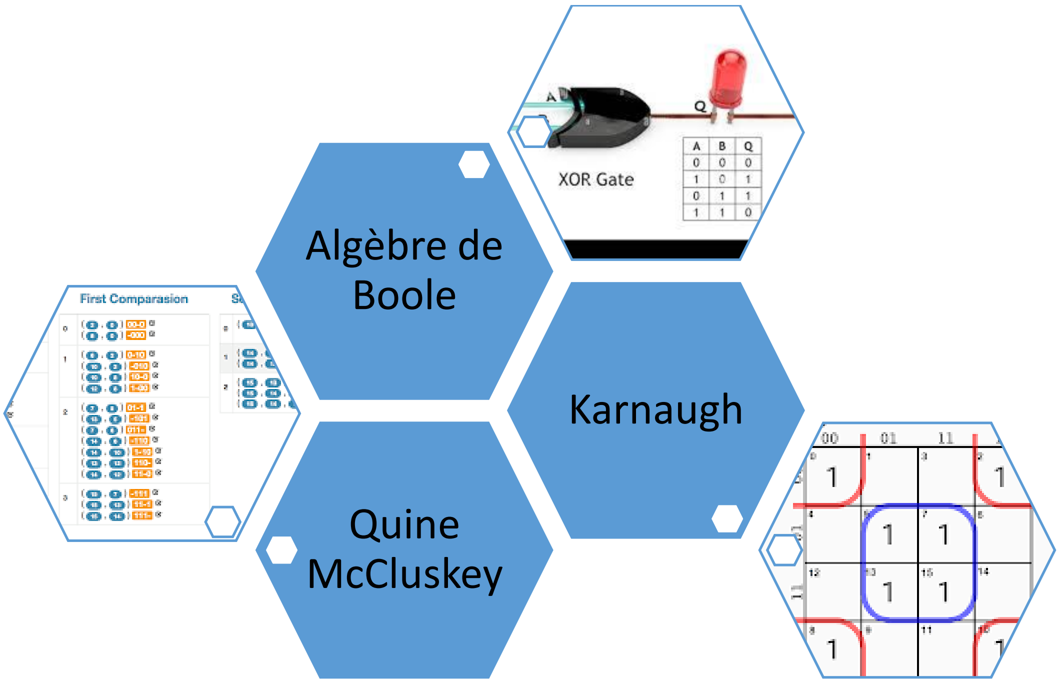
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université Djilali Bounaâma de Khemis Miliana
Faculté des sciences et techniques
Département de maths et informatique
Niveau : Licence 1ère année MI

Rédigé par Dr. MAHROUG RABIAA
E-mail : r.mahroug@univ-dbkm.dz

Chapitre IV : L'algèbre de Boole binaire



Année universitaire 2022-2023

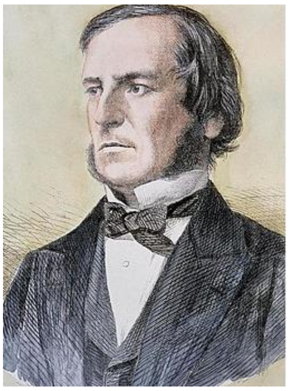
Table des matières

Abréviation

Bit	: Binary Digit
B ou b	: Base
MSB	: Most Significant Bit
LSB	: Least Significant Bit
SVA	: Signe et Valeur Absolue
Cà1	: Complément à 1
Cà2	: Complément à 2
BCD	: Binary Coded Decimal
BCD+3	: Binary Coded Decimal+3
IEEE 754	: IEEE Standard for Floating-Point Arithmetic 754
ASCII	: Americain Standard Code Information Interchange
EBCDIC	: Extended Binary Coded Decimal Interchange Code
UTF	: (Unicode Transformation Format
F ou f	: Fonction
FND	: La première forme canonique disjonctive
FNC	: La deuxième forme canonique conjonctive

Chapitre IV : L'algèbre de Boole binaire

4.1. Définition et axiomes de l'algèbre de Boole



George Boole, né le 2 novembre 1815 à Lincoln (Royaume-Uni) et mort le 8 décembre 1864 à Ballintemple (Irlande), est un logicien, mathématicien et philosophe britannique. Il est le créateur de la logique moderne, fondée sur une structure algébrique et sémantique, que l'on appelle algèbre de Boole en son honneur.
Il a aussi travaillé dans d'autres domaines mathématiques, des équations différentielles aux probabilités en passant par l'analyse. Autodidacte, il publia ses premiers travaux d'algèbre tout en exerçant son métier d'instituteur et de directeur d'école dans la région de Lincoln. Ses travaux lui valurent en 1844 la Royal Medal de la Royal Society, puis une chaire de mathématiques à l'université (Queen's College) de Cork en 1849.

En électronique numérique on manipule des variables logiques conventionnellement repérées par les valeurs 0 ou 1. Ces grandeurs obéissent à des règles d'algèbre particulières qu'il est indispensable de maîtriser avant d'entreprendre l'analyse ou la synthèse de circuits numériques. Dans ce chapitre nous énoncerons les principes et les règles de calcul de l'algèbre logique, appelé aussi algèbre de Boole, puis nous les appliquerons à l'écriture et à la manipulation des fonctions logiques.

4.1.1. Variables et fonctions logiques

4.1.1.1. Variables logiques

On appelle variable logique une variable qui ne peut prendre que deux valeurs conventionnellement repérées par 0 et 1. On parle aussi de variable binaire. Chacune de ces deux valeurs est associée à une grandeur physique, par exemple la tension collecteur d'un transistor, ce qui permet de faire le lien entre une étude théorique utilisant l'algèbre de Boole et un circuit électronique [17].

4.1.1.2. Fonctions logiques

Une fonction logique F des n variables logiques (x_1, x_2,…, x_n) , notée par exemple F = F(x_1, x_2,..., x_n) , associe une valeur 0 ou 1 aux différentes combinaisons possibles des n variables logiques (x_1, x_2,..., x_n). Chaque variable logique xi pouvant prendre la valeur 0 ou 1, il y a au total 2n combinaisons possibles des variables logiques (x_1, x_2,..., x_n) et on définit complètement une fonction logique en donnant sa valeur pour chacune de ces combinaisons [18].

Les fonctions logiques peuvent être représentées sous forme de tables, appelées tables de vérité, donnant la valeur de la fonction pour chaque combinaison des variables logiques. Une table de vérité est une énumération complète de toutes les combinaisons des valeurs des entrées du circuit avec, pour chacune d'elles, la valeur des sorties associées [19].

Considérons par exemple une fonction F de deux variables **X** et **Y**. Il y a donc 2^2 = 4 combinaisons possibles de ces deux variables. Une table de vérité donne la valeur de F pour chacune des 2^2 combinaisons possibles de ces 2 variables. On trouve généralement 2 types de représentations comme indiqué ci-dessous.

X	Y	F
0	0	0
0	1	1
1	0	0
1	1	1

Y \ X	0	1
0	0	1
1	0	1

L'écriture de la table de vérité fait partie de l'analyse d'un système donné. A l'inverse une fois la table de vérité connue, il faut pouvoir déterminer le schéma électronique permettant de réaliser cette table : c'est la phase de synthèse.

4.2. Les opérateurs de base

Une algèbre logique se définit par l'existence de trois lois, ou fonctions logiques de base. L'outil mathématique qui permet de décrire les systèmes combinatoires est l'Algèbre de Boole [20] . Par la combinaison des trois fonctions de base que sont le NON, OU, ET, on va pouvoir décrire chacune des sorties en fonction des entrées.

4.2.1. Fonction inversion NON (NOT)

La fonction NON est un opérateur qui affecte à la variable de sortie l'état complémentaire de la variable d'entré. Cette fonction est également appelée complément

Notation : $S = \overline{X}$

Table de vérité

X	$S = \overline{X}$
0	1
1	0

$S = \overline{X}$

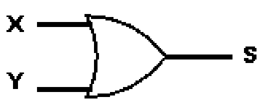
4.2.2. Fonction OU (OR)

C'est une fonction de deux variables également appelée somme logique

Notation : $S = X + Y$

Table de vérité

X	Y	$S = X + Y$
0	0	0
0	1	1
1	0	1
1	1	1



$S = X + Y$

La fonction OU vaut 1 si au moins une des variables vaut 1.

4.2.3. Fonction ET (AND)

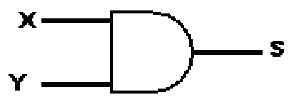
C'est une fonction de deux variables également appelée produit logique

Notation : $S = X \cdot Y$

Table de vérité

X	Y	$S = X \cdot Y$
0	0	0
0	1	0
1	0	0
1	1	1

$S = X \cdot Y$



La fonction ET ne vaut 1 que si toutes les variables valent 1

4.2.4. Autres opérateurs logiques

4.2.4.1. Circuits NAND et NOR

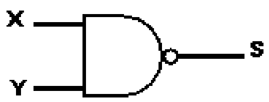
NOR : C'est une fonction de deux variables également appelée complément d'un produit logique

Notation : $S = \overline{X \cdot Y}$

Table de vérité

X	Y	$S = \overline{X \cdot Y}$
0	0	1
0	1	1
1	0	1
1	1	0

$S = \overline{X \cdot Y}$



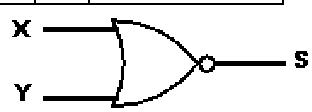
NOR : C'est une fonction de deux variables également appelée complément d'une somme logique.

Notation : $S = \overline{X + Y}$

Table de vérité

X	Y	$S = \overline{X + Y}$
0	0	1
0	1	0
1	0	0
1	1	0

$S = \overline{X + Y}$



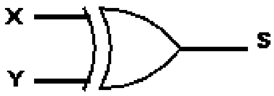
4.2.4.2. Ou exclusif et NON-OU exclusif

- **Ou exclusif**

Table de vérité

X	Y	$S = X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

$$S = X \oplus Y = \overline{X}Y + X\overline{Y}$$

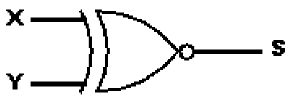


- **NON-OU exclusif – XNOR**

Table de vérité

X	Y	$S = X \otimes Y$
0	0	1
0	1	0
1	0	0
1	1	1

$$S = X \otimes Y = \overline{X \oplus Y} = \overline{X}\overline{Y} + XY$$



4.2.4.3. Implication

4.3. Théorèmes et propriétés de l'algèbre de Boole

Propriétés des opérateurs logiques

- **Associativité :** l'ordre des opérations n'est pas important s'ils ont la même priorité

$$(X + Y) + Z = X + (Y + Z) = X + Y + Z$$
$$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z) = X \cdot Y \cdot Z$$

- **Commutativité :** l'ordre de la variable n'est pas important. Commencer la formule par A ou par B revient au même.

$$X + Y = Y + X$$
$$X \cdot Y = Y \cdot X$$

- **Distributivité :** Cette loi gère l'ouverture des parenthèses. Pour aller de la distributivité de l'addition par rapport à la multiplication vers la distributivité de la multiplication par rapport à l'addition, il suffit de changer les « · » par les « + » et vise versa.

$$X(Y + Z) = XY + XZ$$
$$X + (YZ) = (X + Y)(X + Z)$$

• **Idempotence :** La variable additionnée ou multipliée par elle garde toujours sa valeur.

$$X + X + X + \ldots\ldots\ldots + X = X$$
$$X \cdot X \cdot X \ldots\ldots\ldots\ldots \cdot X = X$$

• **L'identité :** La variable additionnée par 0 ou multiplié par 1 reste inchangé, identique à elle-même.

$$X + 0 = X$$
$$X \cdot 1 = X$$

• **Absorption :** A absorbe B et reste A

$$X + XY = X$$
$$X(X + Y) = X$$

• **Éléments neutres :** Une variable multipliée par 0 donne 0 et additionnée par 1 donne 1

$$X \cdot 0 = 0$$
$$X + 1 = 1$$

• **La double négation :** La négation de la négation donne la variable $\overline{\overline{X}} = X$

• **Complémentarité :** La variable additionnée à son complément donne 1, multiplié à complément donne 0

$$X \cdot \overline{X} = 0$$
$$X + \overline{X} = 1$$

Théorème de De Morgan

Auguste (ou Augustus) De Morgan (27 juin 1806 à Madurai (Tamil Nadu) 18 mars 1871) est un mathématicien et logicien britannique, né en Inde. Il est le fondateur avec Boole de la logique moderne ; il a notamment formulé les lois de De Morgan qui sont à la base des systèmes logiques électroniques et informatiques. D'abord, il a introduit le terme induction mathématique et il donne le concept strict. Enfin, il a officialisé pour la première algèbre relationnelle.

C'est une des propriétés les plus importantes des fonctions logiques. Elle repose sur la remarque suivante :

Les relations caractéristiques des lois ET et OU sont invariantes dans leur ensemble lors de la transformation $+ \rightarrow \cdot$, $\cdot \rightarrow +$, $X \rightarrow \overline{X}$, $\overline{X} \rightarrow X$.

Le théorème de De Morgan est symbolisé par [21] :

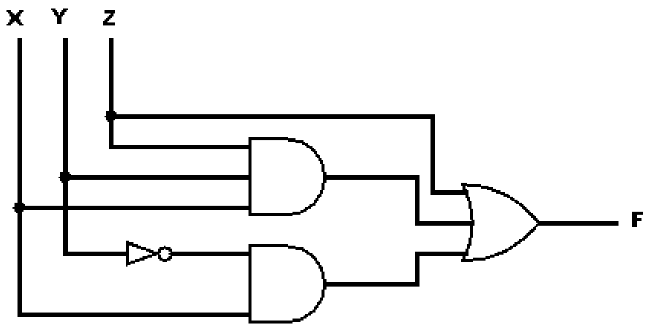
$$\begin{cases} \overline{X + Y} = \overline{X} \cdot \overline{Y} \\ \overline{X \cdot Y} = \overline{X} + \overline{Y} \end{cases}$$

Mais il est très général et porte sur toutes relations. Ainsi le complément d'une fonction logique sera obtenu en remplaçant les variables par leur complément, les signes + par des $\cdot$ et les signes $\cdot$ par des +.

