

Chapitre 4

Algèbre de Boole

I- Introduction

L'algèbre de Boole, du nom du mathématicien anglais George Boole (1815-1864), est une formulation algébrique de tous les processus en résonnant logiquement par un oui (désignant le vrai) ou bien un non (désignant le faux). D'importantes applications du domaine des ordinateurs et des appareils de mesure numérique reposent sur cette algèbre.

II. Définitions

- ❖ L'**algèbre de Boole** est un ensemble de variables à deux états, de valeur de vérité 1 (vrai), 0 (faux) muni d'un nombre limité d'opérateurs : NON, ET, OU
- ❖ Un ordinateur ne manipule que des données binaires, on appelle donc **variable logique (variable booléenne)** une donnée binaire, c'est-à-dire une donnée ayant deux états possibles: 0 ou 1.
- ❖ **Table de vérité** : Les valeurs des variables soumises aux opérateurs pour toutes les combinaisons possibles des valeurs de ces variables sont indiquées sous forme de tables appelées table de vérité.

III. Opération de base de l'algèbre de Boole

III.1 Opération à une variable (opération NON)

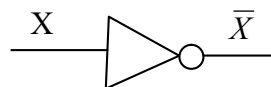
Soit X une variable booléenne ; sa négation NON X , appelée aussi complément de X sera noté $\bar{X}$ (X barre), sera aussi une variable booléenne.

Par définition X peut prendre deux valeurs 0 ou 1 donc pour l'opération NON la table de vérité est la suivante :

Entrée		Sortie
X		$\bar{X}$
0		1
1		0

Table de vérité (ou TV)

On voit que $\bar{X}$ est une variable booléenne « inversée » par rapport à X d'où le nom d'**inverseur** donné au dispositif effectuant cette opération, appelé aussi l'opérateur d'inversion



Symbole de l'inverseur
Opérateur NON (NOT)

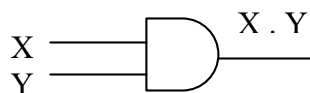
III.2 Opération à deux variables

III.2.1 Opération ET (AND)

Soit X et Y deux variables booléennes. Le résultat de X ET Y ($X \text{ AND } Y$) est donné selon la table de vérité suivante :

Entrée			Sortie
X	Y		$X \text{ ET } Y$
0	0		0
0	1		0
1	0		0
1	1		1

Les combinaisons possibles des entrées apparaissent dans l'ordre binaire naturel. L'opérateur ET (AND) est noté « . » c.a.d $X \text{ ET } Y = X . Y$, le résultat peut être désigné par une autre lettre Z par exemple d'où $X \text{ ET } Y = Z$ (ou bien $Z = X . Y$). Le dispositif effectuant cette opération est appelée opérateur (porte) ET (AND) représenté graphiquement (symboliquement) comme suit :



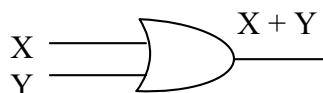
Symbole ET (AND)

III.2.2 Opération OU (OR)

Soit X et Y deux variables booléennes. Le résultat de X OU Y ($X \text{ OR } Y$) est donné selon la table de vérité suivante :

Entrée			Sortie
X	Y		$X \text{ OU } Y$
0	0		0
0	1		1
1	0		1
1	1		1

L'opérateur OU (OR) est noté « + » c.a.d $X \text{ OU } Y = X + Y$, le résultat peut être désigné par une autre lettre Z par exemple d'où $X \text{ OU } Y = Z$ (ou bien $Z = X + Y$). Le dispositif effectuant cette opération est appelée opérateur (porte) OU (OR) représenté graphiquement (symboliquement) comme suit :



Symbole OU (OR)

IV. Axiomes ou lois fondamentales de l'algèbre de Boole

Soit A, B, C trois variables booléennes et soit les opérateurs NON, ET et OU ainsi que leurs tables de vérités vue précédemment.

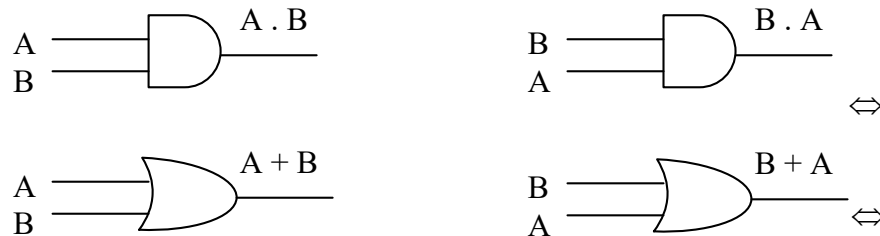
Les lois fondamentales de l'algèbre de Boole sont des règles logiques qui permettent de simplifier l'écriture des expressions algébriques.

IV.1 Lois de fermeture

- a- Le résultat de l'opération Booléenne $A \cdot B$ est une variable booléenne (0 ou 1).
- b- Le résultat de l'opération Booléenne $A + B$ est une variable booléenne (0 ou 1).

IV.2 Lois de commutativité

- $A \cdot B = B \cdot A$
- $A + B = B + A$



IV.3 Lois d'associativité

- $(A \cdot B) \cdot C = A \cdot (B \cdot C)$
- $(A + B) + C = A + (B + C)$

IV.4 Lois de distributivité

- $A + (B \cdot C) = (A + B) \cdot (A + C)$
- $A \cdot (B + C) = A \cdot B + A \cdot C$

IV.5 Lois d'idempotence

- $A \cdot A = A$
- $A + A = A$

IV.6 Lois de complémentarité

- $A + \bar{A} = 1$
- $A \cdot \bar{A} = 0$

IV.7 Lois de la double négation (Involution)

- $\overline{\overline{A}} = A$

IV.8 L'élément neutre

- $A \cdot 1 = A$ donc 1 est l'élément neutre de l'opération (.) AND
- $A + 0 = A$ donc 0 est l'élément neutre de l'opération (+) OU

IV.9 L'absorption

- $A \cdot 0 = 0$ donc 0 est l'élément d'absorption de l'opération (.) AND
- $A + 1 = 1$ donc 1 est l'élément d'absorption de l'opération (+) OU

IV.10 Théorème de De Morgan

- La négation d'un produit de variables est égale à la somme des négations des variables.

Exemple : $\overline{xyz} = \overline{x} + \overline{y} + \overline{z}$

- La négation d'une somme de variables est égale au produit des négations des variables.

Exemple : $\overline{x+y+z} = \overline{x} \cdot \overline{y} \cdot \overline{z}$

Il s'ensuit donc les expressions suivantes

$$\left. \begin{array}{l} 0 + 0 = 0 \\ 1 + 1 = 1 \end{array} \right\} \text{Duales}$$

$$\left. \begin{array}{l} 1 \cdot 1 = 1 \\ 0 \cdot 0 = 0 \end{array} \right\} \text{Duales}$$

$$\left. \begin{array}{l} 1 \cdot 0 = 0 \\ 0 + 1 = 1 \end{array} \right\} \text{Duales}$$

$$\left. \begin{array}{l} \overline{0} = 1 \\ \overline{1} = 0 \end{array} \right\} \text{Duales}$$

Nous appelons ces expressions "duales" car en remplaçant dans une même équation logique, les 0 par 1, les « . » par des « + » est inversement, cette équation reste vérifiée.

Exemple : En utilisant la TV démontrer le premier théorème de De Morgan.

Solution

X	Y	Z	$\bar{X}$	$\bar{Y}$	$\bar{Z}$	X.Y.Z	$\overline{X.Y.Z}$	$\bar{X} + \bar{Y} + \bar{Z}$
0	0	0	1	1	1	0	1	1
0	0	1	1	1	0	0	1	1
0	1	0	1	0	1	0	1	1
0	1	1	1	0	0	0	1	1
1	0	0	0	1	1	0	1	1
1	0	1	0	1	0	0	1	1
1	1	0	0	0	1	0	1	1
1	1	1	0	0	0	1	0	0

D'après la table on peut conclure que le théorème 1 de De Morgan est vérifié soit $\overline{X.Y.Z} = \bar{X} + \bar{Y} + \bar{Z}$

Remarques

- La soustraction n'existe pas dans l'algèbre de Boole :

$$a + b = a + c \text{ n'implique pas que } b = c.$$

- Une variable booléenne n'a ni puissance ni multiple :

$$a.a.a.a.... = a \text{ il n'existe pas de } a^n$$

$$a + a + a + = a \text{ il n'exista pas de } n.a$$

V. Fonctions logiques**V.1 Définition**

On appelle **fonction logique (fonction booléenne)** une entité constituée d'une combinaison de variables booléennes reliées par les opérateurs « NON, ET, OU » acceptant plusieurs valeurs logiques en entrée et dont la sortie (il peut y en avoir plusieurs) peut avoir deux états possibles : 0 ou 1.

Exemple

$$z = (x + \bar{y})(x + y) \longrightarrow \text{Fonction booléenne de 2 variables x et y}$$

Une fonction booléenne peut être définie par :

- Sa table de vérité.
- Son expression.
- Sa table de Karnaugh.
- Son logigramme.

V.2 Définition par table de vérité

Une fonction de n variables présente 2^n résultats possibles donc on peut décrire cette fonction avec une TV de 2^n lignes, chaque ligne de cette table représente la valeur de la fonction pour une configuration binaire de n variables.

Exemple

$M = F(a,b,c)$ est une fonction booléenne qui prend « 0 » si la majorité des variables vaut « 0 » et « 1 » dans le cas contraire. Donner la TV de la fonction M .

Solution

a	b	c	M
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

V.3 Définition par expression

Il existe une notation plus pratique pour représenter (définir) une fonction booléenne :

- On spécifie toutes les combinaisons des variables d'entrée qui fournissent à la fonction un résultat égal à 1.
- Par convention on place une barre horizontale sur une variable pour indiquer qu'elle vaut 0.
- L'absence de cette barre signifie qu'elle vaut 1.
- On utilise « + » pour exprimer le OU et « . » (ou pas de signe) pour le ET.

Exemple 1

$\overline{A}B + B\overline{C}$ Signifie ($A = 1$ et $B = 0$) ou ($B = 1$ et $C = 0$)

Exemple 2

Donner l'expression de la fonction booléenne M définie précédemment par la TV.

Solution

$$M = \overline{a}bc + a\overline{b}c + ab\overline{c} + abc$$

V.4 Définition par Table de Karnaugh

La table de vérité présente un inconvénient : le nombre de ligne croît en même temps que le nombre de variables c.a.d : si on a n variable $\Rightarrow 2^n$ lignes (3 variables $\Rightarrow 8$ lignes, 4 variables $\Rightarrow 16$ lignes ...etc).

