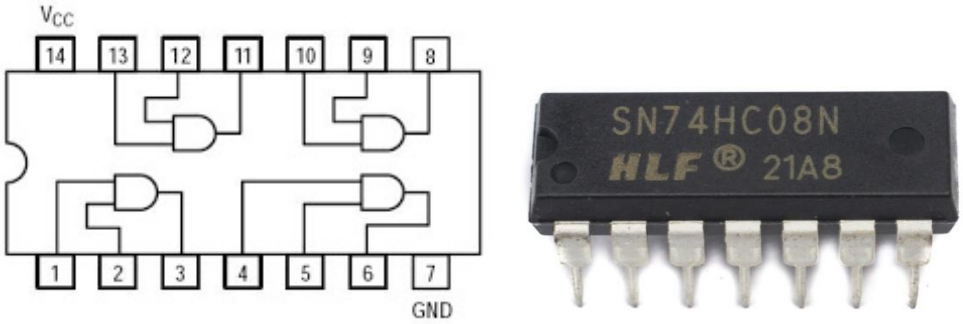
Table 1. Logic symbols for a two-input AND gate

For a 2-input AND Gate:

- Output (X) is HIGH (1) only when inputs A and B are HIGH
- Output (X) is LOW (0) when either A or B, or when both A and B are LOW.

The AND gate determines when certain conditions are simultaneously true. The AND gate is an all-or-nothing gate because there must be a 1 at all of the inputs to obtain a 1 at the output. The HIGH (1) level is the prime or the active output level for the AND gate.

7408- AND (four gates per chip)



IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the AND gate logic block (7408 IC) as shown in Figure A.

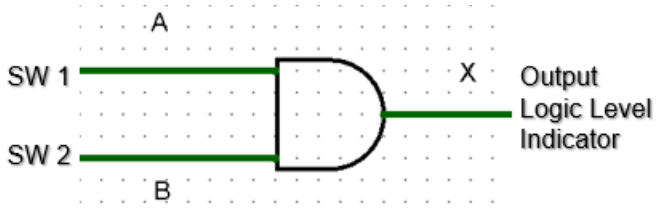


Figure A

3. Set the AND gate to each input combination shown in truth table A.1, observe its output, and record each in the truth table.
4. Using the same circuit shown in Figure A, complete the truth tables A.2 and A.3.
5. Connect the circuit as shown in Figure B.
6. Set the AND gate to the required input level as shown in truth table B, observe its output, and record it in the truth table.

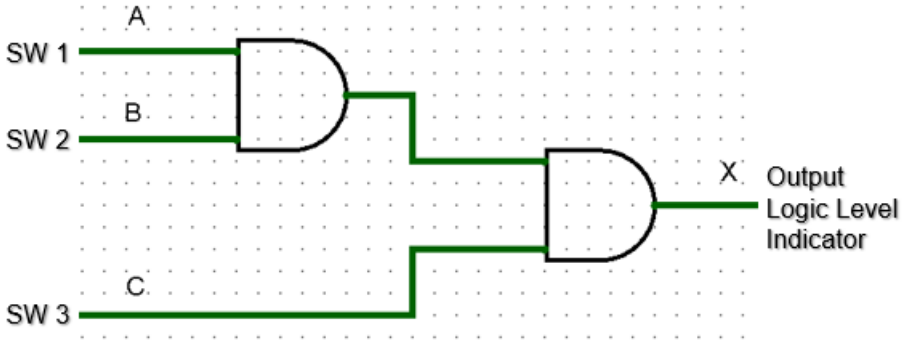


Figure B

ACTIVITY # 2

Name: _____

Title of the Experiment: _____

Truth Table A.1

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table A.2

A	B	X
0	Hang	
1	Hang	

Truth Table A.3

A	B	X
Hang	0	
Hang	1	

Truth Table B

A	B	X
0	0	
0	0	
0	1	
0	1	
1	0	
1	0	
1	1	
1	1	

Questions:

1. What is the equivalent of a floating input of a TTL AND Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

3. What are the two ways that must be done on a three-input AND Gate for it to function as a two-input AND Gate?

   A. _____

   _____

   _____

   B. _____

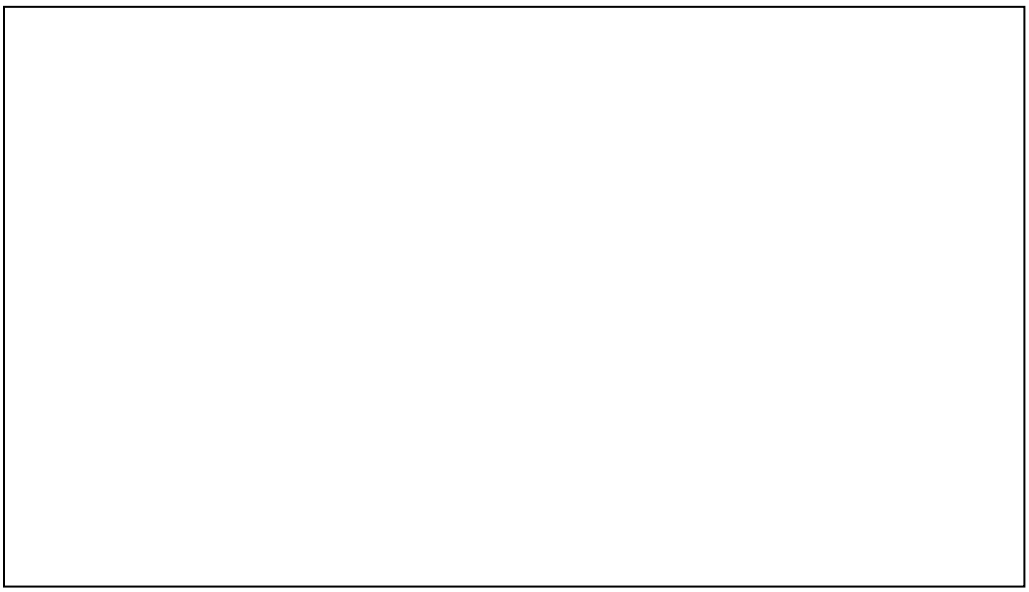
   _____

   _____

4. Draw the two possible connections on how the three two-input AND Gates can be wired to function as a four-input AND Gate. Write also the Boolean expression for each output of the AND Gate.

   Circuit 1

Circuit 2



5. What are the possible applications of AND Gates?

_____

_____

Conclusion:

_____

_____

_____

_____

_____

_____

_____

_____

_____

1.3 NOT GATE

I. Objective

1. To examine the characteristics and operation of a TTL NOT Gate.

II. Equipment and Materials

- Digital Logic Trainer Board
- 3 pcs NOT Gate logic blocks (7404 IC, hex inverter)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The NOT gate (commonly called the Inverter) performs the operation called ***inversion*** or ***complementation***.

The NOT gate has a single input and its logic symbols are shown in Table 1. The output is the complement of the input. If A is the input of the NOT gate, its output will be Ā or A'. The small circle or bubble, at the end of the output of the NOT gate is the standard symbol representing inversion or complementation.

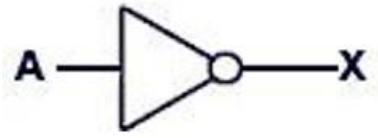
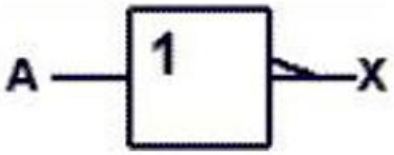
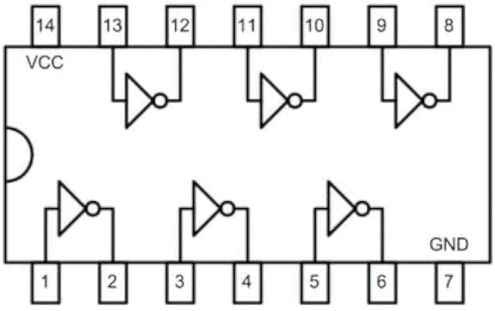
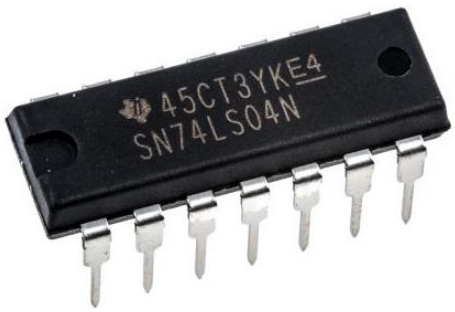
ANSI Symbol	IEC Symbol
	

Table 1. Logic symbols for a NOT gate

7404 – NOT (six gates per chip)



7404 Hex Inverter



IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the NOT gate logic block (7404 IC) as shown in Figure A.