4.4. Représentation des fonctions logiques

Les différentes fonctions booliennes sont décrites sous plusieurs formes :

- Une représentation logique (symbole logique= logigramme)

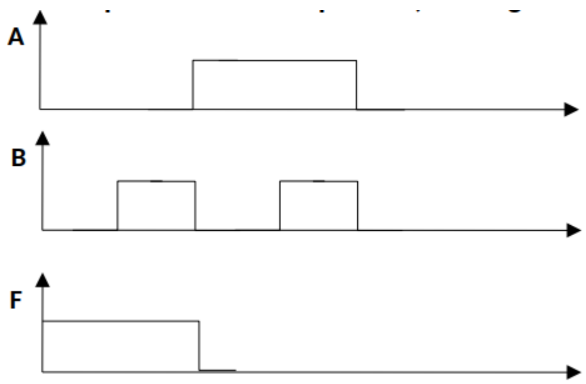


- Une représentation arithmétique (table de vérité)

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

- Une représentation temporelle (chronogramme)

C'est la représentation de la fonction logique en fonction du temps pour diverses valeurs des variables d'entrée [22].



- Une représentation algébrique (équation algébrique)

$$F(X,Y,Z) = XYZ + X\overline{Y} + Z$$

4.5. Table de vérité d'une fonction logique

Une table de vérité définit les relations entrées/sorties en faisant la liste de toutes les possibilités, une ligne à la fois dans la table.

$$F(X,Y,Z) = XYZ + X\overline{Y} + Z$$

1ère méthode :

X	Y	Z	XYZ	$\overline{Y}$	$X\overline{Y}$		F
0	0	0	0	1	0		0
0	0	1	0	1	0		1
0	1	0	0	0	0		0
0	1	1	0	0	0		1
1	0	0	0	1	1		1
1	0	1	0	1	1		1
1	1	0	0	0	0		0
1	1	1	1	0	0		1

2ème méthode :

$$F(X,Y,Z) = XYZ + X\overline{Y} + Z$$
$$= 111 + 10\_ + \_\,\_1$$

X	Y	Z		F
0	0	0		0
0	0	1		1
0	1	0		0
0	1	1		1
1	0	0		1
1	0	1		1
1	1	0		0
1	1	1		1

4.6. Les formes canoniques d'une fonction logique

Une forme est dite canonique quand toutes les variables constituant le mot d'entrée apparaissent dans les termes exprimant la valeur de la fonction. Il existe deux formes canoniques pour une fonction booléenne donnée, appelée 1ère forme canonique et 2ème forme canonique [23]:

- ⇒ La première forme canonique disjonctive : FND : mintermes = somme des produits
- ⇒ La deuxième forme canonique conjonctive : FNC : maxtermes = produits des sommes

Exemple : $F(X,Y,Z) = XYZ + X\overline{Y} + Z$

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

$$F_{FND}(X,Y,Z) = \overline{X}\overline{Y}Z + \overline{X}YZ + X\overline{Y}\overline{Z} + X\overline{Y}Z + XYZ$$

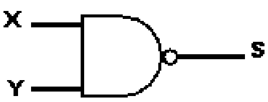
$$F_{FNC}(X,Y,Z) = (X+Y+Z)(X+\overline{Y}+Z)(\overline{X}+\overline{Y}+Z)$$

2ème méthode pour la forme FND

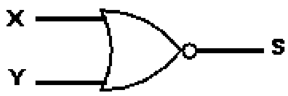
$$\begin{aligned} F(X,Y,Z) &= XYZ + X\overline{Y} + Z \\ &= XYZ + X\overline{Y}(Z+\overline{Z}) + (X+\overline{X})(Y+\overline{Y})Z \\ &= XYZ + X\overline{Y}Z + X\overline{Y}\overline{Z} + XYZ + X\overline{Y}Z + \overline{X}YZ + \overline{X}\overline{Y}Z \\ &= XYZ + X\overline{Y}Z + X\overline{Y}\overline{Z} + \overline{X}YZ + \overline{X}\overline{Y}Z \end{aligned}$$

4.7. Expression d'une fonction logique avec des circuits NANDs ou NOR exclusivement

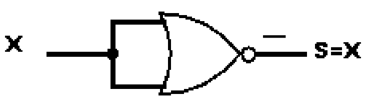
NAND (NON ET)

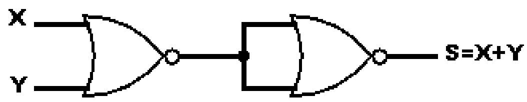


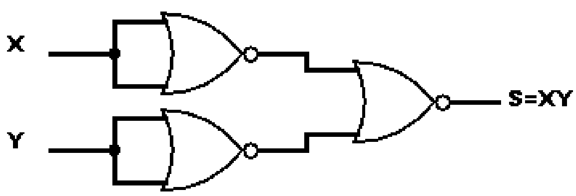
NOR (NON OU)



Réalisation des fonctions NON, OU et ET en utilisant uniquement des portes NOR ou NAND







4.8. Simplification d'une fonction logique

La simplification d'une fonction consiste à obtenir son expression la plus compacte possible afin de minimiser le nombre d'opérateurs logiques nécessaires à sa réalisation. L'objectif de la simplification des fonctions logiques est de :

– réduire le nombre de termes dans une fonction

– et de réduire le nombre de variables dans un terme

• Cela afin de réduire le nombre de portes logiques utilisées et de réduire le coût du circuit

• Plusieurs méthodes existent pour la simplification, On distingue trois méthodes de simplification :

– La Méthode algébrique

– Les Méthodes graphiques : (ex : tableaux de Karnaugh)

– La Méthode de Quine Mc-Cluskey.

4.8.1. Simplification algébrique

Les théorèmes de l'algèbre de Boole étudiés précédemment peuvent nous être utiles pour simplifier une expression logique. Exemple :

$$\begin{aligned} F(X,Y,Z) &= XY\overline{Z} + X\overline{Y}Z + XYZ \\ &= XY\left(\overline{Z} + Z\right) + X\overline{Y}Z \\ &= XY + X\overline{Y}Z \\ &= X\left(Y + \overline{Y}Z\right) \\ &= X\left(\left(Y + \overline{Y}\right)\left(Y + Z\right)\right) \\ &= XY + XZ \end{aligned}$$

$$\begin{aligned} G(X,Y) &= \overline{X}\overline{Y} + X\overline{Y} + \overline{X}Y \\ &= \overline{X}\overline{Y} + \overline{X}\overline{Y} + X\overline{Y} + \overline{X}Y \\ &= \overline{X}\left(\overline{Y} + Y\right) + \overline{Y}\left(X + \overline{X}\right) \\ &= \overline{X} + \overline{Y} \end{aligned}$$

4.8.2. Simplification par tableau de Karnaugh (Méthode graphique) :

Maurice Karnaugh, né le 4 octobre 1924 à New York et mort le 8 novembre 2022, est un ingénieure en télécommunications américain américain. Il a développé la table de Karnaugh aux laboratoires Bell en 1953. De **nationalité** américaine, c'est un spécialiste de physique et mathématique logique (**Algèbre de Boole**).

Activités: Physicien, mathématicien, professeur d'université, ingénieur, informaticien. **A travaillé pour** : IBM, École d'ingénierie Tandon de l'université de New York. **Membre de** :Institute of Electrical and Electronics Engineers

Cette méthode repose sur l'utilisation des tableaux de Karnaugh.

a. Tableau de Karnaugh

La table de KARNAUGH est un outil de simplification des expressions logiques [24]. C'est une table de vérité à deux dimensions. L'intersection d'une ligne avec une colonne constitue une case. Les variables sont divisées en deux groupes : des variables lignes et des variables colonnes. Le tableau est construit tel que deux cases adjacentes correspondent à deux combinaisons adjacentes. Voilà des exemples de tableaux de Karnaugh représentants 2, 3, 4 ou 5 variables logiques d'entrée.

b. Description de la table de Karnaugh

X / Y	0	1
0		
1		

Tableau à 2 variables

Z / XY	**00**	**01**	**11**	**10**
0				
1				

X / YZ	0	1
00		
01		
11		
01		

Tableau à 3 variables

XY / ZW	00	01	11	10
00				
01				
11				
10				

Tableau à 4 variables

XYZ / WT	000	001	011	010	110	111	101	100
00								
01								
11								
10								

Tableau à 5 variables

- On utilise obligatoirement le code Gray

b. Règles de regroupement

1. On ne regroupe que les points vrais de la fonction qui sont adjacents (contenant des 1).
2. On ne peut regrouper que 2^k cases adjacentes (nombre pair).
3. Un point vrai peut être utilisé plusieurs fois dans des groupements différents.
4. On doit utiliser au moins une fois tous les points vrais de la fonction.
5. On doit rechercher les groupements les plus grands possible pour minimiser le nombre des variables.
6. Si une fonction est exprimée avec N variables, un regroupement de 2^k cases conduit à un terme produit simplifié de (N–k) variables. Les k variables éliminés sont celle qui ont varié dans le regroupement.
7. La fonction simplifiée est la réunion des différents regroupements.
8. Tous les « 1 » doivent être contenus dans au moins un groupement.
9. Une même case peut être utilisée pour des groupements différents.

Chaque groupement obtenu représente un impliquant premier. Un impliquant premier qui contient au moins 1 ne pouvant être inclus dans aucun autre impliquant premier est dit impliquant premier essentiel. Pour obtenir la forme minimale, on choisit en premier lieu les impliquants premiers essentiels. Ensuite, on choisit parmi les impliquants premiers restants ceux qui sont nécessaires pour couvrir complètement la fonction originale. Si la forme minimale ne contient que des impliquants premiers essentiels, alors elle est unique.

c. Principe de simplification

- Réaliser des groupements de '1' adjacents, dans l'ordre, par 16, 8, 4 ,2 ou 1. Il faut toujours s'arranger à regrouper le maximum de '1' pour diminuer la taille des termes.

