A logo of a university

Description automatically generated

**قــســــــــــم الامــــــــــــــــــــــــــن الــــــــــــــــــــــــــــــــــــسيبرانــــــــــــــــــــــــــــــــــــي**

**Department of Cyber Security**

**Subject:**

**COMPUTER ORGANIZATION & LOGIC DESIGN**

**Class:**

**First**

**Digital Logic Lab: (1)**

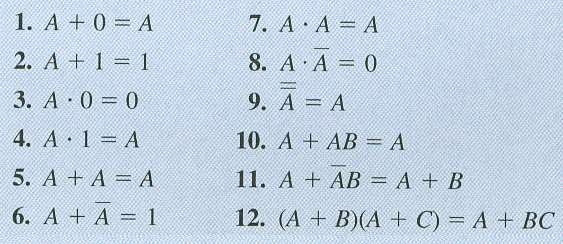
**Msc :Muntather AL-mussawee**

# Boolean algebra

Boolean algebra is the mathematics of digital systems. It is important that you understand is principles thoroughly because a basic knowledge of Boolean algebra is indispensable to the study and analysis of logic circuits.

**--------------------------------------------------------**

**Rules of Boolean Algebra**



1. **Rule 1**  **A + 0**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **A** | | | **0** |  | | |
|  | **0** |  | **0** |  | **0** |  |
|  | **1** |  | **0** |  | **1** |  |

1. **Rule 2**  **A + 1**

|  |  |  |
| --- | --- | --- |
| **A** | **1** |  |
| **1** | **1** | **1** |
| **0** | **1** | **1** |

1. **Rule 3**  **A . 0**

|  |  |  |
| --- | --- | --- |
| **A** | **0** |  |
| **0** | **0** | **0** |
| **1** | **0** | **0** |

1. **Rule 4**  **A . 1**

|  |  |  |
| --- | --- | --- |
| **A** | **1** |  |
| **0** | **1** | **0** |
| **1** | **1** | **1** |

1. **Rule 5**  **A + A**

|  |  |  |
| --- | --- | --- |
| **A** | **A** |  |
| **0** | **0** | **0** |
| **1** | **1** | **1** |

1. **Rule 6**  **A + A'**

|  |  |  |
| --- | --- | --- |
| **A** | **Not - A** |  |
| **0** | **1** | **1** |
| **1** | **0** | **1** |

1. **Rule 7**  **A . A**

|  |  |  |
| --- | --- | --- |
| **A** | **A** | **Not - A** |
| **0** | **0** | **0** |
| **1** | **1** | **1** |

1. **Rule 8**  **A . A'**

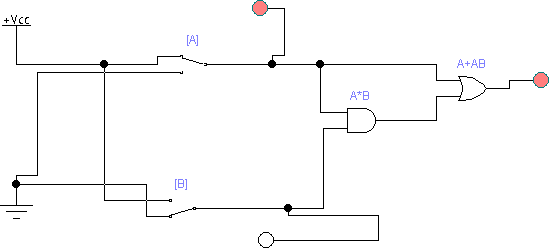
|  |  |  |
| --- | --- | --- |
| **A** | **Not- A** | **Not - A** |
| **0** | **1** | **0** |
| **1** | **0** | **0** |

1. **Rule 9**  **A''**

|  |  |  |
| --- | --- | --- |
| **A** | **Not- A** | **Not - A** |
| **0** | **1** | **0** |
| **1** | **0** | **1** |

1. **Rule 10**  **A + AB = A**

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **A\*B** | **A + AB** |
| **0** | **0** | **0** | **0** |
| **0** | **1** | **0** | **0** |
| **1** | **0** | **0** | **1** |
| **1** | **1** | **1** | **1** |

****

1. **Rule 11**  **A + ĀB = A + B**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** | **Ā** | **ĀB** | **A + ĀB** | **A + B** |
| **0** | **0** | **1** | **0** | **0** | **0** |
| **0** | **1** | **1** | **1** | **1** | **1** |
| **1** | **0** | **0** | **0** | **1** | **1** |
| **1** | **1** | **0** | **0** | **1** | **1** |