Pour remédier à cet inconvénient on utilise une nouvelle table appeler **Table de Karnaugh**.

- ❖ Les lignes et les colonnes doivent être numérotées selon le code binaire réfléchi.
- ❖ Le nombre de cases, constituant le tableau, est fonction du nombre de variables mises en jeu.

y \ x	0	1
0		
1		

Tableau à 2 variables

z \ xy	00	01	11	10
0				
1				

Tableau à 3 variables

zt \ xy	00	01	11	10
00				
01				
11				
10				

Tableau à 4 variables

tu \ xyz	000	001	011	010	110	111	101	100
00								
01								
11								
10								

Tableau à 5 variables

Exemple

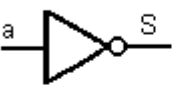
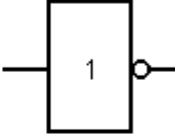
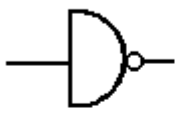
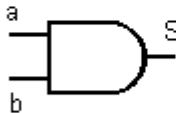
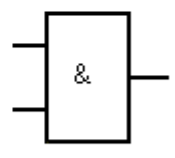
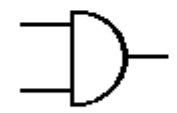
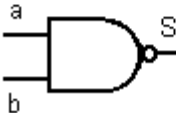
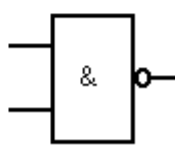
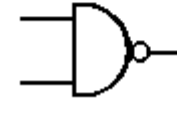
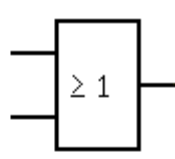
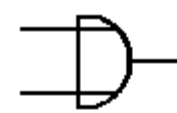
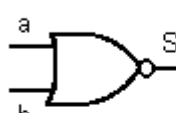
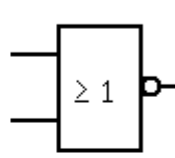
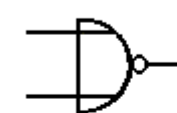
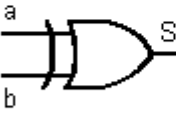
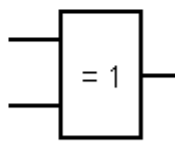
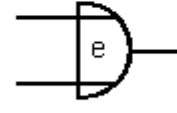
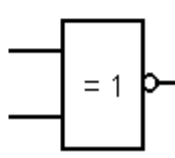
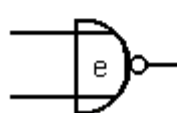
Soit $F = a\bar{b} + \bar{c}\bar{d}$ donner la table de karnaugh de F

Solution

ab \ cd	00	01	11	10
00	1	0	0	0
01	1	0	0	0
11	1	0	0	0
10	1	1	1	1

V.5 Définition par Logigramme

Connaissant les symboles logiques (voir tableau ci dessous), on peut construire immédiatement le logigramme (digramme logique) d'une fonction booléenne.

FONCTION	EQUATION	SYMBOLES			TABLES DE VERITE		
		International	Français	Allemand	a	b	S
NON	$S = \overline{a}$				a	S	
					0	1	
ET	$S = a \cdot b$				0	0	0
					0	1	0
					1	0	0
					1	1	1
NAND	$S = \overline{a \cdot b}$				0	0	1
					0	1	1
					1	0	1
					1	1	0
OU	$S = a + b$				0	0	0
					0	1	1
					1	0	1
					1	1	1
NOR	$S = \overline{a + b}$				0	0	1
					0	1	0
					1	0	0
					1	1	0
OU Exclusif	$S = a \oplus b$				0	0	0
					0	1	1
					1	0	1
					1	1	0
NOR Exclusif	$S = \overline{a \oplus b}$				0	0	1
					0	1	0
					1	0	0
					1	1	1

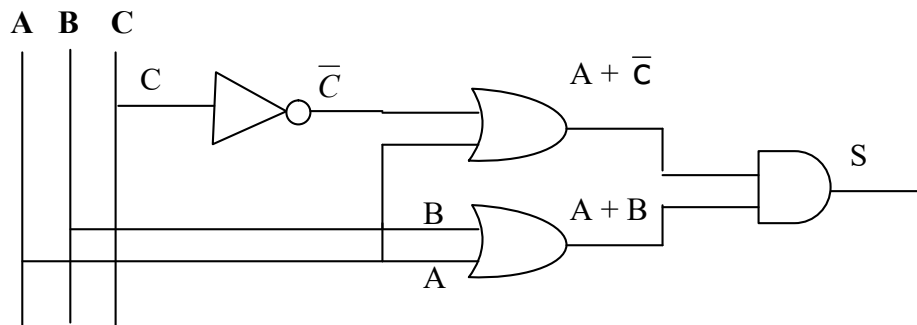
NB : Les symboles sur lesquels nous nous basons pour la conception de nos logigrammes sont ceux de la norme internationale.

Remarque

- Les portes ET , OU , NAND , NOR peuvent avoir plus que deux entrées.
- Il n'existe pas de OU exclusif à plus de deux entrées

Exemple

Soit l'expression algébrique de la fonction booléenne S. Dresser le logigramme de S
 $S = (A+B).(A+\bar{C})$

Solution**VI. Universalité des portes NAND et NOR**

En utilisant les portes logique NAND et NOR on peut exprimer n'importe qu'elle expression logique. Pour cela, Il suffit d'exprimer les opérateurs de base (NON, ET, OU) avec des NAND et des NOR.