- Lorsqu'il ne reste plus de '1' isolé, les regroupements sont terminés.
- L'équation simplifiée est déduite de ces groupements
- Il est également possible et c'est parfois facile de regrouper les états 0 de la fonction F et de considérer que nous étudions F

Exemples :

C \ AB	00	01	11	10
0	1	1	1	1
1	1	0	0	1

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

BC \ A	0	1
00	1	1
01	0	1
11	0	0
10	1	1

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

CD \ AB	00	01	11	10
00	0	1	1	0
01	1	1	1	1
11	1	1	1	1
10	0	1	1	0

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

CD \ AB	00	01	11	10
00	1	0	0	1
01	1	1	1	1
11	1	1	1	1
10	1	0	0	1

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

CDE \ AB	00	01	11	10
000	0	0	0	1
001	1	1	0	1
011	1	1	0	0
010	1	1	0	0
110	0	0	0	0
111	0	0	0	0
101	1	1	0	0
100	1	1	0	0

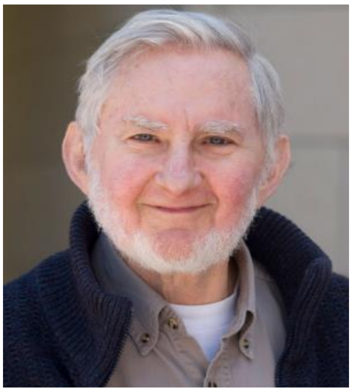
$$F(A,B,C,D,E)=\overline{A}C\overline{D}+\overline{A}\,\overline{C}E+\overline{A}\,\overline{C}D+A\overline{B}\,\overline{C}\,\overline{D}$$

ABC \ DE	000	001	011	010	110	111	101	100
00	1	1	1	1	0	0	0	1
01	1	1	1	1	0	0	0	1
11	0	1	1	0	0	0	0	1
10	0	1	1	0	0	0	0	1

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

4.8.3. Simplification par la méthode de Quine Mc Cluskey

Plus lourde à appliquer que celle de Karnaugh, cette méthode n'est en générale employée que quand le nombre des variables est important (> 5). Elle présente l'intérêt d'être systématique et donc programmable.



Edward Joseph McCluskey, né le 16 octobre 1929 à New York, et mort le 13 février 20161 est professeur émérite à l'université Stanford. Il est un pionnier dans le domaine de l'électrotechnique. McCluskey a travaillé sur les systèmes de commutation électronique aux Bell Telephone Laboratories de 1955 à 1959.

McCluskey a servi de mentor plus de 70 étudiants au doctorat et a une famille grandissante des universitaires « petits-enfants ». Il dispose également d'une collection de chapeaux. McCluskey est décédé le 13 Février 2016 à New York (États-Unis).

Méthode :

La technique de Quine-McCluskey s'applique de la même manière aux expressions disjonctives qu'aux expressions conjonctives. Nous nous concentrerons dans ce qui suit sur le cas des expressions disjonctives [25].

L'algorithme s'exprime ainsi :

1. Exprimer la fonction sous forme canonique disjonctive ;
2. Exprimer les minterms sous forme binaire ;
3. Grouper les termes selon leur poids ;
4. Unir les termes deux à deux ;
5. Répéter l'étape (4) autant de fois que nécessaire ;
6. Identifier les impliquants premiers ;
7. Identifier les impliquants premiers essentiels ;
8. Si la fonction est entièrement exprimée par ses impliquants premiers essentiels, arrêter ;
9. Sinon, choisir, les impliquants premiers non essentiels permettant une couverture complète.

Exemple :

Considérons la fonction suivante exprimée par sa table de vérité :

A	B	C	D	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
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	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1		0

1. **Exprimer la fonction sous forme canonique disjonctive**

$$F(A,B,C,D) = \overline{A}\,\overline{B}C\overline{D} + \overline{A}\,\overline{B}CD + \overline{A}BC\overline{D} + A\overline{B}\,\overline{C}\,\overline{D} + A\overline{B}\,\overline{C}D + A\overline{B}C\overline{D} + A\overline{B}CD + AB\overline{C}\,\overline{D} + AB\overline{C}D + ABC\overline{D}$$

2. **Exprimer les minterms sous forme binaire**

$$F(A,B,C,D) = 0010+0011+0110+1000+1001+1010+1011+1100+1101+1110$$

Pour chaque minterm, on remplace les variables par leur équivalent binaire. Si la variable est inversée, on pose 0, si elle ne l'est pas, on pose 1. Par exemple :

$\overline{A}\,\overline{B}C\overline{D}$ $\Rightarrow$0010

$\overline{A}\,\overline{B}CD$ $\Rightarrow$0011

$\overline{A}BC\overline{D}$ $\Rightarrow$0110

$A\overline{B}\,\overline{C}\,\overline{D}$ $\Rightarrow$1000

$A\overline{B}\,\overline{C}D$ $\Rightarrow$1001

3. **Grouper les termes selon leurs poids**

Le mot poids renvoie au nombre de 1 contenus dans la forme binaire des minterms.

1.	0010	poids 1
2.	1000	
3.	0011	
4.	0110	
5.	1001	
6.	1010	poids 2
7.	1100	
8.	1011	
9.	1101	poids 3
10.	1110	

4. Unir les termes deux à deux

Cette procédure est clé. Il s'agit de générer une nouvelle colonne de termes en réunissant deux à deux les termes de la colonne précédente.