1. **Rule 12**  **(A + B)(A + C) = A + BC**

- **Proof of the rule**

**(A + B)(A + C) = AA + AC + BA + BC**

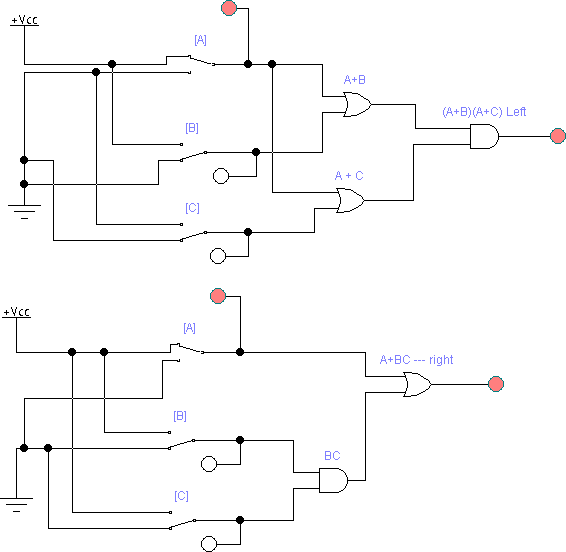
**= A + AC + BA + BC**

**= A + BA + BC**

**= A + BC**

- **Truth Table**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **c** | **A+B** | **A+C** | **(A+B)(A+C)** | **BC** | **A + BC** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** |
| **0** | **1** | **0** | **1** | **0** | **0** | **0** | **0** |
| **0** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |
| **1** | **0** | **0** | **1** | **1** | **1** | **0** | **1** |
| **1** | **0** | **1** | **1** | **1** | **1** | **0** | **1** |
| **1** | **1** | **0** | **1** | **1** | **1** | **0** | **1** |
| **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |

****

# DeMorgan’s Theorems

**--------------------------------------------------------**

Two of the most important theorems of Boolean algebra were contributed by a great mathematician named DeMorgan. DeMorgan’s theorems are extremely useful in simplifying expressions in which a product of sum of variables is inverted. The two theorems are:

|  |  |
| --- | --- |
| ***Theorem 1*** | ***Theorem 2*** |
| **XY**  **X**  **Y** | **X**  **Y**  **XY** |
| XYZW | X+Y+Z+W |
| \_ \_ \_ \_  X + Y + Z + W | \_ \_ \_ \_  XY Z W |

**Ex1: -**

**Sol: -**



**= X + Y + W + Z**

ـــــ\_ـــــــــــــــــــــــــــــــــــ

Ex2:- (A + B) + C

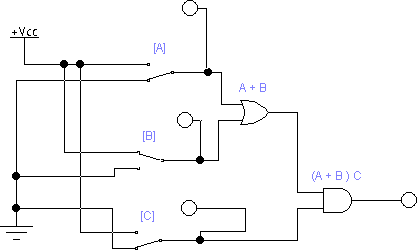
**Sol: -**

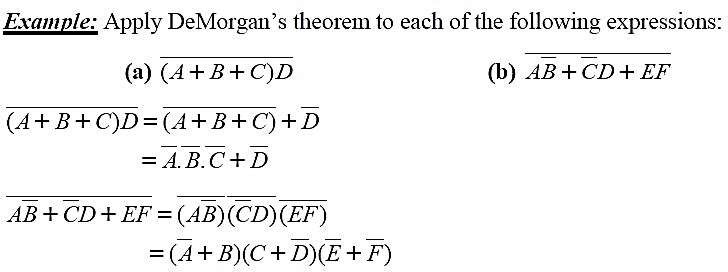
ـــــ ــــــــــــــــــــــــ

* 1. (A + B)C  rule –9 boolean A'' = A
  2. (A'' + B")C''
  3. (A + B)C

23 = 8 (0---7)

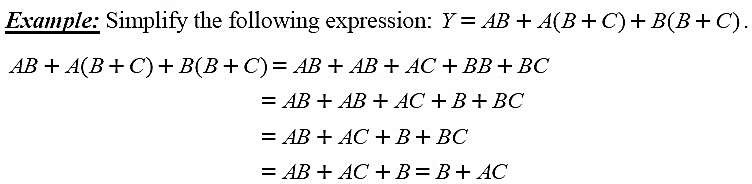
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | A + B | (A+B)C |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |





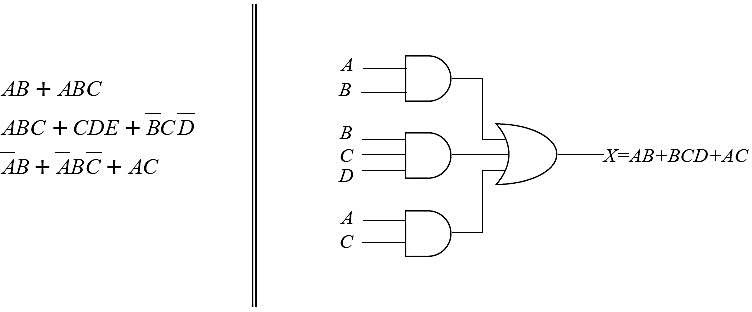
# Simplification using Boolean algebra:

Many times in the application of Boolean algebra, we have to reduce a particular expression to its simplest form or change its form to a more convenient one to implement the expression most efficiently. The purpose of simplifying Boolean expression is to use the fewest gates possible to implement a given expression.



## The Sum-of-Product (SOP) form:

When two or more product terms are summed by Boolean addition, the resulting expression is a sum of product (SOP). Some examples are:



## The Product-of-Sum (POS) form:

When two or more sum terms are multiplied, the resulting expression is a product of sum (POS). Some examples are:

