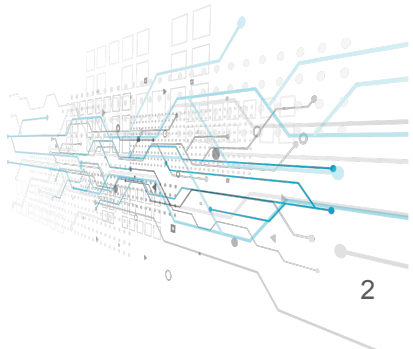
Ministry of Higher Education and Scientific Research
University of Larbi Ben M'Hidi, Oum El Bouaghi
Faculty of Exact Sciences and Natural and Life Sciences
Department of Mathematics and Computer Science

Computer Structure 1

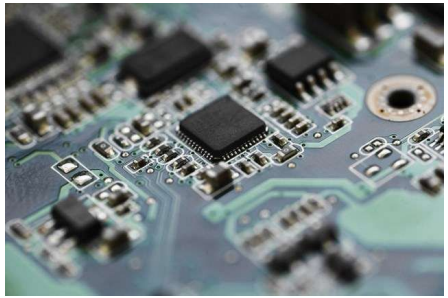
Presented by: **Dr. NASRI/A**
nasri.ahlem1988@gmail.com
nasri.ahlem@univ-oeb.dz

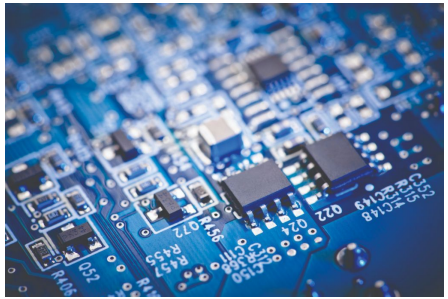
2023-2024

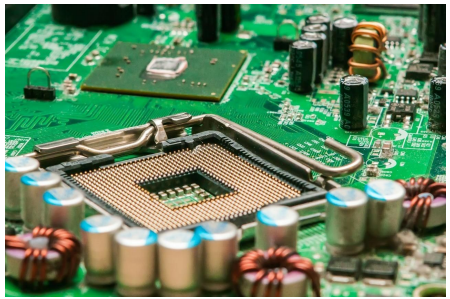
Chapter 3: Boolean Algebra



Introduction







Introduction

- To design and create such a circuit, we must have the mathematical model of its function performed. The mathematical model used is that of Boolean algebra (named after the English mathematician George Boole 1915 - 1864).

- George Boole developed an algebra for manipulating logical propositions using mathematical equations where the statements TRUE and FALSE are represented by the values 1 and 0, while that the AND and OR operators become algebraic multiplication and addition operators.

- Boolean algebra concerns the logic of binary systems.

A- Logical Variable:

A logical (or Boolean) variable is a variable that can take either the value 0 or the value 1. Generally, it is expressed by a single uppercase alphabetical character (A, B, S, etc.)

B- Logical Operators:

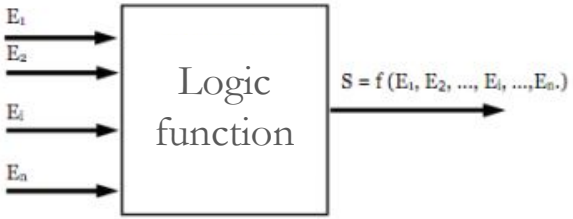
In Boolean Algebra, there are three basic operators: NOT, AND, OR.

C- Logic Gates:

A logic gate is an elementary electronic circuit enabling the function of a logic operator to be carried out.

D- Logic Function

It takes one or more Boolean variables as input, according to which it returns a Boolean value (1 or 0) as output. It is presented either by its logical expression or by its truth table.



D1- Logical Expression:

It is a combination of several variables via logical operators as well as parentheses.

D2- Truth table:

If a logical function has **n logical variables** then the truth table has:

- **(n+1) columns**: n input variables and one output variable
- **2^n lines**: In fact, the function has 2^n Boolean value combinations.

D- Logic Function

Exemple: The function F of three variables A, B and C

Logical Expression: F (A, B, C) = AB + C

Truth table of 4 columns and 8 rows

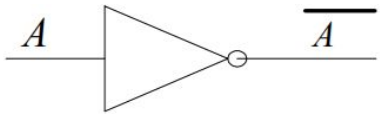
A	B	C	F(A,B,C)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

1- Basic Operators:

All functions can be expressed using the three basic operators: NOT, AND and OR.

1.1- The NOT Operator (Negation):

NOT is a unary operator (a single variable) which has the role of inverting the value of a logical variable.

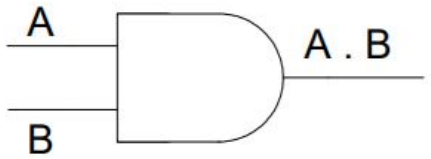
Representation	Truth table	logical gate
F(A)= Not(A)= $\overline{A}$	A | $\overline{A}$ 0 | 1 1 | 0	

1.2- And Operator (Conjunction)

It is defined as follows: a AND b is TRUE if and only if a is TRUE and b is TRUE. This law is also denoted by a point '.'

Representation	Truth table	logical gate
F(A,B)= A*B = A . B = AB		

A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

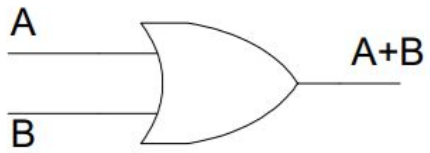


1.3- Or Operator (Disjunction)

It is defined as: a OR b is TRUE if and only if a is TRUE or b is TRUE.
In particular, if a is TRUE and b is also TRUE, then a OR b is true.
This law is also denoted by a plus (+).

Representation	Truth table	logical gate
F(A,B)= A+B		

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1



1.4- Algebraic Properties:

- In the definition of the AND , OR operators above, we just gave the basic definition with only two logical variables.
However, the AND operator (respectively OR) can produce the logical product (resp. the sum) of several logical variables (Example: A *B * C * D / A + B + C +D).

- To evaluate a logical expression (or function), we start by evaluating the sub-expressions between the parentheses, then the complement (NO), then the logical product (AND) and finally the logical sum (OR).

Associativity	$(a+b)+c = a+(b+c) = a+b+c$ As with normal operations, some parentheses are unnecessary: $(a.b).c = a.(b.c) = a.b.c$
Commutativity	$a+b = b+a$ The order is irrelevant: $a.b = b.a$
Distributivity	$a.(b+c) = a.b + a.c$ $a+(b.c) = (a+b)(a+c)$
Idempotence	$a+a+a+a+a\cdots+a = a$ $a.a.a.a.a\ldots.a = a$
Identity	$a+0 = a$ $a.1 = a$
Absorption	$a+1 = 1$ $a.0 = 0$
Simplification	$a+\overline{a}.b = a+b$ $a.(\overline{a}+b) = a.b$
Redundancy	$a.b+\overline{a}.c+b.c = a.b+\overline{a}.c$
Complimentary	$a = \overline{\overline{a}}$ $a.\overline{a} = 0$ $a+\overline{a} = 1$

1.4- Algebraic Properties:

Example:

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

A	B	A+B	AB	$\overline{AB}$	F(A,B)
0	0	0	0	1	0
0	1	1	0	1	1
1	0	1	0	1	1
1	1	1	1	0	0

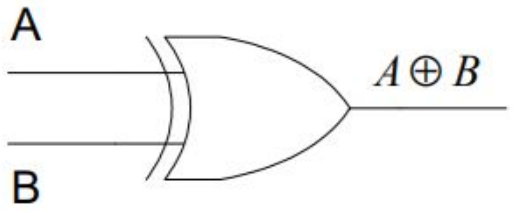
2- Compound Operators:

2.1- The XOR Operator (Exclusive OR)

The XOR operator is 1 if one and only one entry is 1.

Representation	Truth table	logical gate
$F(A, B) = \overline{A}B + A\overline{B} = A \oplus B$		

A	A	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0



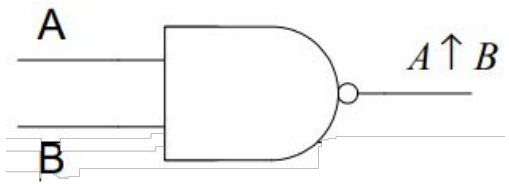
2- Compound Operators:

2.2- The NAND Operator (NOT AND)

The NAND operator is 0 if all input variables are 1.

Representation	Truth table	logical gate
$F(A,B)= \overline{A * B} = A \uparrow B$		

A	A	$A \uparrow B$
0	0	1
0	1	1
1	0	1
1	1	0



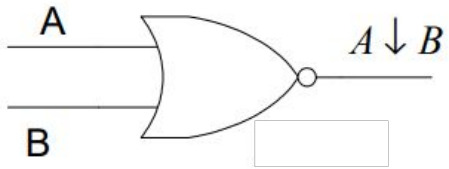
2- Compound Operators:

2.3- The NOR Operator (NOT OR)

The NOR operator is equal to 1 if all input variables are 0.

Representation	Truth table	logical gate
$F(A,B) = \overline{A + B} = A \downarrow B$		

A	A	$A \downarrow B$
0	0	1
0	1	0
1	0	0
1	1	0



NB:

Two logical functions are identical if only if:

- We can show via the properties of Boolean algebra that their logical expressions are identical.
- Their truth tables are identical.

Example: prove that $A \oplus B = \overline{AB}\,(A + B)$

Example: prove that $A \oplus B = \overline{AB}\,(A + B)$

1st using truth table:

A	B		$A \oplus B$		A + B	AB	$\overline{AB}$	$\overline{AB}\,(A + B)$
0	0		**0**		0	0	1	**0**
0	1		**1**		1	0	1	**1**
1	0		**1**		1	0	1	**1**
1	1		**0**		1	1	0	**0**

Example: prove that $A \oplus B = \overline{AB}\,(A + B)$

2nd using algebraic Properties

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

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

$$= 0 + \bar{A}B + \bar{B}A + 0$$

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

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

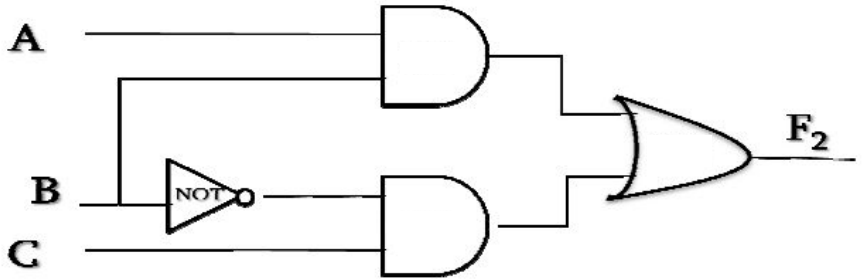
$$= A \oplus B$$

3- Logical Functions:

3.1- Logigram

- The logigram (or logic diagram) is the translation of the logical function into an electronic diagram.
- The principle consists of replacing each logical operator with the logic gate that corresponds to it

Example: $F_2(A, B, C) = AB + \bar{B}C$



3- Logical Functions:

3.2- extracting the logical expression of a function from its truth table

For a logical function with n variables:

- A minterm is the product of n variables (which can be complemented)
- A maxterm is the sum of n variables (which can be complemented)

From the truth table, the logical expression is defined either as:

- **The sum of minterms** where a minterm is determined for each value of the function equal to **1.**
- **The product of the maxterms** where a maxterm is determined for each value of the function equal to **0.**

Example:

Example:

A	B	C	F	Type of term	Expression of term
0	0	0	0	Maxterm	$A + B + C$
0	0	1	0	Maxterm	$A + B + \bar{C}$
0	1	0	0	Maxterm	$A + \bar{B} + C$
0	1	1	1	Minterm	$\bar{A}\,B\,C$
1	0	0	0	Maxterm	$\bar{A} + B + C$
1	0	1	1	Minterm	$A\,\bar{B}\,C$
1	1	0	1	Minterm	$A\,B\,\bar{C}$
1	1	1	1	Minterm	$A\,B\,C$