VI.1 Portes logiques A base de NAND

$$\bar{A} = \overline{A.A}$$

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

$$A.B = \overline{\overline{A.B}}$$

VI.2 Portes logiques A base de NOR

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

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

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

VII. Formes canoniques des fonctions logiques

On appelle forme canonique (normale) d'une fonction logique la forme où chaque terme de la fonction comportent toutes les variables. Il existent plusieurs formes canoniques : les plus utilisées sont la première et la deuxième forme.

Pour une fonction logique à n variables

- Un minterme : groupe des n variables (pouvant être complémentées) liées par des ET.
- Un maxterme : groupe des x variables (pouvant être complémentées) liées par des OU.

Exemple : Soit f une fonction logique de 4 variables $\{a, b, c, d\}$

Mintermes : $a.b.c.d, a.\bar{b}.c.\bar{d}, \bar{a}.b.c.\bar{d}, \dots$ etc.

Maxtermes: $(a + b + c + d)$, $(a + b + c + \bar{d})$, $(\bar{a} + b + \bar{c} + d)$

Pour la fonction f : $a.\bar{b}.c$ n'est pas un minterme

Pour la fonction f : $(\bar{a} + b + d)$ n'est pas un maxterme

VII.1 Première forme canonique (forme disjonctive)

La première forme canonique est une union (OU) de mintermes c-a-d est une somme de produits de mintermes on dit aussi une disjonction de conjonctions. Cette forme est la forme la plus utilisée.

Il faut noter que la première forme canonique c'est l'expression des « 1 » de la fonction logique.

Exemple

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

VII.2 Deuxième forme canonique (forme conjonctive)

La deuxième forme canonique est une intersection (ET) de maxtermes c-a-d un produit de sommes de maxtermes on dit aussi une conjonction de disjonctions

Il faut noter que la deuxième forme canonique c'est l'expression des « 0 » de la fonction logique.

Exemple

$$F(a, b, c) = (a + b + c). (a + \bar{b} + \bar{c}). (\bar{a} + \bar{b} + \bar{c}). (\bar{a} + b + c)$$

Exemple : donner la première et la deuxième forme canonique à partir de la table de vérité ci-dessous :

A	B	C	S	
0	0	0	0	$\rightarrow A + B + C$: Max terme
0	0	1	0	$\rightarrow A + B + \bar{C}$: Max terme
0	1	0	0	$\rightarrow A + \bar{B} + C$: Max terme
0	1	1	1	$\rightarrow \bar{A}.B.C$: Min terme
1	0	0	0	$\rightarrow \bar{A} + B + C$: Max terme
1	0	1	1	$\rightarrow A.\bar{B}.C$: Min terme
1	1	0	1	$\rightarrow A.B.\bar{C}$: Min terme
1	1	1	1	$\rightarrow A.B.C$: Min terme

Première forme canonique : $S =$ somme de min termes

$$S = \bar{A}.B.C + A.\bar{B}.C + A.B.\bar{C} + A.B.C$$

Deuxième forme canonique : $S =$ produit de max termes

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

Remarque 1

- Les deux formes canoniques sont équivalentes. Mais pour faire un choix, On choisit celle qui donne le résultat le plus simple c-à-d :
 Pour un minimum de « 0 » on choisit la deuxième forme canonique.
 Pour un minimum de « 1 » on choisit la première forme canonique.
- Si la fonction n'est pas sous forme normale c-a-d une des variables (au moins) ne se trouve pas dans un des termes, la fonction est dite sous une forme simplifiée.

Exemple

$$\begin{aligned}
 F &= x.y.z + \bar{x}.y.\bar{z} + x.y.\bar{z} \quad (\text{Première forme canonique}) \\
 &= x.y(z + \bar{z}) + \bar{x}.y.\bar{z} \quad (\text{Forme simplifiée}) \\
 &= y(x + \bar{x}.\bar{z}) \quad (\text{Forme simplifiée})
 \end{aligned}$$

Remarque 2

- On peut toujours ramener n'importe quelle fonction logique à l'une des formes canoniques. Cela revient à rajouter les variables manquants dans les termes qui ne contiennent pas toutes les variables (les termes non canoniques).
- Cela est possible en utilisant les règles de l'algèbre de Boole :
 - Multiplier un terme avec une expression qui vaut 1
 - Additionner à un terme avec une expression qui vaut 0
 - Par la suite faire la distribution

Exemple

1. Le passage à la première forme canonique de la fonction F est comme suit :

$$\begin{aligned}
 F(A, B) &= A + B \\
 &= A(B + \bar{B}) + B(A + \bar{A}) \quad \text{Car } (A + \bar{A}) = 1 \text{ et } (B + \bar{B}) = 1 \\
 &= AB + A\bar{B} + AB + \bar{A}B \\
 &= AB + A\bar{B} + \bar{A}B
 \end{aligned}$$

2. Faire le passage à la première forme canonique de la fonction suivante en utilisant les règles de l'algèbre de Boole : $S(a, b, c) = ab + \bar{b}c + a\bar{c}$

En analysant la fonction S nous remarquons que pour avoir une première forme canonique il faut transformer les 3 produits en mintermes.