1.	0010	✓	001x	généré en combinant 1 et 3
2.	1000	✓	0x10	généré en combinant 1 et 4
3.	0011	✓	x010	généré en combinant 1 et 6
4.	0110	✓	100x	généré en combinant 2 et 5
5.	1001	✓	10x0	généré en combinant 2 et 6
6.	1010	✓	1x00	généré en combinant 2 et 7
7.	1100	✓	x011	généré en combinant 3 et 8
8.	1011	✓	x110	généré en combinant 4 et 10
9.	1101	✓	10x1	généré en combinant 5 et 8
10.	1110	✓	1x01	généré en combinant 5 et 9
			101x	généré en combinant 6 et 8
			1x10	généré en combinant 6 et 10
			110x	généré en combinant 7 et 9
			11x0	généré en combinant 7 et 10

5. Répéter l'étape (4) autant de fois que nécessaire

On réunit les termes de la nouvelle colonne comme nous l'avions fait à l'étape (4). Pour que deux termes soient unis, en plus des conditions précédentes, il faut que les x soient au même endroit.

1.	001x	✓	x01x	généré en combinant 1 et 11
2.	0x10	✓	xx10	généré en combinant 2 et 12
3.	x010	✓	x01x	généré en combinant 3 et 7
4.	100x	✓	xx10	généré en combinant 3 et 8
5.	10x0	✓	10xx	généré en combinant 4 et 11
6.	1x00	✓	1x0x	généré en combinant 4 et 13
7.	x011	✓	10xx	généré en combinant 5 et 9
8.	x110	✓	1xx0	généré en combinant 5 et 14
9.	10x1	✓	1x0x	généré en combinant 6 et 10
10.	1x01	✓	1xx0	généré en combinant 6 et 12
11.	101x	✓		
12.	1x10	✓		
13.	110x	✓		
14.	11x0	✓		

Si le même terme est généré plusieurs fois, on ne garde qu'une seule copie. On répète alors l'étape (4) :

x01x		
xx10		
~~x01x~~		x01x
~~xx10~~		xx10
10xx	⇒	10xx
1x0x		1x0x
~~10xx~~		1xx0
1xx0		
~~1x0x~~		
~~1xx0~~		

- Impossible de combiner aucun des termes car il n'y a pas de termes possédant les x au même endroit.
- Ne pouvant plus réunir aucune paire de termes, l'étape (5) est terminée.

6. Identifier les impliquants premiers

On reprend ici l'ensemble des étapes effectuées :

0010 ✓	001x ✓	x01x
1000 ✓	0x10 ✓	xx10
0011 ✓	x010 ✓	10xx
0110 ✓	100x ✓	1x0x
1001 ✓	10x0 ✓	1xx0
1010 ✓	1x00 ✓	
1100 ✓	x011 ✓	
1011 ✓	x110 ✓	
1101 ✓	10x1 ✓	
1110 ✓	1x01 ✓	
	101x ✓	
	1x10 ✓	
	110x ✓	
	11x0 ✓	

Tous les termes qui ne sont pas marqués (✓) sont des impliquants premiers : x01x, xx10, 10xx, 1x0x, 1xx0. Cette écriture binaire se lit $\bar{B}C, C\bar{D}, A\bar{B}, A\bar{C}$ *et* $A\bar{D}$ respectivement.

7. Identifier les impliquants premiers essentiels

Pour identifier les impliquants premiers essentiels, on utilise un tableau tel que sur les lignes, on dispose tous les impliquants premiers identifiés et que sur les colonnes, on pose les minterms de la fonction. On procède alors par identification :

	0010	0011	0110	1000	1001	1010	1011	1100	1101	1110
x01x	✓	✓				✓	✓			
xx10	✓		✓			✓				✓
10xx				✓	✓	✓	✓			
1x0x				✓	✓			✓	✓	
1xx0				✓		✓		✓		✓

Un impliquant premier est essentiel s'il est le seul à être associé à au moins un minterm. Ainsi, un minterm appartient à un impliquant premier essentiel si sa colonne ne comporte qu'une seule astérisque (*).

	0010	0011	0110	1000	1001	1010	1011	1100	1101	1110
x01x	✓	(✓)				✓	✓			
xx10	✓		(✓)			✓				✓
10xx				✓	✓	✓	✓			
1x0x				✓	✓			✓	(✓)	
1xx0				✓		✓		✓		✓

Un impliquant premier est essentiel s'il comporte au moins une étoile entre parenthèses. Dans notre exemple, les impliquants premiers essentiels sont : x01x, xx10 et 1x0x.