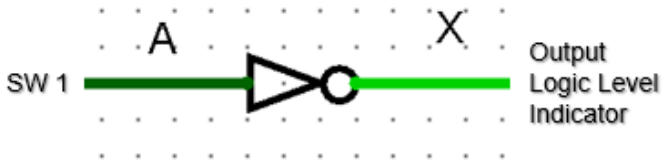


Figure A

3. Complete the truth table A of the NOT gate by observing and recording the output for each input state and condition given in the table.
4. Connect the circuit as shown in Figure B.

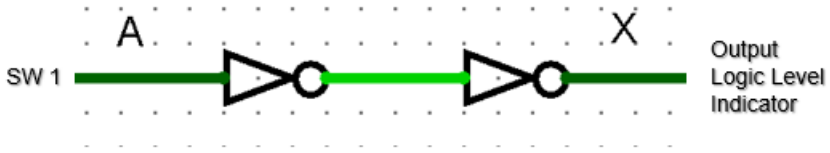


Figure B

5. Complete the truth table B of the two NOT gates connected in series.
6. Connect the circuit as shown in Figure C.

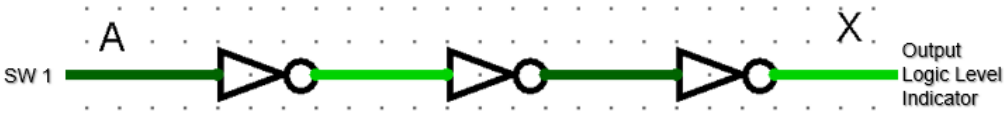


Figure C

7. Complete the truth table C of the three NOT gates connected in series.

ACTIVITY # 3

Name: _____

Title of the Experiment: _____

Truth Table A

A	X
0	
1	
hang	

Truth Table B

A	X
0	
1	

Truth Table C

A	X
0	
1	

Questions:

1. What is the equivalent of a floating input of a TTL NOT Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

   Its simplified expression is _____

3. What is the Boolean expression of the circuit shown in Figure B?

   _____

   Its simplified expression is _____

4. What are the possible applications of NOT Gates?

   _____

Conclusion:

_____

_____

_____

_____

1.4 NOR GATE

I. Objectives

1. To examine the characteristics and operation of a TTL NOR Gate.
2. To implement NOR function using combination of OR and NOT Gates.

II. Equipment and Materials

- Digital Logic Trainer Board
- 1 pc NOR Gate logic block (7402 IC, a quad 2-input NOR Gate)
- 1 pc NOT Gate logic block (7404 IC, a hex inverter)
- 1 pc OR Gate logic block (7432 IC, a quad 2-input OR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The logic symbols for the NOR gate are shown in Table 1. It consists of the OR symbol with a small circle or bubble added to the output. NOR is a contraction of NOT OR. As shown in Figure 1, NOR gate is equivalent to an OR gate followed by a NOT or INVERTER gate. The Boolean expression for the NOR function consists of the OR expression with a bar drawn over the entire expression. The Boolean expression for NOR function becomes $\overline{A+B}$, it can also be written as (A+B)'. NOR gate has an output of 0 when any or all the inputs are 1, and an output of 1 only when all the inputs are 0.

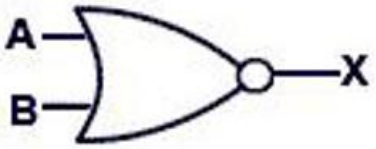
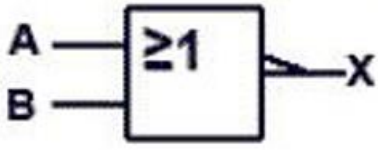
ANSI Symbol	IEC Symbol
	

Table 1. Logic symbols for a two-input NOR gate

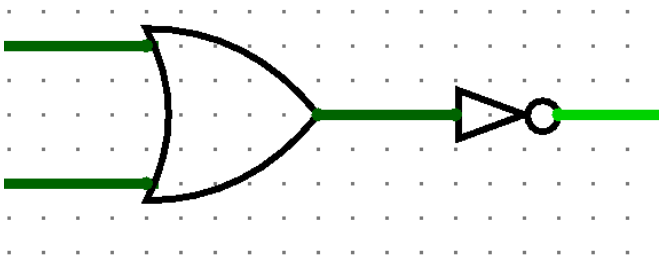
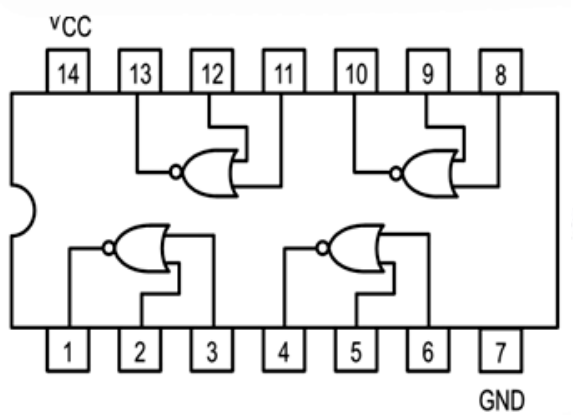


Figure 1. Combinations of logic gates required to implement NOR function

7402 – NOR (four gates per chip)



IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the NOR gate logic block (7402 IC) as shown in Figure A.

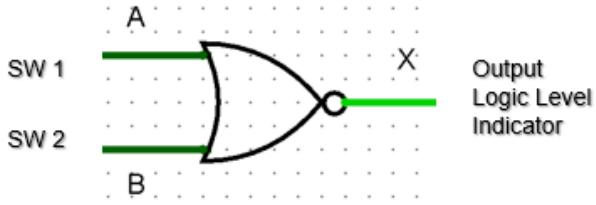


Figure A

3. Set the NOR gate to each input combination shown in truth table A.1, observe its output, and record each in the truth table.
4. Using the same circuit as shown in Figure A, complete truth tables A.2 and A.3.
5. Connect the circuit as shown in Figure B.
6. Set the circuit to each input combination shown in truth table B, observe its output, and record each in the truth table.

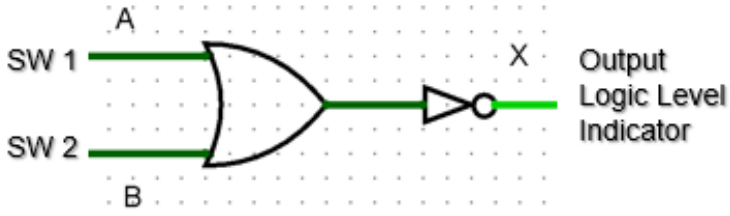


Figure B

ACTIVITY # 4.1

Name: _____

Title of the Experiment: _____

Truth Table A.1

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table A.2

A	B	X
0	Hang	
1	Hang	

Truth Table A.3

A	B	X
Hang	0	
Hang	1	

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

Questions:

1. What is the equivalent of a floating input of a TTL NOR Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

3. What are the two ways that must be done on a three-input NOR Gate for it to function as a two-input NOR Gate?

   A. _____

   _____

   _____

   B. _____

   _____

   _____

4. What are the possible applications of NOR Gates?

   _____

   _____

Conclusion:

_____

_____

_____

_____

_____

1.4.1 NOR GATE AS A UNIVERSAL GATE

I. Objectives

1. To implement different gate operations using NOR gate/s.

II. Equipment and Materials

- Digital Logic Trainer Board
- 2 pcs. NOR Gate logic blocks (7402 IC, a quad 2-input NOR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The NOR gate is also called a **universal gate** because it can be connected to other NOR gate/s to generate any logic function.

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the circuit as shown in Figure A.

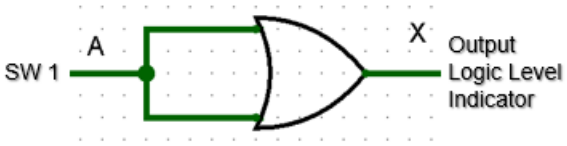


Figure A

3. Set the NOR gate to the required input level as shown in truth table A, observe its output, and record it in the truth table.
4. Identify the equivalent logic function of the circuit.
5. Connect circuit as shown in Figure B.

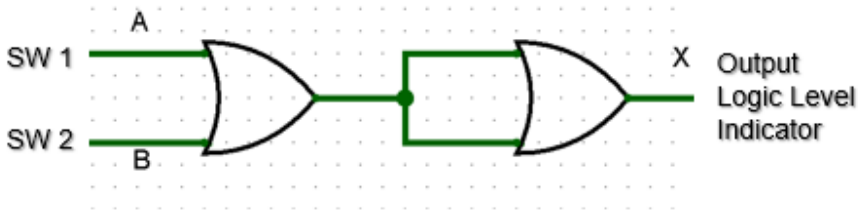


Figure B

6. Set the experimental circuit to each input combination as shown in table B, observe its output, and record it in the truth table.
7. Identify the equivalent logic function of the circuit.

8. Using two-input NOR gates, design a logic circuit that is equivalent to a two-input AND gate. Verify its operation by completing truth table C.
9. Using two-input NOR gates, design a logic circuit that is equivalent to a two-input NAND gate. Verify its operation by completing truth table D.
10. Using two-input NOR gates, design a logic circuit that is equivalent to a two-input XOR gate. Verify its operation by completing truth table E.
11. Using two-input NOR gates, design a logic circuit that is equivalent to a two-input XNOR gate. Verify its operation by completing truth table F.

ACTIVITY # 4.2

Name: _____

Title of the Experiment: _____

Truth Table A

A	X
0	
0	

What is the equivalent logic function of the circuit shown in Figure A?

_____

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

What is the equivalent logic function of the circuit shown in Figure B?

_____

Truth Table C

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input AND gate using two-input NOR gates

Truth Table D

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input NAND gate using two-input NOR gates

Truth Table E

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input XOR gate using two-input NOR gates

Truth Table F

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input XNOR gate using two-input NOR gates

1.5 NAND GATE

I. Objectives

1. To examine the characteristics and operation of a TTL NAND Gate.
2. To implement NAND function using combination of AND and NOT Gates.

II. Equipment and Materials

- Digital Logic Trainer Board
- 1 pc NAND Gate logic block (7400 IC, a quad 2-input NAND Gate)
- 1 pc NOT Gate logic block (7404 IC, a hex inverter)
- 1 pc AND Gate logic block (7408 IC, a quad 2-input AND Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The standard (ANSI) logic symbol for a 2-input NAND gate is shown in Table 1. The symbol consists of am AND gate followed by a small circle or bubble to indicate complementation. NAND is a contraction of NOT AND, and the Boolean expression for its output is expressed as $\overline{A \cdot B}$ or $\overline{AB}$, it can also be written as (A·B)' or (AB)'.

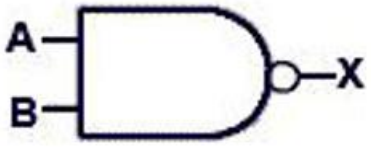
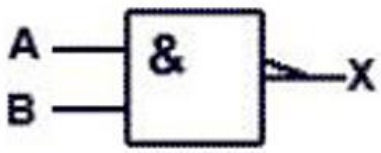
ANSI Symbol	IEC Symbol
	

Table 1. Logic symbols for a two-input NAND gate

An AND gate followed by a NOT or INVERTER gate, as shown in Figure 1, is the equivalent logic circuit of a NAND gate. NAND gate may have more than two inputs. A NAND gate has an output of 1 when any or all the inputs are 0, and an output of 0 when all the inputs are 1.

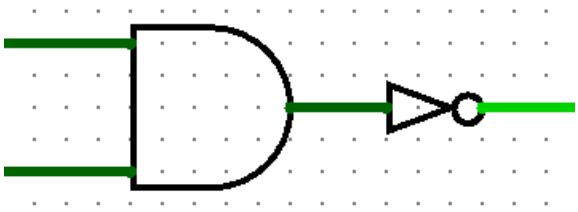
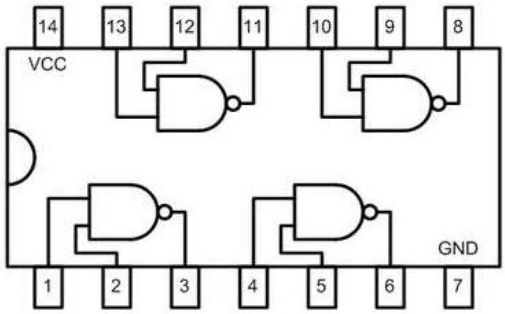


Figure 1. Combinations of Logic gates required to implement NAND function

7400 – NAND (four gates per chip)



7400 Quad 2 Input NAND

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the NAND gate logic block (7400 IC) as shown in Figure A.

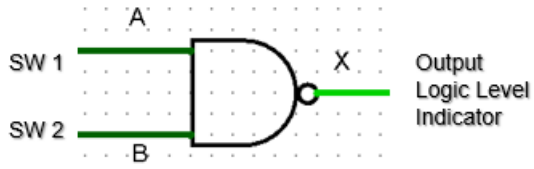


Figure A

3. Set the NAND gate to each input combination shown in truth table A.1, observe its output, and record each in the truth table.
4. Using the same circuit as shown in Figure A, complete truth tables A.2 and A.3.
5. Connect the circuit as shown in Figure B.
6. Set the circuit to each input combination shown in truth table B, observe its output, and record each in the truth table.

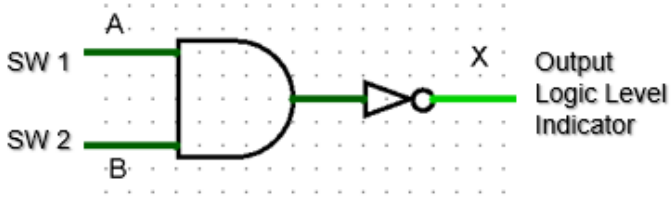


Figure B

ACTIVITY # 5.1

Name: _____

Title of the Experiment: _____

Truth Table A.1

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table A.2

A	B	X
0	Hang	
1	Hang	

Truth Table A.3

A	B	X
Hang	0	
Hang	1	

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

Questions:

1. What is the equivalent of a floating input of a TTL NAND Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

3. What are the two ways that must be done on a three-input NAND Gate for it to function as a two-input NAND Gate?

   A. _____

   _____

   _____

   B. _____

   _____

   _____

4. What are the possible applications of NAND Gates?

   _____

   _____

Conclusion:

_____

_____

_____

_____

_____

1.4.2 NAND GATE AS A UNIVERSAL GATE

I. Objectives

1. To implement different gate operations using NAND gate/s.

II. Equipment and Materials

- Digital Logic Trainer Board
- 2 pcs. NAND Gate logic blocks (7400 IC, a quad 2-input NAND Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The NAND gate is sometimes called a **universal gate** because it can be connected to other NAND gate/s to generate any logic function.

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the circuit as shown in Figure A.

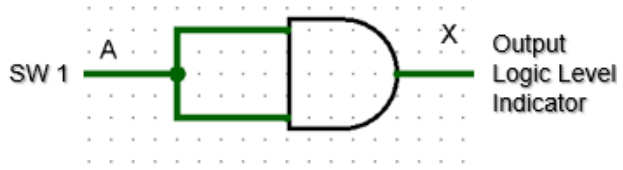


Figure A

3. Set the NAND gate to the required input level as shown in truth table A, observe its output, and record it in the truth table.
4. Identify the equivalent logic function of the circuit.
5. Connect circuit as shown in Figure B.

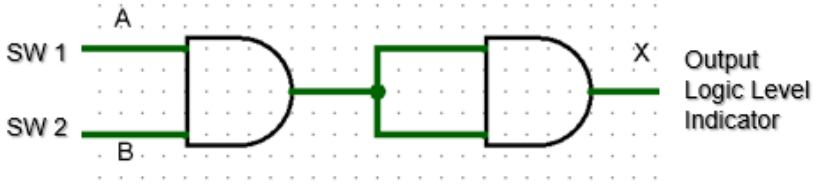


Figure B

6. Set the experimental circuit to each input combination as shown in table B, observe its output, and record it in the truth table.
7. Identify the equivalent logic function of the circuit.

8. Using two-input NAND gates, design a logic circuit that is equivalent to a two-input OR gate. Verify its operation by completing truth table C.
9. Using two-input NAND gates, design a logic circuit that is equivalent to a two-input NOR gate. Verify its operation by completing truth table D.
10. Using two-input NAND gates, design a logic circuit that is equivalent to a two-input XOR gate. Verify its operation by completing truth table E.
11. Using two-input NAND gates, design a logic circuit that is equivalent to a two-input XNOR gate. Verify its operation by completing truth table F.

ACTIVITY # 5.2

Name: _____

Title of the Experiment: _____

Truth Table A

A	X
0	
0	

What is the equivalent logic function of the circuit shown in Figure A?

_____

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

What is the equivalent logic function of the circuit shown in Figure B?

_____

Truth Table C

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input OR

gate using two-input NAND gates

Truth Table D

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input NOR gate using two-input NOR gates

Truth Table E

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input XOR gate using two-input NOR gates

Truth Table F

A	B	X
0	0	
0	1	
1	0	
1	1	

Logic circuit of a two-input XNOR gate using two-input NOR gates

1.6 EXCLUSIVE-OR (XOR) GATE

I. Objectives

1. To examine the characteristics and operation of a TTL XOR Gate.
2. To implement XOR function using combinations of logic gates.

II. Equipment and Materials

- Digital Logic Trainer Board
- 2 pcs. NOT Gate logic blocks (7404 IC, a hex inverter)
- 2 pcs. AND Gate logic blocks (7408 IC, a quad 2-input AND Gate)
- 1 pc. OR Gate logic block (7432 IC, a quad 2-input OR Gate)
- 1 pc XOR Gate logic block (7486 IC, a quad 2-input XOR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

The normal OR operation applied to two-input variables A and B results in an output equal to 1 when either A=1 or B=1, or when both A=1 and B=1. A very useful Boolean operation, the EXCLUSIVE-OR operation, excludes the case when A=B=1. The output of the EXCLUSIVE-OR operation is 1 when A=1 or B=1, but not when both inputs are equal to 1. The EXCLUSIVE-OR operation is used in digital logic systems to decide if two binary numbers are equal or not equal.

The logic expression for this function is $X = A\bar{B} + \bar{A}B$. The symbol for the EXCLUSIVE-OR operation is $\oplus$. The given equation can be written as follows:

$$X = A\bar{B} + \bar{A}B = A \oplus B$$

One of the methods of implementing the EXCLUSIVE-OR function is to use two AND gates, two INVERTERS, and one OR gate as shown in Figure 1. Table 1 shows the logic symbol for a single gate which can implement this function.

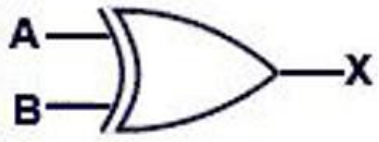
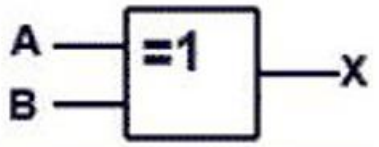
ANSI Symbol	IEC Symbol
	

Table 1. Logic symbols for a two-input NAND gate

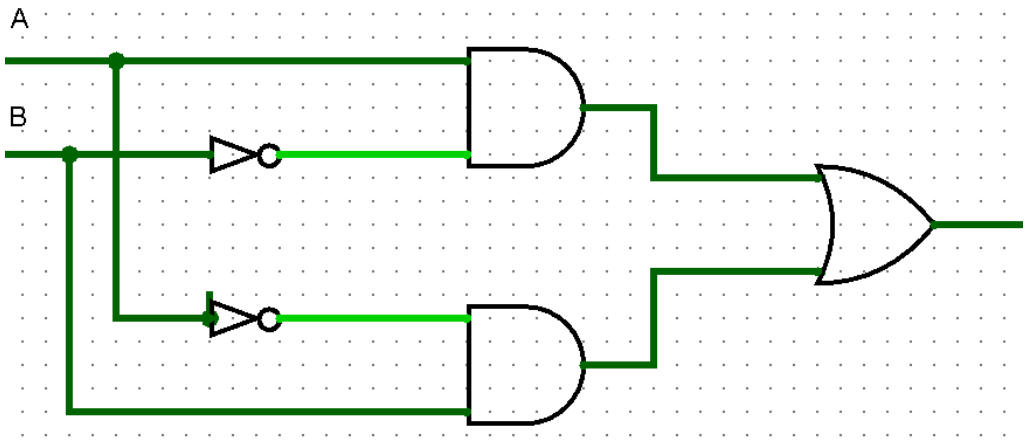
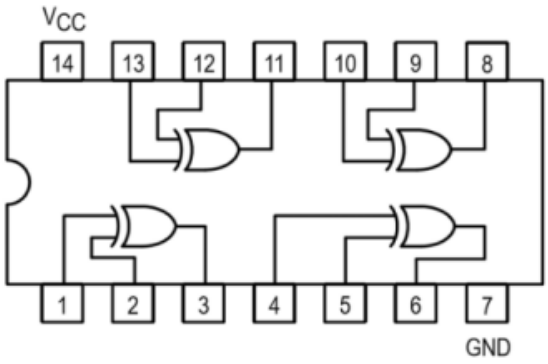
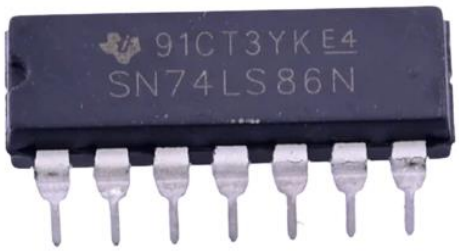


Figure 1. Implementation of the EX-OR function

7486 – XOR (four gates per chip)





IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the XOR gate logic block (7486 IC) as shown in Figure A.

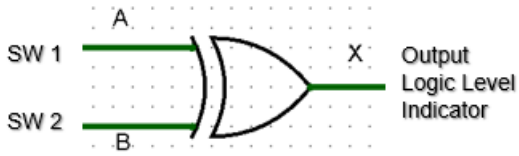


Figure A

3. Set the circuit to each input combination shown in truth table A.1, observe its output, and record each in the truth table.

4. Using the same circuit shown in Figure A, complete the truth tables A.2 and A.3.
5. Connect the circuit as shown in Figure B.

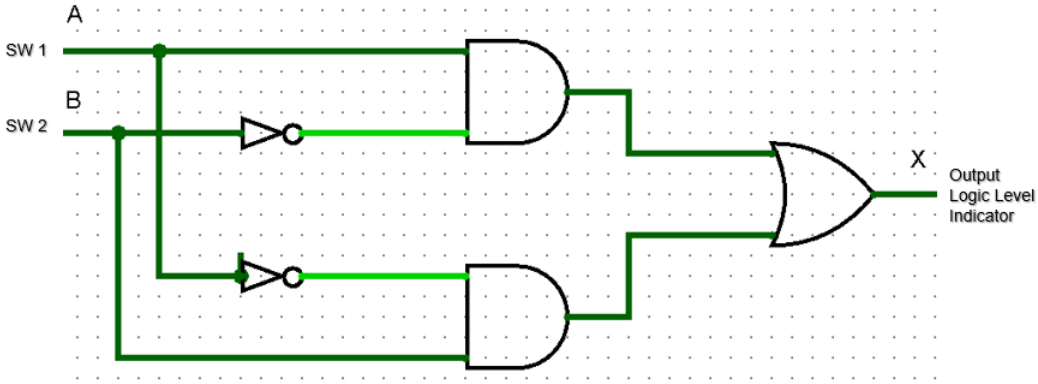


Figure B

6. Set the circuit to each input combination as shown in truth table B, observe its output, and record it in the truth table.

ACTIVITY # 6

Name: _____

Title of the Experiment: _____

Truth Table A.1

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table A.2

A	B	X
0	Hang	
1	Hang	

Truth Table A.3

A	B	X
Hang	0	
Hang	1	

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

Questions:

1. What is the equivalent of a floating input of a TTL XOR Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

3. What must be done on a three-input XOR Gate for it to function as a two-input XOR Gate?

   _____

   _____

   _____

4. What are the possible applications of XOR Gates?

   _____

   _____

Conclusion:

_____

_____

_____

_____

_____

_____

_____

1.7 EXCLUSIVE-NOR (XNOR) GATE

I. Objectives

1. To examine the characteristics and operation of a TTL XNOR Gate.
2. To implement XNOR function using combinations of logic gates.

II. Equipment and Materials

- Digital Logic Trainer Board
- 1 pc. NOR Gate logic block (7402 IC, a quad 2-input NOR Gate)
- 2 pcs. NOT Gate logic blocks (7404 IC, a hex inverter)
- 2 pcs. AND Gate logic blocks (7408 IC, a quad 2-input AND Gate)
- 1 pc XNOR Gate logic block (4077 IC, a quad 2-input XNOR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

When an INVERTER or NOT gate is added to the EXCLUSIVE-OR circuit, the result is an EXCLUSIVE-NOR function. A method of implementing the EXCLUSIVE-NOR function is to use two NOT gates, two AND gates, and a NOR gate as shown in Figure 1. The logic symbols for a single gate which can implement the EXCLUSIVE-NOR function is shown in Table 1. The logic expression for the EXCLUSIVE-NOR operation is