F = Sum of Minterm

F (A, B, C) = $\bar{A}\,B\,C + A\,\bar{B}\,C + A\,B\,\bar{C} + A\,B\,C$ → 1st canonical form (Disjunctive)

F (A, B, C) = $\sum(011,101,110,111)$ = numerical form

F = Product of Maxterm

F (A, B, C) = $(A + B + C)(A + B + \bar{C})(A + \bar{B} + C)(\bar{A} + B + C)$ → 2nd canonical form (Conjunctive)

F (A, B, C) = $\prod(000, 001, 010, 100) = \prod(0, 1, 2, 4)$ → numerical form

3- Logical Functions:

3.3- Canonical Form

We call the canonical form of a function the form where each term of the function includes all the variables. There are two canonical forms:

- First Canonical Form: Disjunctive Form: which is the sum of the minterms; A disjunction. This form is the most used form.
- Second Canonical Form: Conjunctive Form: which is the product of the maxterms: a conjunction.

Note here that the first and second canonical forms are equivalent.

3- Logical Functions:

3.3- Canonical Form

Transition to canonical forms:

We can always reduce any logical function to one of the canonical forms.

To do, we add the missing variables in the terms which do not contain all the variables (the non-canonical terms).

This is possible using the rules of Boolean algebra:
- Multiply a term with an expression that is worth 1
- Add to a term with an expression that is 0
- Subsequently distribute

3- Logical Functions:

3.3- Canonical Form

Transition to canonical forms:

Example: $F_2(A,B,C) = \overline{A\bar{B}\bar{C} + \bar{A}B}$

Conjonctive Form

$$F_{2C}(A,B,C) = \overline{A\bar{B}\bar{C} + \bar{A}B}$$

$$= \left(\overline{A\bar{B}\bar{C}}\right)\left(\overline{\bar{A}B}\right)$$

$$= (\bar{A} + \bar{B} + C)(A + \bar{B})$$

$$= (\bar{A} + \bar{B} + C)\left((A + \bar{B}) + C\bar{C}\right)$$

$$= (\bar{A} + \bar{B} + C)(A + \bar{B} + C)\,(A + \bar{B} + \bar{C})$$

3- Logical Functions:

3.3- Canonical Form

Transition to canonical forms:

Example: $F_2(A,B,C) = \overline{AB\bar{C}} + \bar{A}B$

Disjonctive Form

$$F_{2D}(A,B,C) = \overline{AB\bar{C}} + \bar{A}B$$

$$= (\overline{AB\bar{C}})(\overline{\bar{A}B})$$

$$= (\bar{A} + \bar{B} + C)(A + \bar{B})$$

$$= \bar{A}A + \bar{A}\bar{B} + \bar{B}A + \bar{B}\bar{B} + CA + C\bar{B}$$

$$= 0 + \bar{A}\bar{B} + A\bar{B} + \bar{B} + AC + \bar{B}C$$

$$= \bar{A}\bar{B}(\boldsymbol{C} + \bar{\boldsymbol{C}}) + A\bar{B}(\boldsymbol{C} + \bar{\boldsymbol{C}}) + (\boldsymbol{A} + \bar{\boldsymbol{A}})\bar{B}(\boldsymbol{C} + \bar{\boldsymbol{C}}) + A(\boldsymbol{B} + \bar{\boldsymbol{B}})C + (\boldsymbol{A} + \bar{\boldsymbol{A}})\bar{B}C$$

$$= \bar{A}\bar{B}\boldsymbol{C} + \bar{A}\bar{B}\bar{\boldsymbol{C}} + A\bar{B}\boldsymbol{C} + \boldsymbol{A}\bar{B}\bar{\boldsymbol{C}} + A\bar{B}\boldsymbol{C} + \boldsymbol{A}\bar{B}\bar{\boldsymbol{C}} + \bar{A}\bar{B}\boldsymbol{C} + \bar{A}\bar{B}\bar{\boldsymbol{C}} + ABC + A\bar{B}\boldsymbol{C} + A\bar{B}\boldsymbol{C} + \bar{A}\bar{B}\boldsymbol{C}$$

$$= \bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C} + A\bar{B}C + A\bar{B}\bar{C} + ABC$$

3- Logical Functions:

3.4- Simplification

The canonical forms of a logical function are a correct definition of the function, but they can be simplified to:

- Express the same function with as few terms and as simple as possible.
- Perform the function with fewer electronic elements (logic gates).