8. Vérifier si la fonction est entièrement exprimée par ses impliquants essentiels

Pour ce faire, il faut refaire le tableau (il est aussi possible d'effectuer cette étape sur le même tableau précédent) en ne gardant que les impliquants essentiels :

	0010	0011	0110	1000	1001	1010	1011	1100	1101	1110
x01x	✓	(✓)				✓	✓			
xx10	✓		(✓)			✓				✓
1x0x				✓	✓			✓	(✓)	

Pour que la fonction soit entièrement décrite par ses impliquant essentiels, il faut que chaque colonne comporte au moins une étoile. C'est le cas de notre exemple. La technique de Quine-McCluskey s'arrête à ce point :

$$\begin{aligned} \mathrm{F}(\mathrm{A},\mathrm{B},\mathrm{C},\mathrm{D}) &= \mathrm{x01x} + \mathrm{xx10} + \mathrm{1x0x} \\ &= \overline{\mathrm{B}}\mathrm{C} + \mathrm{C}\overline{\mathrm{D}} + \mathrm{A}\overline{\mathrm{C}} \end{aligned}$$

- **Deuxième exemple sur la méthode de Quine Mc-Cluskey**

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

- **Représentation de la fonction sous la forme canonique disjonctive (FND)**

$$\begin{aligned} F(A,B,C,D) &= \overline{A}\overline{B}CD + A\overline{C}D + ABD + AB\overline{D} + \overline{A}C \\ &= \overline{A}\overline{B}CD + A(B+\overline{B})\overline{C}D + AB(C+\overline{C})D + AB(C+\overline{C})\overline{D} + \overline{A}(B+\overline{B})C(D+\overline{D}) \\ &= \overline{A}\overline{B}CD + AB\overline{C}D + A\overline{B}\overline{C}\,D + ABCD + AB\overline{C}D + ABC\overline{D} + AB\overline{C}\,\overline{D} + \overline{A}BCD + \overline{A}BC\overline{D} + \overline{A}\overline{B}CD + \overline{A}\overline{B}C\overline{D} \\ &= \overline{A}\overline{B}C\overline{D} + \overline{A}\overline{B}CD + \overline{A}BC\overline{D} + \overline{A}BCD + A\overline{B}\overline{C}\,D + AB\overline{C}\,\overline{D} + AB\overline{C}D + ABC\overline{D} + ABCD \end{aligned}$$

- **Transformation en nombres binaires**

$$\begin{aligned} F(A,B,C,D) &= \overline{A}\overline{B}C\overline{D} + \overline{A}\overline{B}CD + \overline{A}BC\overline{D} + \overline{A}BCD + A\overline{B}\overline{C}\,D + AB\overline{C}\,\overline{D} + AB\overline{C}D + ABC\overline{D} + ABCD \\ &= 0010 + 0011 + 0110 + 0111 + 1001 + 1100 + 1101 + 1110 + 1111 \end{aligned}$$

- **Classification**

1.	0010	poids 1
2.	0011	
3.	0110	poids 2
3.	1001	
4.	1100	
5.	0111	
6.	1101	poids 3
7.	1110	
8.	1111	poids 4

➦ **Comparaisons et Identification des impliquants**

0010	✓	001x	✓	0x1x
0011	✓	0x10	✓	~~0x1x~~
0110	✓	0x11	✓	x11x
1001	✓	011x	✓	~~x11x~~
1100	✓	x110	✓	11xx
0111	✓	1x01		~~11xx~~
1101	✓	110x	✓	
1110	✓	11x0	✓	
1111	✓	x111	✓	
		11x1	✓	
		111x	✓	

➦ **Identification des impliquants essentiels**

	0010	0011	0110	0111	1001	1100	1101	1110	1111
1x01					(✓)		✓		
0x1x	(✓)	(✓)	✓	✓					
x11x			✓	✓				✓	✓
11xx						(✓)	✓	✓	✓

Les impliquants essentiels sont : 1x01, 0x1x et 11xx. Puisqu'ils sont suffisants pour représenter toutes les solutions, la fonction simplifiée est :

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

4.9. Fonctions incomplètement définies

Dans des cas pratiques, certaines combinaisons de variables n'ont aucun sens physique et n'apparaissent jamais dans la réalité. Il est donc inutile de spécifier la valeur de la fonction pour de telles combinaisons.

Dans ce cas, le concepteur peut à sa convenance attribuer à ces cases la valeur 0 ou 1 de manière à obtenir le maximum de groupements.

•Pour les cas impossibles ou interdites il faut mettre un X dans la table de vérité.

•Les cas impossibles sont représentés aussi par des X dans la table de Karnaugh.

Il est possible d'utiliser les X dans des regroupements :

–Soit les prendre comme étant des 1

–Ou les prendre comme étant des 0

•Il ne faut pas former des regroupements qui contiennent uniquement des X

Table de vérité

X	Y	Z	W	F
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
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	X
1	1	0	0	X
1	1	0	1	X
1	1	1	0	0
1	1	1	1	1

Simplification de F en utilisant la table de Karnaugh

XY ZW	00	01	11	10
00	0	0	X	1
01	1	1	X	1
11	0	0	1	X
10	1	1	0	0

$$F(X,Y,Z,W) = \overline{Z}W + X\overline{Z} + XW + \overline{X}Z\overline{W}$$