$$X = \overline{A\bar{B} + \bar{A}B} = \overline{A \oplus B}$$

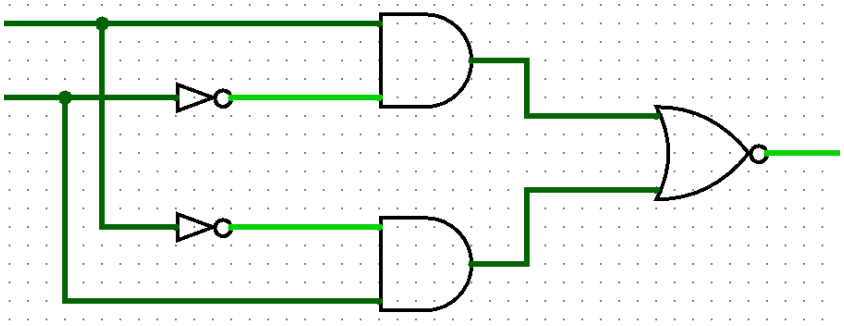


Figure 1. Implementation of the EX-NOR function

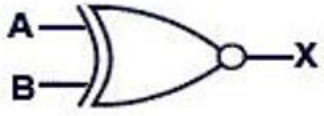
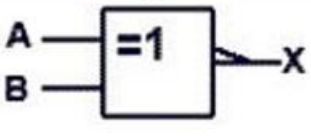
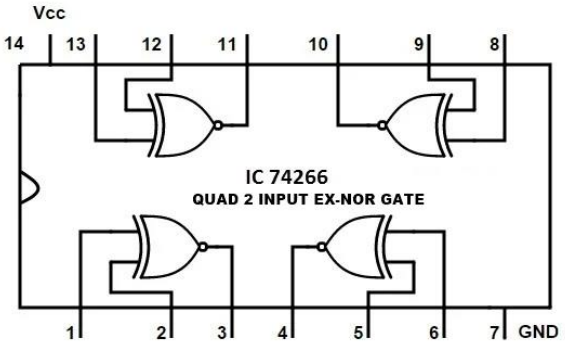
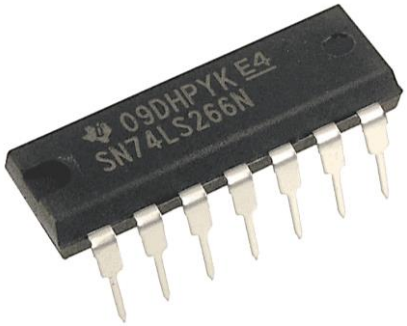
ANSI Symbol	IEC Symbol
	

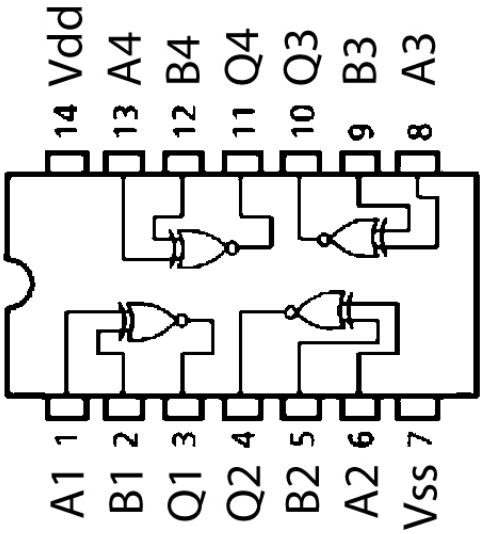
Table 1. Logic symbols for a two-input XNOR gate

74266 – XNOR (four gates per chip)





4077 – XNOR (four gates per chip)



IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the XNOR gate logic block (4077 IC) as shown in Figure A.

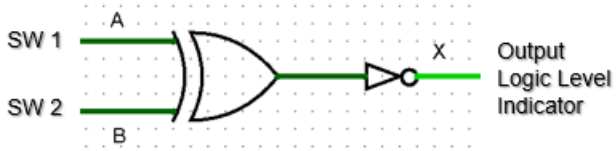


Figure A

3. Set the circuit to each input combination shown in truth table A.1, observe its output, and record each in the truth table.
4. Using the same circuit shown in Figure A, complete the truth tables A.2 and A.3.
5. Connect the circuit as shown in Figure B.

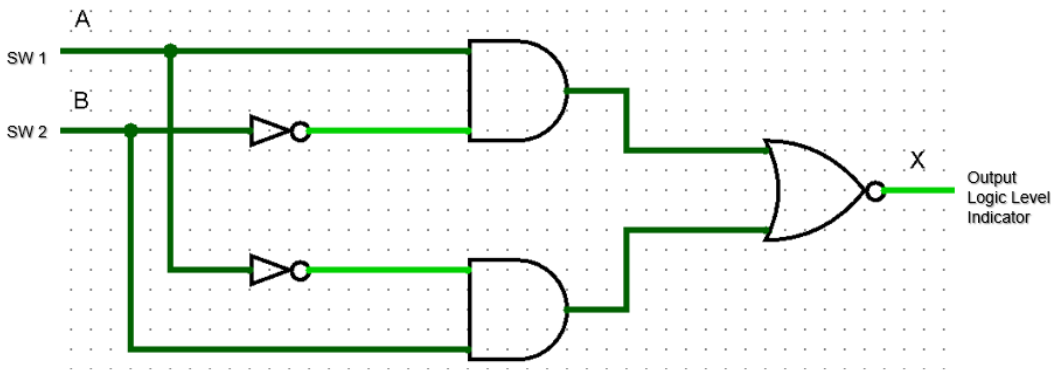


Figure B

6. Set the circuit to each input combination as shown in truth table B, observe its output, and record it in the truth table.

ACTIVITY # 7

Name: _____

Title of the Experiment: _____

Truth Table A.1

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table A.2

A	B	X
0	Hang	
1	Hang	

Truth Table A.3

A	B	X
Hang	0	
Hang	1	

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

Questions:

1. What is the equivalent of a floating input of a TTL XNOR Gate?

   _____

   _____

2. What is the Boolean expression of the circuit shown in Figure B?

   _____

3. What must be done on a three-input XNOR Gate for it to function as a two-input XNOR Gate?

   _____

   _____

   _____

4. What are the possible applications of XNOR Gates?

   _____

   _____

Conclusion:

_____

_____

_____

_____

_____

_____

_____

1.8 COMBINATIONAL CIRCUITS

I. Objective

1. Design and implement combinational logic circuits

II. Equipment and Materials

- Digital Logic Trainer Board
- 4 pcs. AND Gate logic blocks (7408 IC, a quad 2-input AND Gate)
- 3 pcs. NOT Gate logic blocks (7404 IC, a hex inverter)
- 2 pcs. OR Gate logic block (7432 IC, a quad 2-input OR Gate)
- 1 pc XOR Gate logic block (7486 IC, a quad 2-input XOR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

Both the sequential and combinational circuits are the most widely used ones in the arena of digital electronics. They constitute two broad categories of circuits.

The **Combinational Circuits** are memoryless digital logic circuits whose output at any instant in time depends only on the combination of its inputs. Unlike Sequential Logic Circuits whose outputs are dependent on both their present inputs and their previous output state giving them some form of Memory. The outputs of Combinational Logic Circuits are only determined by the logical function of their current input state, logic “0” or logic “1”, at any given instant in time.

The result is that combinational logic circuits have no feedback, and any changes to the signals being applied to their inputs will immediately have an effect at the output.

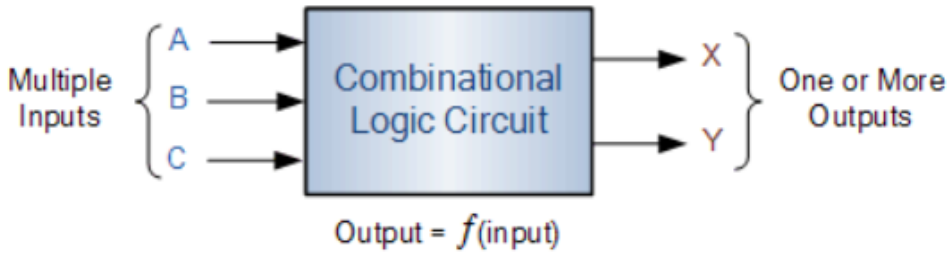


Figure 1. Block Diagram of a Combinational Logic

Combinational Logic Circuits are made up from basic and universal logic such as AND, OR, NOT, NAND, and NOR gates that are “combined” or connected together to produce more complicated

switching circuits. These logic gates are the building blocks of combinational logic circuits.

Example:

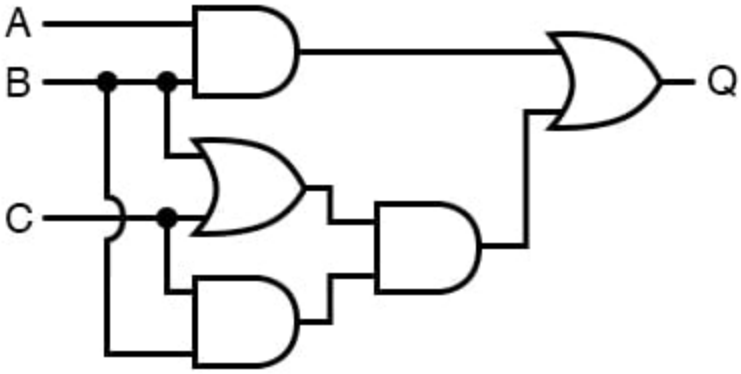


Figure 2. Logic Diagram of Combinational Circuits

The Boolean expression for this circuit is AB + (B+C) · (BC).