Dans le terme ab nous manque la variable c donc :

$$ab = ab(c + \bar{c}) \text{ car } (c + \bar{c}) = 1 \text{ et } \bar{b}c = \bar{b}c(a + \bar{a}) \text{ et } a\bar{c} = a\bar{c}(b + \bar{b})$$

Donc on aura donc

$$\begin{aligned}
 S(a, b, c) &= ab(c + \bar{c}) + \bar{b}c(a + \bar{a}) + a\bar{c}(b + \bar{b}) \\
 S(a, b, c) &= abc + a\bar{b}c + ab\bar{c} + \bar{a}\bar{b}c + a\bar{b}\bar{c}
 \end{aligned}$$

Remarque 3

- Il existe une autre représentation des formes canoniques d'une fonction, cette représentation est appelée forme numérique.

Exemple

Soit une fonction F a trois variables A, B, C :

1. La forme numérique de la première forme canonique est la suivante :

$$F = \Sigma(2,4,6) = \Sigma(010,100,110) = \bar{A}.B.\bar{C} + A.\bar{B}.\bar{C} + A.B.\bar{C}$$

2. La forme numérique de deuxième forme canonique est la suivante :

$$\begin{aligned} F &= \prod(0,1,3,5,7) = \prod(000,001,011,101,111) \\ &= (A + B + C)(A + B + \bar{C})(A + \bar{B} + \bar{C})(A + \bar{B} + C)(\bar{A} + \bar{B} + \bar{C}) \end{aligned}$$

VII. Simplification des fonctions booléennes

VII.1 Simplification algébrique

Simplifier algébriquement une fonction booléenne revient à la développer, effectuer une mise en facteur et la simplifier selon les lois fondamentales. Donc simplifier une fonction revient à l'écrire à l'aide d'un nombre minimum de termes.

Exemple

Simplifier la fonction booléenne suivante $Z = (a+b)(\bar{b}+c)(\bar{a}+c)$

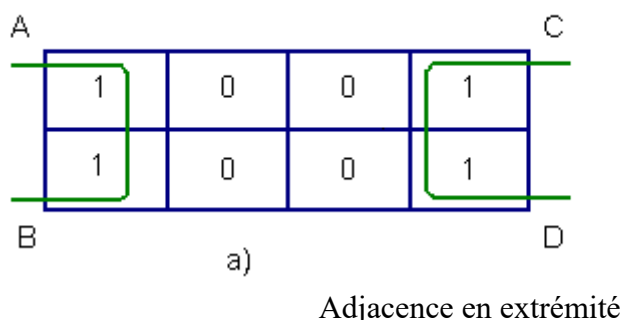
Solution $Z = c(b+a)$

VII.2 Simplification par table de karnaugh

Nous avons vu que les lignes et les colonnes de la table de karnaugh sont numérotées selon le code Gray et on sait que dans ce code deux nombres adjacents (l'un à la suite de l'autre) ne sont différents que par un seul bit. Cette propriété nous servira pour les simplifications.

VII.2.1 Première règle de simplification

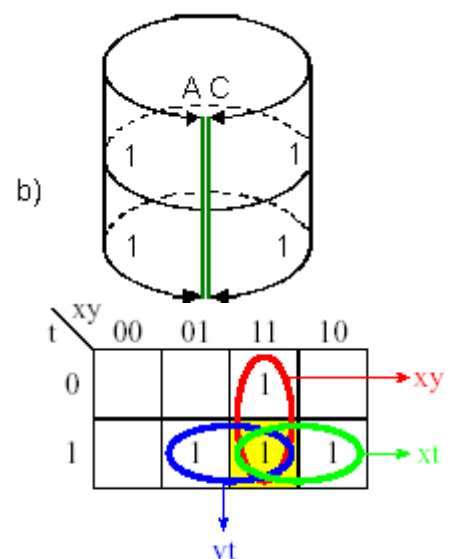
On peut regrouper deux « 1 » adjacents dans une somme de produits en éliminant la variable qui change d'état. Il faut noter aussi que l'adjacence existe aussi pour les extrémités de la table.



Exemple

Nous y observons trois groupements de deux termes, nous pouvons écrire pour la fonction :

$$F = xy + xt + yt$$



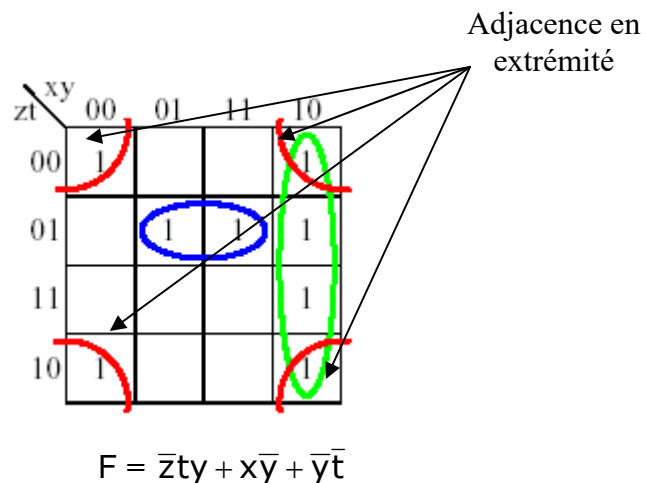
VII.2.2 Deuxième règle de simplification

On peut regrouper quatre « 1 » adjacents dans une somme de produit et éliminer les deux variables qui changent d'état. Cette règle se généralise à toutes les puissances de 2 : 4, 8, 16,etc. Donc on forme des groupes de « 1 » aussi grand que possible : 2, 4, 8, 16,etc.