Simplification can be achieved through either method:

- **The properties of Boolean algebra (Algebraic method).**
- **Map Karnaugh (graphic method).**

3- Logical Functions:

3.4- Simplification

The properties of Boolean algebra (Algebraic method).

- $F_1(A,B) = \bar{A}B + AB$
- $F_2(A,B) = (A+B)(A+\bar{B})$
- $F_3(A,B) = A + AB$
- $F_4(A,B) = A\,(A+B)$

3- Logical Functions:

3.4- Simplification

The properties of Boolean algebra (Algebraic method).

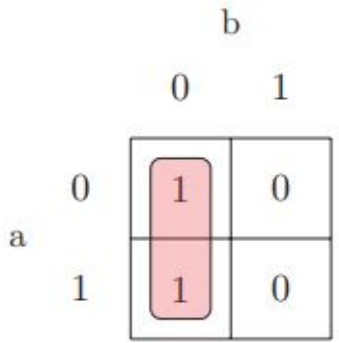
- $F_1(A,B) = \bar{A}B + AB = (\bar{A} + A)B = 1\,B = B$
- $F_2(A,B) = (A+B)(A+\bar{B}) = A\,A + A\,\bar{B} + B\,A + B\,\bar{B} = A + A\bar{B} + A\,B + 0 = A + A\,(\bar{B} + B) = A + A\,1 = A + A = A$
- $F_3(A,B) = A + AB = A\,(1+B) = A\,1 = A$
- $F_4(A,B) = A\,(A+B) = A\,A + A\,B = A + AB$

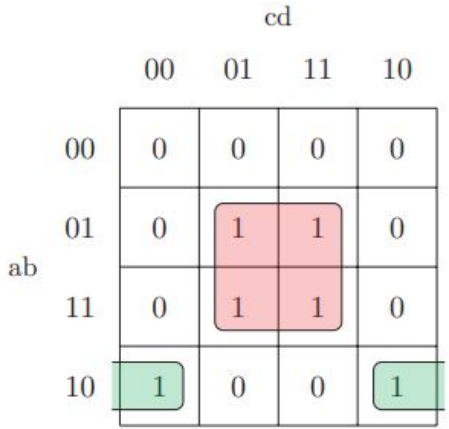
3- Logical Functions:

3.4- Simplification

Map Karnaugh (graphic method)

The Karnaugh map is a graphical tool for simplifying a logic equation or the process of going from a truth table to a corresponding circuit.





3- Logical Functions:

3.4- Simplification

Map Karnaugh (graphic method)

Method:

- Join adjacent "1" in groups of 1, 2, 4, 8,... etc. (2^n, n=0,1,2,... etc)
- The equation of the circuit is given by the sum of the products of the variables which do not change state in each grouping.

- Then, in the previous examples: $S_1 = \overline{b}$ and $S_2 = b.d + a.\overline{b}.\overline{d}$

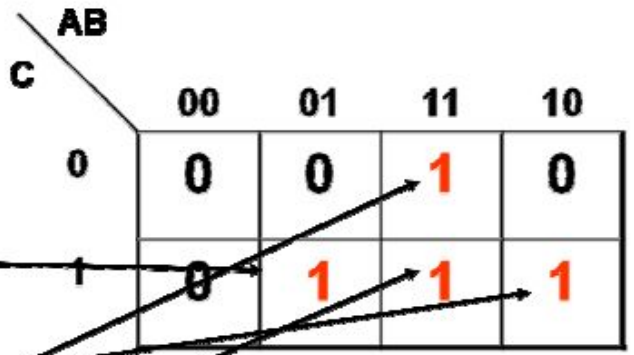
NB: An output $\overline{S}$ is obtained by the groupings of zeros.

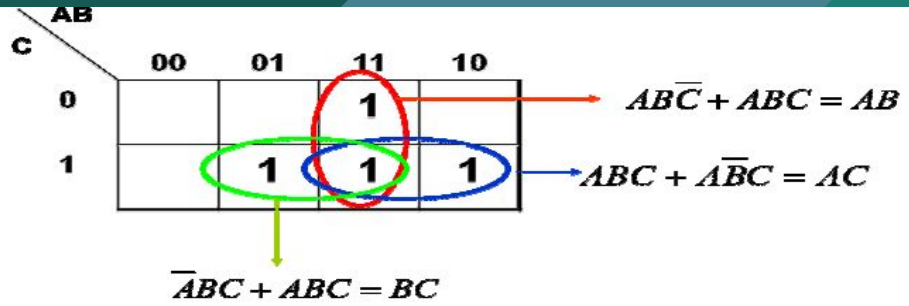
3- Logical Functions:

3.4- Simplification

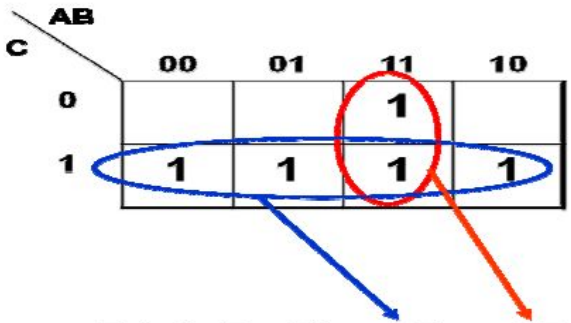
Exemple:

A	B	C	S
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

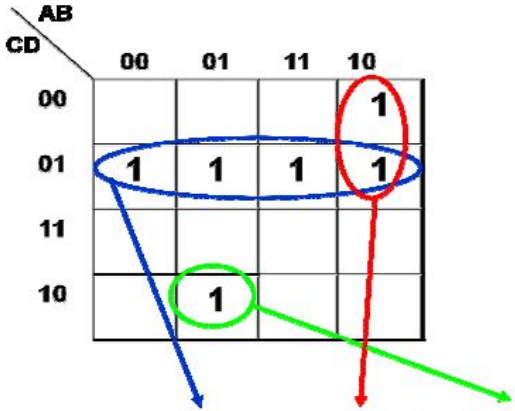




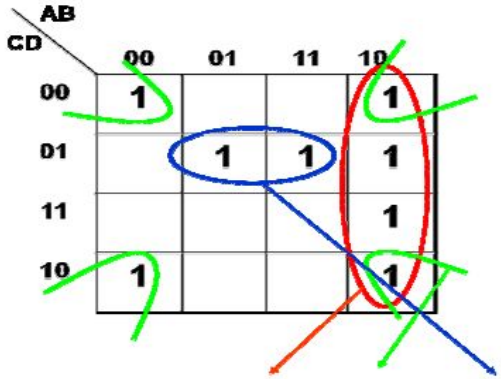
$$F(A,B,C) = AB + AC + BC$$



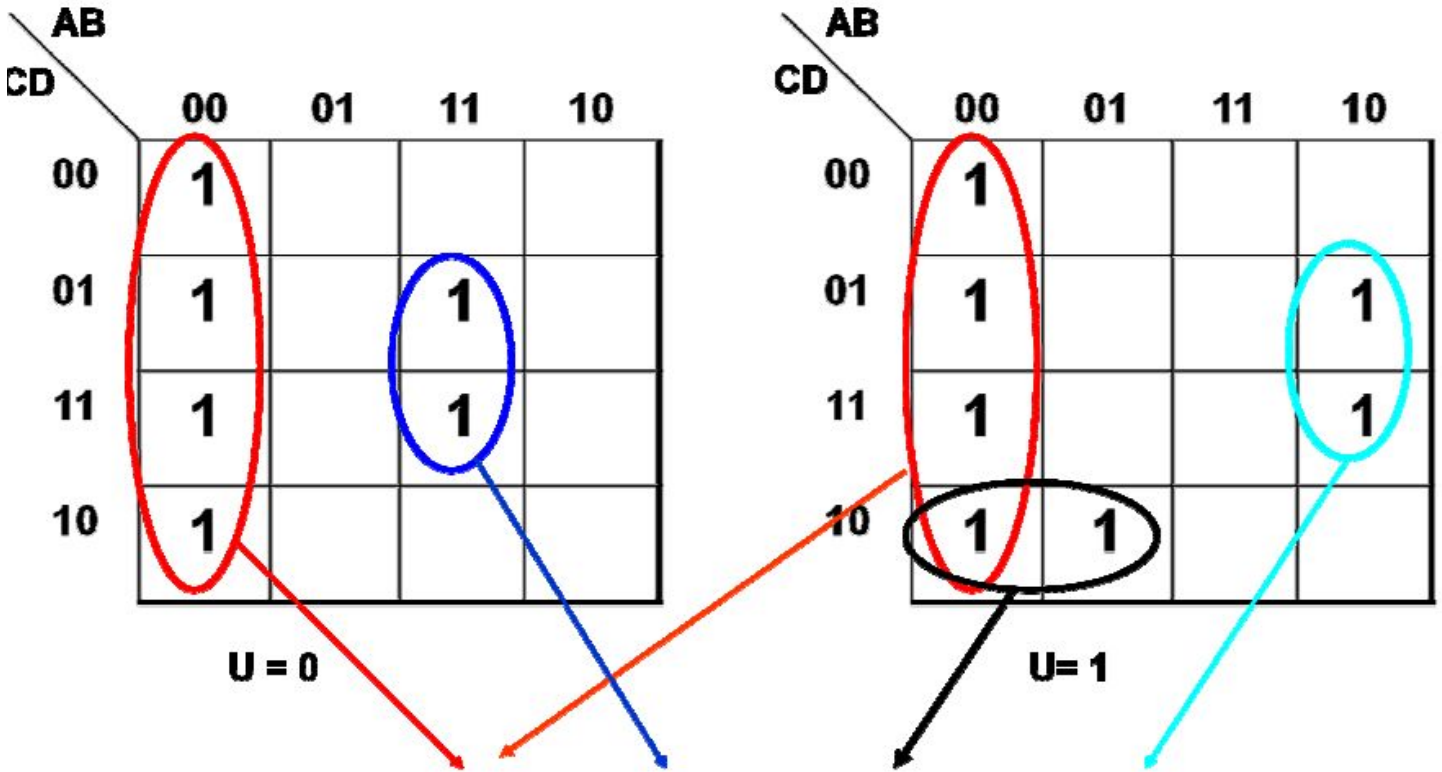
$$F(A,B,C) = C + AB$$



$$F(A,B,C,D) = \overline{C}.D + A.\overline{B}.\overline{C} + \overline{A}.B.C.\overline{D}$$



$$F(A,B,C,D) = A\overline{B} + \overline{B}\overline{D} + B\overline{C}D$$



$$F(A, B, C, D, U) = \overline{A}\,\overline{B} + A.B.D.\overline{U} + \overline{A}.C.\overline{D}.U + A.\overline{B}.D.U$$

3- Logical Functions:

3.4- Simplification

Example with 5 variables:

E = 0

CD \ AB	00	01	11	10
00				
01				
11				
10				

E = 1

CD \ AB	00	01	11	10
00				
01				
11				
10				

Map with 5 variables

3- Logical Functions:

3.4- Simplification

Example with 6 variables:

EF = 00

AB / CD	00	01	11	10
00				
01				
11				
10				

EF = 01

AB / CD	00	01	11	10
00				
01				
11				
10				

EF = 11

AB / CD	00	01	11	10
00				
01				
11				
10				

EF = 10

AB / CD	00	01	11	10
00				
01				
11				
10				

Map with 6 variables

3- Logical Functions:
3.4- Study of a logic function

Steps :

1. Truth table
2. Canonical Forms
3. Simplification (algebric or Karnaugh map)
4. logigram drawing
   (diagram of logic gates)