A practical application and example of a combinational circuit is a **decoder**, which converts the binary code data present at its input into a number of different output lines, one at a time producing an equivalent decimal code at its output.

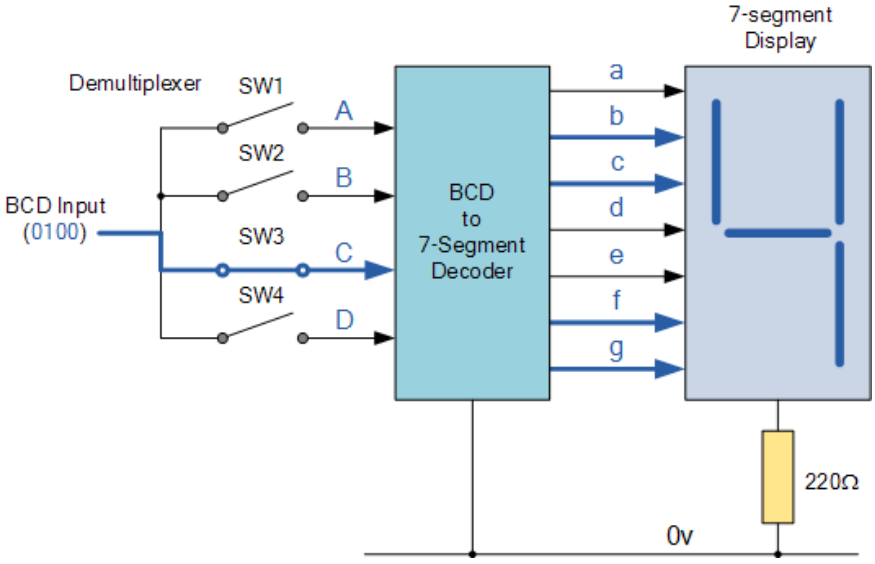


Figure 3. Pinout Diagram of BCD to 7-segment Decoder

Image from: https://www.electronics-tutorials.ws/combination/comb_6.html

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.

2. Analyze the different combinational logic circuits which are given by the following Boolean expressions:
   A. ABD' + B'D' = X
   B. (A'B') + (E' · (AB)) = X
   C. (A'D) ⊕ (AB') = X

3. For each given circuit, draw its logic diagram, connect the circuit, and complete the truth tables by observing and recording the output for each input state and condition given in the tables.

ACTIVITY # 8

Name: _____

Title of the Experiment: _____

Boolean Expression: ABD' + B'D' = X

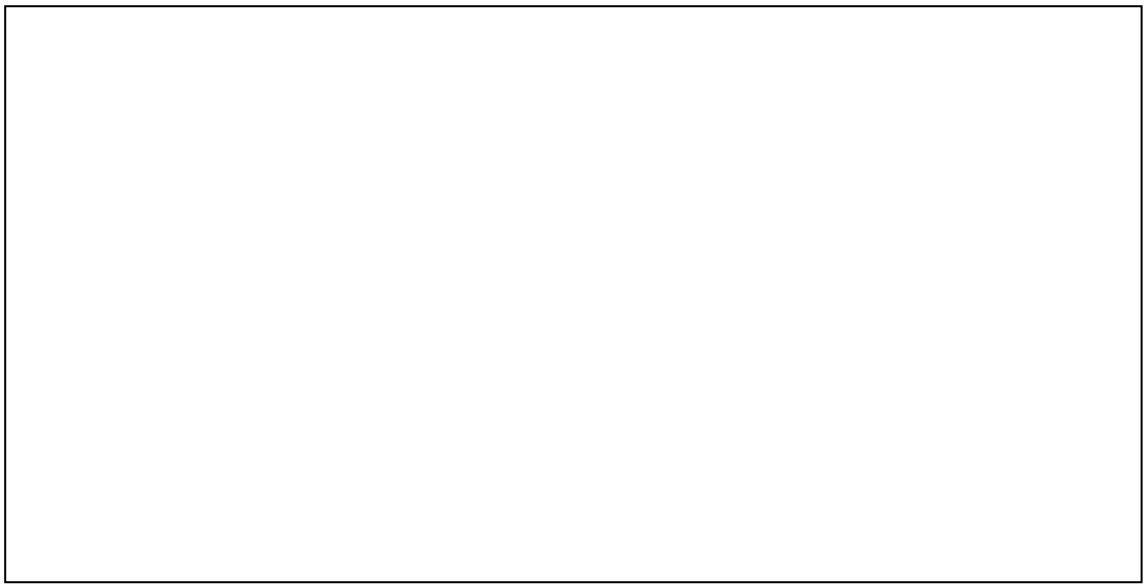
The logic circuit:

Truth Table A

A	B	D	X
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Boolean Expression: (A’B’) + (E’ · (AB)) = X

The logic circuit:



Truth Table B

A	**B**	**E**	**X**
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Boolean Expression: (A'D) $\oplus$ (AB') = X

The logic circuit:

Truth Table C:

A	**B**	**D**	**X**
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Questions:

1. What is the primary difference between a combinational circuit and a sequential circuit?

   _____

   _____

   _____

2. What is the output of the circuit A if A = D = 1; B = 0?

   _____

3. How about in circuit B, what is the output if A = 1; B = E = 0?

   _____

4. Give some applications of combinational logic circuits in digital systems.

   _____

   _____

Conclusion:

_____

_____

_____

_____

_____

_____

_____

Republic of the Philippines
UNIVERSITY OF RIZAL SYSTEM
MORONG CAMPUS

COLLEGE OF ENGINEERING
ELECTRONICS ENGINEERING

ECE 6

Digital Electonics 1:
Logic Circuits and Switching Theory

CHAPTER 2:
Boolean Algebra Fundamentals

CHAPTER 2: BOOLEAN ALGEBRA FUNDAMENTALS

Learning Objectives:

At the end of this lesson, the students are expected to:

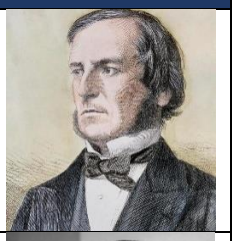
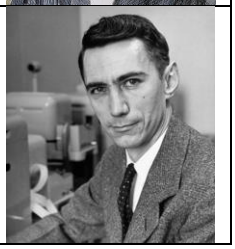
1. Know the brief history of Boolean Algebra
2. Know the basic operations in Boolean Algebra
3. Different application of Boolean Algebra
4. Apply the Principle of Boolean Algebra to Manipulate and Minimize Logic Expressions

INTRODUCTION:

Boolean Algebra is the mathematics use to analyze digital gates and circuits. It is use both reduce and simplify a complex Boolean expression in an attempt to reduce the number of logic gates required.

Boolean Algebra is therefore a system of mathematics based on logic that has its own set of rules or laws which are used to define and reduce Boolean expressions.

The basic mathematics needed for the study of the logic design of digital systems is Boolean algebra.

History of Boolean Algebra		
1854	**George Boole** introduced a systematic treatment of logic and developed for this purpose an algebraic system	
1938	**Claude Shannon** introduced a 2-valued Boolean Algebra, in which he demonstrated that the properties of bistable electrical switching circuits can be represented by this algebra.	

Duality Principle

Every algebraic expression deducible from the postulates of Boolean Algebra remains valid if its operators and identity elements are interchanged.

If the dual of an algebraic expression is desired, we simply interchange OR and AND operators and replace 1's by 0's and 0's by 1's.

Applications of Boolean Algebra

Boolean algebra has many other applications including set theory and mathematical logic.

In this course its application is restricted to switching circuits, specifically is logic design simplification. Because all of the switching devices which we will use are essentially two-state devices (such as a transistor with high or low output voltage), we will study the special case known as two-valued Boolean algebra often referred to as switching algebra.

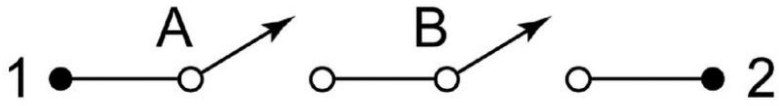
Switches

Now, applying switching theory, we will label each switch with a variable. If switch X is open, then we will define the value of X to be 0; if switch X is closed, then we will define the value of X to be 1.

X = 0 switch open
X = 1 switch closed

Recall in circuit concepts that an open circuit (similar to an open switch) may also be referred to as a LOW state or logic 0. Conversely, a closed circuit (similar to a closed switch) may be referred to as a HIGH state or logic 1.

Now consider a circuit composed of 2 switches in series. We will define the transmission between the terminals as T = 0 if there is an open circuit between the terminals and T = 1 if there is a closed circuit between the terminals.