Exemple :

Soit **F** une fonction définie par sa table de vérité. Représenter et simplifier **F** par la table de Karnaugh.

x	y	z	t	F
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

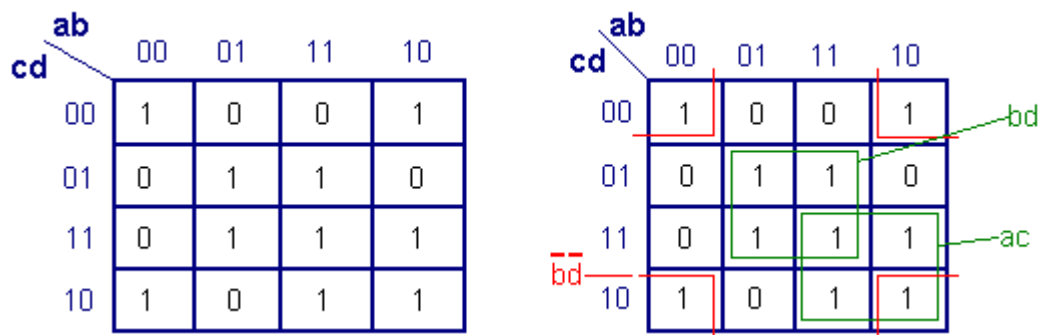


VII.2.3 Troisième règle de simplification

Un même « 1 » peut être introduit dans plusieurs groupes.

Exemple : simplifier la fonction $L = bd + \bar{b}\bar{c}\bar{d} + a\bar{b}cd + \bar{b}c\bar{d} + abcd$

Après simplification la fonction est donnée par $L = ac + bd + \bar{b}\bar{d}$



VII.2.4 Cas particulier de simplification

Il arrive que la fonction logique ne soit pas spécifiée pour certaines combinaisons de variables d'entrée. On dit alors que les valeurs dans ces cases de la table de karnaugh sont indifférentes et on les désigne par x ou -. Ces valeurs indifférentes seront choisies de manière à faciliter la formation des groupes de « 1 ». On peut attribuer n'importe quelle valeur à ces cases.

Exemple

$M = F(a,b,c,d)$ est une fonction booléenne qui prend « 0 » si la majorité des variables vaut « 0 » et « 1 » dans le cas contraire. Donner la TV de la fonction M, puis dresser et simplifier F par karnaugh.

Solution

Dec	a	b	c	d	M
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	-
4	0	1	0	0	0
5	0	1	0	1	-
6	0	1	1	0	-
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	-
10	1	0	1	0	-
11	1	0	1	1	1
12	1	1	0	0	-
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

ab \ cd	00	01	11	10
00	0	0	-	0
01	0	-	1	-
11	-	1	1	1
10	0	-	1	-

D'après la table de Karnaugh nous obtenons : $M = F(a,b,c) = ad + bc$

VII.3 Simplification par la méthode de QUINE-MAC CLUSKEY

Bien que peu employée, nous devons quand même citer une méthode différente de celle des tableaux de Karnaugh pour la simplification des fonctions de plusieurs variables. Il s'agit d'une méthode imaginée par le mathématicien américain **W. V. QUINE** et remodelée par son compatriote le **Docteur Edwards J. Mac CLUSKEY Junior** dans une thèse de doctorat qu'il présenta au **M. I. T. (Massachusetts Institute of Technology)** en juin 1956.

Description de la méthode

Pour examiner cette méthode, nous prendrons un exemple à quatre variables. Partons de la table de vérité de la fonction $f(a, b, c, d)$ qui est donnée par la TV suivante :

Nous pouvons constater que la fonction f est «vraie» (égale à 1) pour onze combinaisons des variables a, b, c et d .

Donnons comme nom à chacune de ces combinaisons la valeur décimale qui correspond au code binaire formé par les variables a, b, c, d en considérant que a est le bit de poids fort et d le bit de poids faible.

Exemple

$0110 = \bar{a}bcd$ nous l'appellerons la combinaison **6** puisque **0110** en binaire équivaut à **6** en décimal. Si nous récapitulons, la fonction **f** est donc la somme des combinaisons résumées au tableau suivant :

a	b	c	d	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

a	b	c	d	Combinaisons
0	0	0	0	Combinaison 0
0	0	0	1	Combinaison 1
0	0	1	0	Combinaison 2
0	1	0	0	Combinaison 4
0	1	1	0	Combinaison 6
1	0	0	0	Combinaison 8
1	0	0	1	Combinaison 9
1	1	0	0	Combinaison 12
1	1	0	1	Combinaison 13
1	1	1	0	Combinaison 14
1	1	1	1	Combinaison 15

Cette liste peut être récapitulée de la façon suivante :

$$f = \sum (0, 1, 2, 4, 6, 8, 9, 12, 13, 14, 15)$$

↓ Fonction
 «somme de»
 ↓ Liste des combinaisons

A partir de cette liste de combinaisons commence réellement la méthode de QUINE-MAC CLUSKEY.

La première chose à faire est de classer ces combinaisons en un ensemble successif en fonction du nombre de zéros de la forme binaire de chaque combinaison en commençant par celle qui en compte le plus.

