Department of Biomedical Engineering
Digital Logic Design Lab / Second stage
Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

Experiment No. 5
Combinational Logic Analysis

1. Introduction

Basic combinational logic circuits are digital circuits in which the output depends only on the present input values, without memory or feedback loops. These circuits perform logical operations such as AND, OR, NOT, XOR, etc., and are widely used in arithmetic and data processing applications.

1.1 Objective

The main objectives of this experiment are:

1. To understand the working principles of combinational logic circuits.
2. To design and implement combinational logic circuits based on the basic logic gates (AND, OR, NOT).
3. To analyze the behavior of these circuits through practical experimentation and truth tables, and to minimize them using a suitable method
4. To develop skills in using digital simulation software.

1.2 Work Environment

The CircuitMaker software is used to design and simulate logical circuits.

1.3 Theory

Combinational Logic Circuits

Combinational logic circuits are circuits in which the output depends on the present inputs, without regard to past inputs or states. These circuits are fundamental in digital systems and are used to perform a variety of tasks such as arithmetic operations, data selection, encoding, and decoding. The key characteristic of combinational logic circuits is there is no memory or feedback; once the inputs are provided, the outputs are immediately determined.

Basic Logic Gates: The building blocks of combinational circuits are **logic gates**. These gates perform basic logical functions based on binary inputs. The most common gates used in combinational circuits are:

1. **AND Gate:**

- Output is 1 only when both inputs are 1.
- Logical expression: A. B

2. **OR Gate:**

- Output is 1 if at least one input is 1.
- Logical expression: A+ B

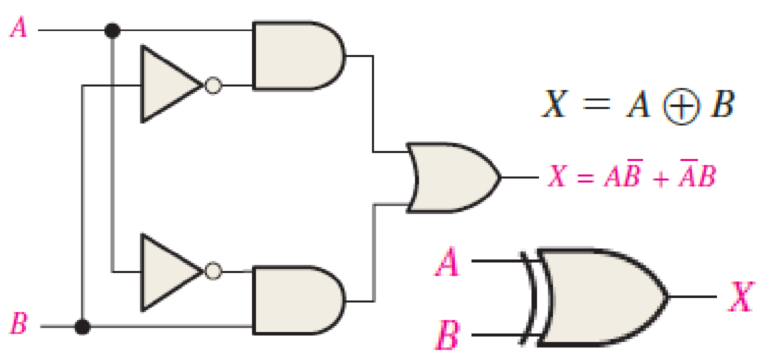
3. **NOT Gate (Inverter):**

- The output is the opposite of the input.
- Logical expression: $\bar{A}$

Lecturer: Dr. Hussam Jawad Kadhim AL_Janabi
Email: hussam.jawad@uomus.edu.iq

4. XOR Gate (Exclusive OR):

The exclusive-OR gate is a combination of two AND gates, one OR gate, and two inverters, as shown in the Figure below



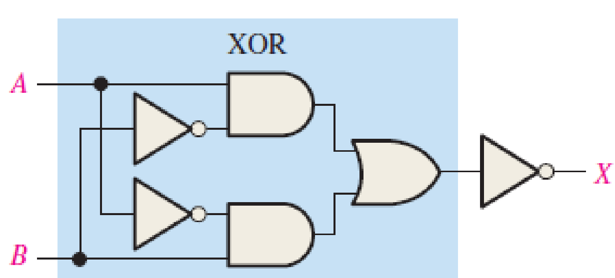
Input		Output
B	A	Y
0	0	0
0	1	1
1	0	1
1	1	0

The output is HIGH only when the two inputs are at opposite levels.

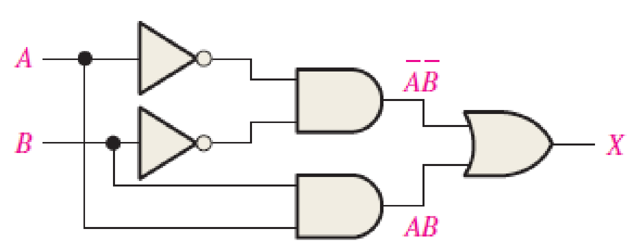
5. XNOR Logic

the complement of the exclusive-OR function is the exclusive-NOR, which is derived as follows:

$$X = \overline{A\overline{B} + \overline{A}B} = \overline{(A\overline{B})}\,\overline{(\overline{A}B)} = (\overline{A} + B)(A + \overline{B}) = \overline{A}\,\overline{B} + AB$$



(a) $X = \overline{A\overline{B} + \overline{A}B}$

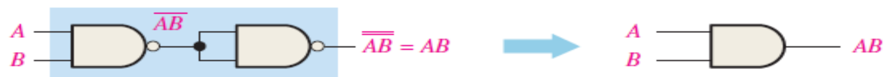


(b) $X = \overline{A}\overline{B} + AB$

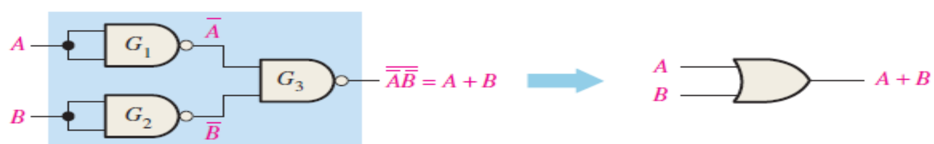
6. NAND

The NAND gate is a **Universal gate** because it can be used to produce the NOT, the AND, the OR, and the NOR functions. An inverter can be made from a NAND gate by connecting all of the inputs together and creating, in effect, a single input

(a) One NAND gate used as an inverter



(b) Two NAND gates used as an AND gate



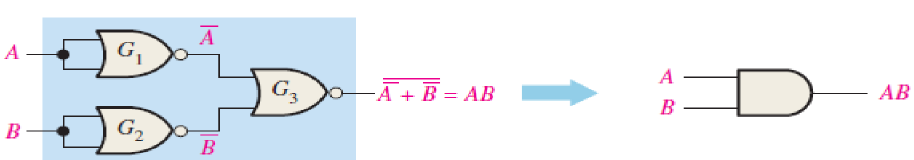
(c) Three NAND gates used as an OR gate

7. NOR

The NOR gate is a **Universal gate** because it can be used to produce the NOT, the AND, the OR, and the NAND functions

(a) One NOR gate used as an inverter

(b) Two NOR gates used as an OR gate

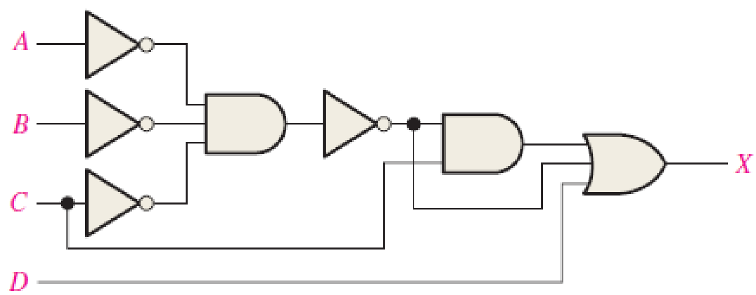


(c) Three NOR gates used as an AND gate

Experiment:

Using CircuitMaker, complete the following steps:

1- Write the logic expression for the output (X) of the following logic circuit



2- Simplify the logic expression using a suitable method.
3- Find the truth table before and after the simplification
4- Draw the simplified logic circuit

Discussion:

1- Which method did you use to simplify the logic expression?
2- Why the simplification is important in logic design?
3- How many gates do you use before simplification and after simplification?