T = 0 open circuit between terminals 1 and 2
T = 1 closed circuit between terminals 1 and 2

$$T = AB$$

2.1 SIMPLIFICATION OF BOOLEAN EXPRESSION

I. Objective

1. Simplify and implement logic circuit expressions using different rules, laws, and theorems in Boolean Algebra

II. Equipment and Materials

- Digital Logic Trainer Board
- 5 pcs. AND Gate logic blocks (7408 IC, a quad 2-input AND Gate)
- 5 pcs. NOT Gate logic blocks (7404 IC, a hex inverter)
- 5 pcs. OR Gate logic block (7432 IC, a quad 2-input OR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

Boolean algebra is a branch of mathematics that deals with the manipulation of variables which can assume only two truth values, true or false, denoted by 1 and 0, respectively. And there are different Boolean algebra laws used to perform operations on Boolean variables.

BASIC RULE INVOLVING SINGLE VARIABLE

Operations with 0 and 1 (proving can be done by switch Analogy)

1. a) X + 0 = X b) X ⋅ 1 = X
2. a) X + 1 = 1 b) X · 0 = 0

Idempotent Laws

3. a) X + X = X b) X · X = X

Involution Law:

4. a) (X')' = 1 b) $\bar{\bar{X}} = 1$

Complementary Laws

5. a) X + X' = 1 b) X · X = 0

LAWS & RULES INVOLVING MORE THAN ONE VARIABLE

Commutative Laws

6. a) X · Y = Y · X b) X + Y = Y + X

Associative Laws

7. a) (XY) Z = X (YZ) = XYZ
   b) (X + Y) + Z = X + (Y+Z) = X + Y + Z

8. a) X Y Z = 1, if X = Y = Z = 1

b) X + Y + Z = 0, if X = Y = Z = 0

Distributive Laws

9. a) X (Y + Z) = XY + XZ
   b) X + YZ = (X+Y) (X+Z)

Absorption Law

10. a) X (X + Y) = X
    b) X + XY = X

THEOREMS ON BOOLEAN ALGEBRA

De Morgan's Law

11. a) $\overline{AB} = \bar{A} + \bar{B}$
    b) $\overline{A + B} = \bar{A}\,\bar{B}$

Consensus Theorem

12. XY + X'Z + YZ = XY + X'Z

PROBLEMS AND EXAMPLES ON LAWS OF BOOLEAN ALGEBRA:

1. Using Boolean algebra techniques, simplify [AB] + [AB']
   F = A · [B+B'] ; Complementary law X + X' = 1
   F = A · 1 ; operations with 0 and 1 X · 1 = X
   F = A

2. Using Boolean algebra techniques, simplify AB + A(B+C) + B(B+C).
   Solution:
   (The following is not the only approach.)

   Step 1: Apply Distributive Law to the second and last terms in the expression.

   AB + AB + AC + BB + BC

   Step 2: Apply (BB = B) to the fourth term

   AB + AB + AC + B + BC

   Step 3: Apply (AB + AB = AB) to the first two terms

   AB + AC + B

Step 4: Apply (AB + B = B) to the first & third terms

B + AC

Therefore, **AB + A(B+C) + B(B+C)** is reduced to **B + AC**.

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Interpret the different problems which are given by the following Boolean expressions:
    A. [A'CD] + [AC'D'] + [A'C'D'] + [AC'D] + [ACD] = X
    B. [BC'] + [B'D'E'] + [B'C'E] + [B'C'DE'] = X
3. Simplify the output expressions if possible. Use different rules, laws, and theorems in Boolean Algebra
4. Implement the circuit for the final, simplified expression.
5. For each given circuit, draw the logic diagram of the simplified expression, connect the circuit, and complete the truth tables by observing and recording the output for each input state and condition given in the tables.

ACTIVITY # 8

Name: _____

Title of the Experiment: _____

Boolean Expression: [A'CD] + [AC'D'] + [A'C'D'] + [AC'D] + [ACD] = X

Simplified Output Expression: _____

The simplified logic circuit:

Truth Table A

A	C	D	X
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Boolean Expression: [BC’] + [B’D’E’] + [B’C’E] + [B’C’DE’] = X

Simplified Output Expression: _____

The simplified logic circuit:

Truth Table B

B	**C**	**D**	**E**	**X**
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	

1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Questions:

1. What are the rules, laws, and theorems did you use in simplifying the Boolean expression of the given logic circuit?

   _____

   _____

   _____

2. What is the importance of simplifying the logic circuit?

   _____

   _____

   _____

Conclusion:

_____

_____

_____

_____

_____

_____

2.2 DE MORGAN'S FIRST THEOREM

I. Objective

1. To prove experimentally De Morgan's first theorem.

II. Equipment and Materials

- Digital Logic Trainer Board
- 1 pc. NOR Gate logic block (7402 IC, a quad 2-input NOR Gate)
- 1 pc. NOT Gate logic blocks (7404 IC, a hex inverter)
- 1 pc. AND Gate logic blocks (7408 IC, a quad 2-input AND Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

De Morgan's first theorem states that the complement of a sum equals the product of the complements.

The Boolean equation for the Figure 1 is $X = \overline{A + B}$. While the Boolean equation for the Figure 2 is $X = \bar{A}\,\bar{B}$. The first equation describes a NOR gate, and the second equation is a bubbled AND gate. By De Morgan's first theorem, these two equations are equal and can be equated as $\overline{A + B} = \bar{A}\,\bar{B}$. The Figure 3 is the graphical meaning of De Morgan's first theorem.

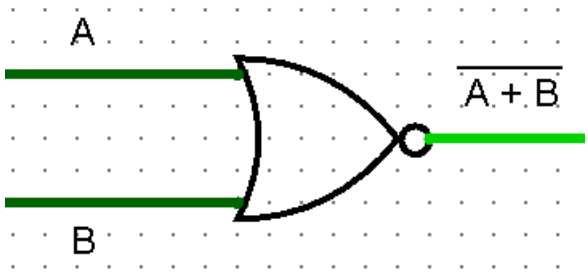
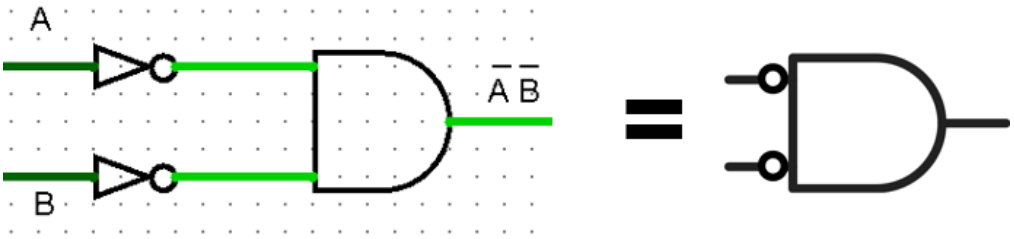


Figure 1. NOR gate



(a) (b)

Figure 2. a. Bubbled AND gate implementation
b. Bubbled AND gate symbol

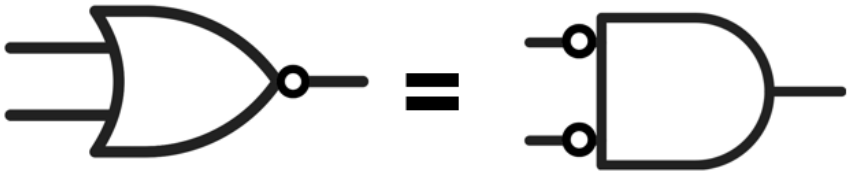


Figure 3. Graphical representation of De Morgan's first theorem

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the circuit as shown in Figure A.

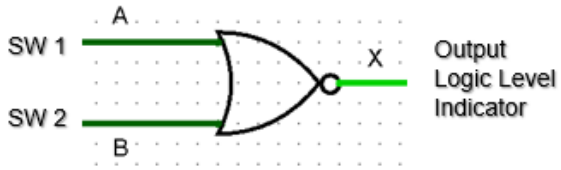


Figure A

3. Set the NOR gate to each input combination shown in truth table A, observe its output, and record each in the truth table.
4. Connect the circuit as shown in Figure B.

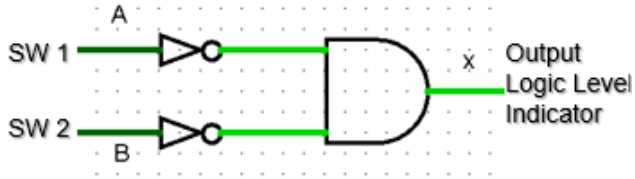


Figure B

5. Set the bubbled AND logic circuit to each input combination as shown in truth table B, observe its output, and record each in the truth table.
6. Connect the circuit as shown in Figure C.

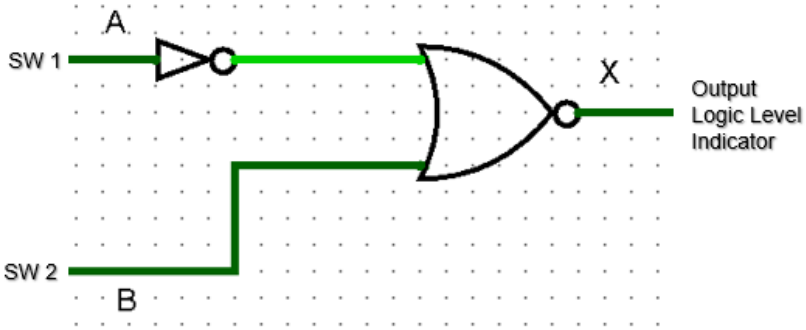


Figure C

7. Set the circuit to each input combination as shown in truth table C, observe its output, and record each in the truth table.
8. Write the logic expression for Figure C.
9. Equate the logic expression for Figure C using De Morgan's first theorem and draw its logic circuit.
10. Connect the circuit that you have drawn and use truth table D to verify the operation of your circuit.

ACTIVITY # 9

Name: _____

Title of the Experiment: _____

Truth Table A

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table C

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table D

A	B	X
0	0	
0	1	
1	0	
1	1	

Answer in step number:

7. _____

8. _____

The logic circuit:

Conclusion:

_____

_____

_____

_____

_____

_____

_____

2.3 DE MORGAN'S SECOND THEOREM

I. Objective

1. To prove experimentally De Morgan's second theorem.

II. Equipment and Materials

- Digital Logic Trainer Board
- 1 pc. NAND Gate logic block (7400 IC, a quad 2-input NAND Gate)
- 1 pc. NOT Gate logic blocks (7404 IC, a hex inverter)
- 1 pc. OR Gate logic blocks (7432 IC, a quad 2-input OR Gate)
- 3-pin connector wires (JST connectors)

III. Introductory Information

De Morgan's second theorem states that the complement of a product equals the sum of the complements.

The Boolean for Figure 1 is $X = \overline{AB}$. The Boolean equation for Figure 2 is $X = \bar{A} + \bar{B}$. The first equation describes a NAND gate, and the second equation, a bubbled OR gate. By De Morgan's second theorem, these two equations are equal and can be equated as $\overline{AB} = \bar{A} + \bar{B}$. Figure 3 is the graphical meaning of De Morgan's second theorem.

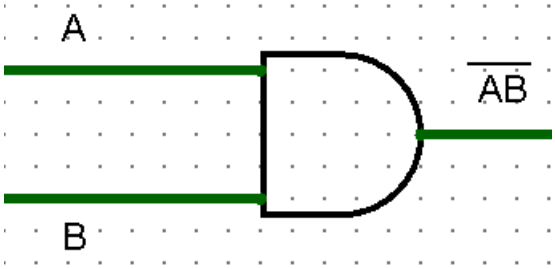
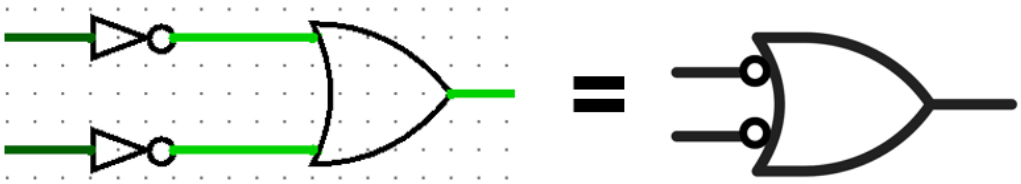


Figure 1. NAND gate



(a) (b)

Figure 2. a. Bubbled OR gate implementation
b. Bubbled OR gate symbol

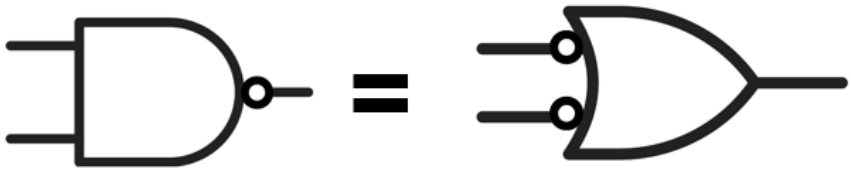


Figure 3. Graphical representation of De Morgan's second theorem

IV. Procedure

1. Plug in the Logic Board and turn on the main switch to power ON the device.
2. Connect the circuit as shown in Figure A.

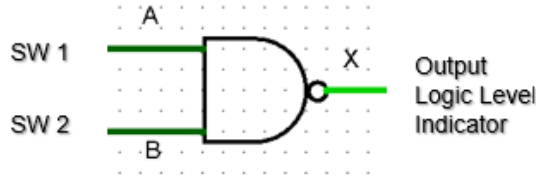


Figure A

3. Set the NAND gate to each input combination shown in truth table A, observe its output, and record each in the truth table.
4. Connect the circuit as shown in Figure B.

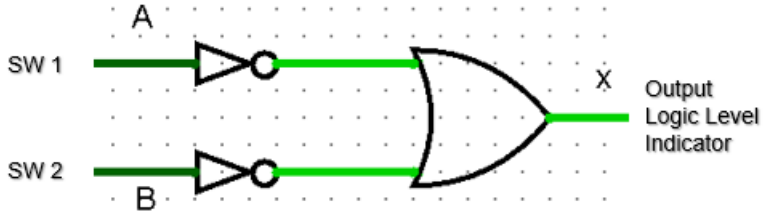


Figure B

5. Set the bubbled OR logic circuit to each input combination as shown in truth table B, observe its output, and record each in the truth table.
6. Connect the circuit as shown in Figure C.

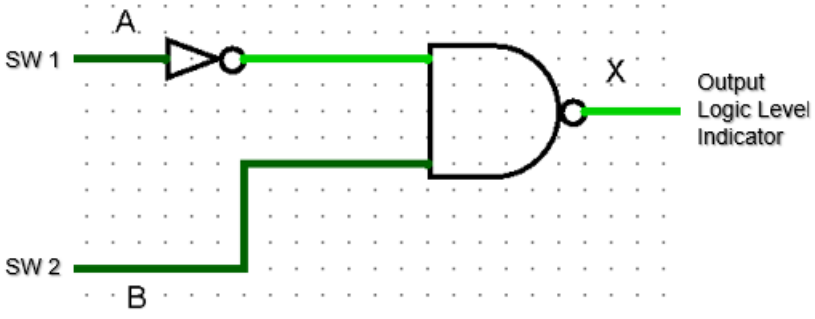


Figure C

7. Set the circuit to each input combination as shown in truth table C, observe its output, and record each in the truth table.
8. Write the logic expression for Figure C.
9. Equate the logic expression for Figure C using De Morgan's second theorem and draw its logic circuit.
10. Connect the circuit that you have drawn and use truth table D to verify the operation of your circuit.

ACTIVITY # 10

Name: _____

Title of the Experiment: _____

Truth Table A

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table B

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table C

A	B	X
0	0	
0	1	
1	0	
1	1	

Truth Table D

A	B	X
0	0	
0	1	
1	0	
1	1	

Answer in step number:

7. _____

8. _____

The logic circuit:

Conclusion:

_____

_____

_____

_____

_____

_____

_____

REFERENCES:

All About Circuits. Multiple-Input Gates. (n.d.). Retrieved October 23, 2024, from https://www.allaboutcircuits.com/textbook/digital/chpt-3/multiple-input-gates/

BYJUS. (2024, February 16). Difference between combinational and sequential circuit. https://byjus.com/gate/difference-between-combinational-and-sequential-circuit/

Electronics Tutorials. (n.d.). Combinational Logic Circuits. Retrieved October 23, 2024, from https://www.electronics-tutorials.ws/combination/comb_1.html

Electronics Tutorials. (n.d.). Digital Logic Gates Summary. Retrieved October 23, 2024, from https://www.electronics-tutorials.ws/logic/logic_10.html

Floyd, T. L. (2015). Digital fundamentals. Utttar Pradesh, India: Pearson India Education Services Pvt. Ltd

GeeksforGeeks. (2024, July 17). Applications of Logic Gates. GeeksforGeeks. https://www.geeksforgeeks.org/applications-of-logic-gates-1/

Gillis, A. S. (2023, December 11). Logic Gate (AND, OR, XOR, NOT, NAND, NOR and XNOR). WhatIs. https://www.techtarget.com/whatis/definition/logic-gate-AND-OR-XOR-NOT-NAND-NOR-and-XNOR

Learn About Electronics. (n.d.). Logic Gates. https://learnabout-electronics.org/Digital/dig21.php

Mano, M. M. (2007). Digital Design. Upper Saddle River, NJ: Prentice-Hall

Mendoza, Romeo G. (n.d.). Digital Logic Trainer Fabrication Manual.

Pedrocillo, R. C. (2023). ECE 6: Logic Circuits and Switching Theory.

Roth, C. H., Jr., Kinney, L. L., & John, E. B. (2020). Fundamentals of Logic Design, 7th edition. Cengage Learning