Le premier ensemble est donc formé par la combinaison **0** dont la forme binaire **0000** comporte quatre zéros. Le second ensemble regroupe les combinaisons comportant trois zéros ... jusqu'au cinquième qui est formé de la combinaison **15** dont la forme binaire **1111** ne comporte pas de zéro du tout. Les cinq ensembles sont représentés par la table suivante :

	a	b	c	d	
Combinaison 0	0	0	0	0	ensemble 1 (4 zéros)
Combinaison 1	0	0	0	1	ensemble 2 (3 zéros)
Combinaison 2	0	0	1	0	
Combinaison 4	0	1	0	0	
Combinaison 8	1	0	0	0	
Combinaison 6	0	1	1	0	ensemble 3 (2 zéros)
Combinaison 9	1	0	0	1	
Combinaison 12	1	1	0	0	
Combinaison 13	1	1	0	1	ensemble 4 (1 zéro)
Combinaison 14	1	1	1	0	
Combinaison 15	1	1	1	1	ensemble 5 (0 zéro)

La seconde phase de la méthode consiste à comparer les termes de chaque ensemble avec les termes de l'ensemble qui suit immédiatement de façon à éliminer une variable si cela est possible.

La combinaison **0000** de l'ensemble **1** doit donc être comparée aux quatre combinaisons de l'ensemble **2**. Puis chacune des quatre combinaisons de cet ensemble **2** sera comparée à chacune des trois combinaisons de l'ensemble **3** et ainsi de suite jusqu'à l'ensemble **5**.

Lorsque deux combinaisons ne diffèrent que d'une variable, celles-ci peuvent être associées pour n'en former plus qu'une dans laquelle la variable qui diffère peut être éliminée.

Quand une variable s'élimine dans une telle association, on signale ce fait en remplaçant cette variable par une croix (ou tout autre signe à votre convenance).

Dans l'exemple que nous avons choisi, les combinaisons **0** et **1** peuvent, par exemple, être associées puisque seule la variable **d** est différent :

Combinaison 0	: 0 0 0 0
Combinaison 1	: 0 0 0 1
Combinaison 0 - 1:	0 0 0 X

Procédons ainsi en comparant chaque terme de chaque ensemble avec chaque terme de l'ensemble suivant et nous obtenons les combinaisons décrites dans la table suivante :

	a	b	c	d	
0 - 1	0	0	0	X	ensembles 1 et 2 I
0 - 2	0	0	X	0	
0 - 4	0	X	0	0	
0 - 8	X	0	0	0	
1 - 9	X	0	0	1	ensembles 2 et 3 II
2 - 6	0	X	1	0	
4 - 6	0	1	X	0	
4 - 12	X	1	0	0	
8 - 9	1	0	0	X	
8 - 12	1	X	0	0	
6 - 14	X	1	1	0	ensembles 3 et 4 III
9 - 13	1	X	0	1	
12 - 13	1	1	0	X	
12 - 14	1	1	X	0	
13 - 15	1	1	X	1	ensembles 4 et 5 IV
14 - 15	1	1	1	X	

Nous obtenons de nouveaux membres regroupés à présent dans quatre ensembles **I**, **II**, **III** et **IV**. Recommençons la même opération avec ces nouveaux ensembles. La table ci-dessous donne les nouvelles combinaisons obtenues.

			a	b	c	d
ensembles I et II	0 - 1	8 - 9	X	0	0	X
	0 - 2	4 - 6	0	X	X	0
	0 - 4	2 - 6	0	X	X	0
	0 - 4	8 - 12	X	X	0	0
	0 - 8	1 - 9	X	0	0	X
	0 - 8	4 - 12	X	X	0	0
ensembles II et III	4 - 6	12 - 14	X	1	X	0
	4 - 12	6 - 14	X	1	X	0
	8 - 9	12 - 13	1	X	0	X
	8 - 12	9 - 13	1	X	0	X
ensembles III et IV	12 - 13	14 - 15	1	1	X	X
	12 - 14	13 - 15	1	1	X	X

Dans les principes généraux de la méthode de QUINE-MAC CLUSKEY, il convient de continuer ce processus jusqu'à ce qu'il n'y ait plus de combinaison possible entre un de ces ensembles et le suivant.

Dans l'exemple choisi, nous en sommes arrivés à ce point. De plus, nous constatons dans cette table que plusieurs combinaisons sont identiques. Nous ne garderons bien sûr que les groupements différents que nous appellerons **A, B, C, D, E** et **F**.

La fonction **f** prend alors la forme :

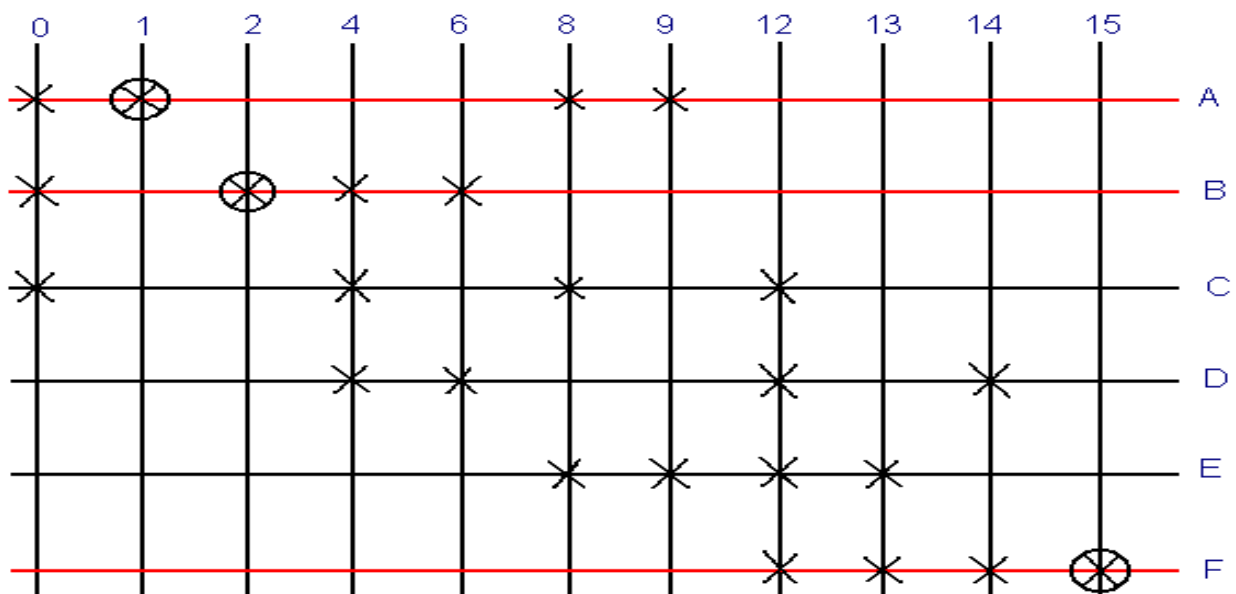
$$f = A + B + C + D + E + F, \text{ Soit : } f = \bar{b}\bar{c} + \bar{a}\bar{d} + \bar{c}\bar{d} + b\bar{d} + a\bar{c} + ab$$

Remarque

Dans tous les regroupements que nous avons effectués, il aurait pu arriver que certains termes d'un ensemble ne se combinent avec aucun terme de l'ensemble suivant. Ces termes sont alors des termes premiers et il convient de ne pas les oublier dans l'équation finale.

Arrivé à ce stade, il convient de vérifier si l'équation obtenue ne peut pas encore être simplifiée. Pour éliminer les éventuels termes superflus, formons une grille dite «diagramme des termes premiers».

Ce diagramme représenté par la figure ci-dessous comporte onze colonnes correspondant aux combinaisons initiales **0, 1, 2, 4, 6, 8, 9, 12, 13, 14** et **15**. Il comprend d'autre part six lignes horizontales correspondant aux six groupements **A, B, C, D, E** et **F** obtenus après les différents processus de simplification.



Sur chaque ligne, marquons d'une croix les combinaisons ayant servies à former le groupement considéré.

Par exemple pour le groupement **A** qui fait intervenir les combinaisons **0, 1, 8** et **9**, nous avons coché sur la ligne **A** les colonnes **0, 1, 8** et **9**. Après avoir fait de même pour les lignes **B, C, D, E** et **F**, nous constatons que certaines colonnes (**1, 2** et **15**) ne comportent qu'une seule croix que nous entourons alors d'un cercle. Ces combinaisons sont situées sur les lignes **A, B**, et **F** qui sont appelées «**lignes de base**».

Nous constatons que d'autre part les combinaisons incluses dans les lignes **C, D, E** se retrouvent au moins une fois dans les lignes **A, B** et **F**. Les lignes **C, D** et **E** peuvent donc être supprimées et la fonction **f** se simplifie donc encore pour devenir :

$$f = A + B + F \text{ donc } f = \overline{b}\overline{c} + \overline{a}\overline{d} + ab$$

VII.4 Comparaison avec la méthode de karnaugh

A titre de comparaison, effectuons le même exercice par la méthode des tableaux de KARNAUGH. A l'aide de la table de vérité, établissons le tableau de KARNAUGH correspondant :

Les regroupements effectués dans ce tableau (cerclés dans la figure) donnent immédiatement la solution :

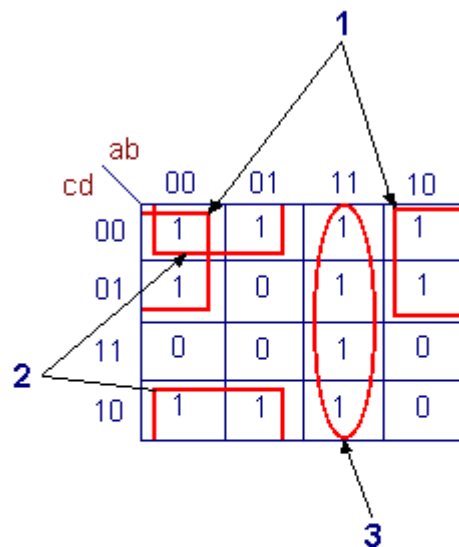
$$f = 1 + 2 + 3$$

$$f = \overline{b}\overline{c} + \overline{a}\overline{d} + ab$$

On retrouve le même résultat que précédemment.

a	b	c	d	f
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

a) Table de vérité



b) Tableau de Karnaugh

VII.5 Conclusion :

A la lumière de cet exemple, résolu d'une part avec la méthode de QUINE MAC CLUSKEY, d'autre part grâce à un tableau de Karnaugh, il apparaît que la seconde est nettement plus simple et rapide que la première. Ceci explique que cette première méthode soit quasiment inemployée.

Précisons toutefois à sa décharge que la méthode de QUINE MAC CLUSKEY s'applique quel que soit le nombre de variables alors que la méthode des tableaux de Karnaugh se complique notablement lorsque le nombre de variables augmente et devient supérieur à 6 ou 7.