Cours Systèmes Logiques

Elaborè par MdmeTounsi Lamia

Département d’informatique et de communication

A.U 2023- 2024

- **Plan du cours**
  - **Introduction**
  - **Algèbre de Boole**
  - **Représentation des fonctions logiques**
    - Formes technologiques
    - Logigrammes
    - Chronogrammes
  - **Simplification des fonctions logiques**
  - **Circuits combinatoires**
  - **Circuits séquentielles**

Introduction

- Un circuit électrique, pneumatique, hydraulique peut avoir 2 états logiques. Ces états peuvent prendre les valeurs 1 ou 0.
  C'est ce que l'on appelle **état logique**. Ces états sont fonctions de l'état des composants en série dans le circuit

– ***État 0 :*** Les actionneurs tels que : moteurs, vérins sont à l'état 0 lorsqu'ils ne sont pas alimentés. Le circuit est alors ouvert. Pour un circuit pneumatique ceci correspond à une absence de pression. Pour un circuit électrique cela correspond à une absence de différence de potentiel entre les bornes du circuit.

Système Logique

-
  - ***État 1 :*** Les actionneurs sont à l'état 1 lorsqu'ils sont alimentés. Pour un circuit pneumatique ou hydraulique ceci correspond à une pression d'air ou d'huile dans le circuit. Pour un circuit électrique cela correspond à une différence de potentiel entre les bornes du circuit

- Il existe 2 types de logique :
  – la logique « *positive* » : le oui est représenté par un 1, et le non par un 0.
  – la logique « *négative* » : le oui est représenté par un 0, et le non par un 1.

Système Logique

- **On dispose pour traiter l'information :**

  – d'un **outil mathématique** : **l'algèbre de Boole**, son rôle est de mettre en équation le fonctionnement d'un système, et de le simplifier en vue de sa réalisation physique.

  – d'un **outil physique** : les portes logiques NON -NO-, ET -AND-, OU -OR-, ..., fonctions de base « pré-câblées » permettant la fabrication du circuit électrique, pneumatique, ou hydraulique

Chapitre1: Algèbre de Boole <<<<<<<

- Pour étudier les fonctions de variables binaires on utilise l algèbre développée au XIXème siècle par le mathématicien anglais : **Georges Boole** le créateur de la logique moderne.

- Dans ce chapitre nous nous proposons de présenter les fonctions de base de l'algèbre booléenne ainsi que leurs représentations symboliques en électronique

- L'algèbre de Boole concerne la logique des systèmes binaires:
  - État logique (binaire)
    - **Élément nul : valeur binaire 0 (faux, non, bas, ouvert, éteint, vide)**
    - **Élément unité : valeur binaire 1 (vrai, oui, haut, fermé, allumé, plein)**
  - Variable logique
    - **Grandeur représentée par un symbole (lettre ou signe) qui peut prendre 2 états logiques dans le cadre de l'algèbre de Boole**.

A- Définition :

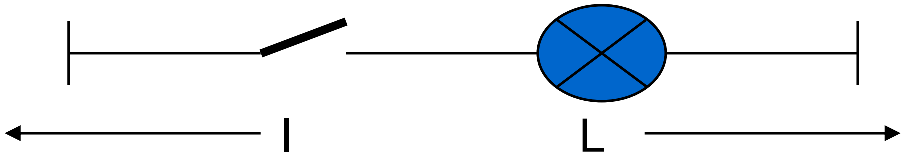
On appelle **fonction logique** (ou booléenne) une fonction définie sur 2^n combinaisons de n variables logiques.

– Une fonction logique est donc une fonction de n variables logiques,

– Une fonction logique peut prendre en sortie 2 valeurs notées 0 et 1.

<u>*système binaire :*</u>

variable logique
ouvert, fermé

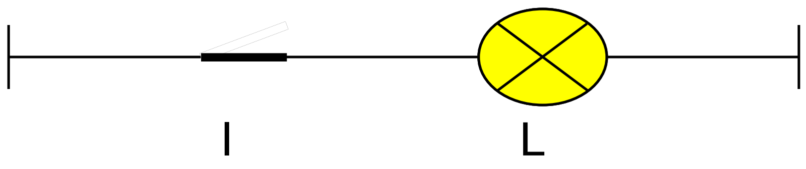


fonction logique
éteint, allumé

1.1 Fonction Logique

Table de vérité

- Soit F, une fonction de *n* variables
- La table de vérité de F est un tableau de *n+1* colonnes et 2^n lignes dans lequel apparaissent toutes les combinaisons d’entrées associées à la valeur correspondante de la fonction.



I	L	
0	0	→ Point faux
1	1	→ Point vrai

B- Fonctions Logiques de bases de l’algèbre de Boole :

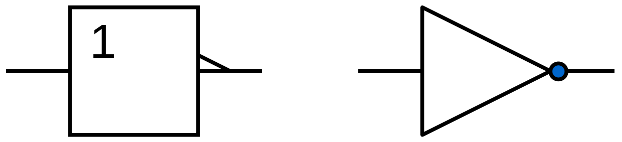
- **Fonction NON** : fonction complément ou fonction inverse. C’est une fonction f d’une variable x telle que :0

$$\boxed{F(x) = \bar{x}}$$

table de vérité :

x	f(x)
0	1
1	0

symbole :

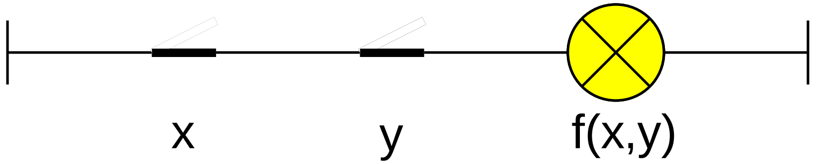


norme CEI   norme IEEE

- **Fonction ET (AND)** : produit logique. C'est une fonction f de plusieurs variables équivalente à <u>l'intersection</u> en théorie des ensembles. Elle prend la valeur 1 si toutes les variables sont simultanément égales à 1. Soient x et y, deux variables booléennes, f(x,y) s'écrit:

$$\boxed{f(x,y) = xy}$$

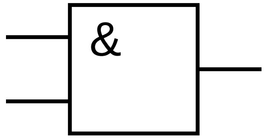
<u>système binaire :</u>
Interrupteurs branchés en série



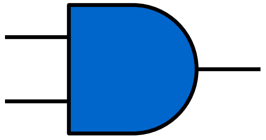
<u>table de vérité :</u>

x	y	f(x,y)
0	0	0
0	1	0
1	0	0
1	1	1

<u>symbole :</u>



norme CEI



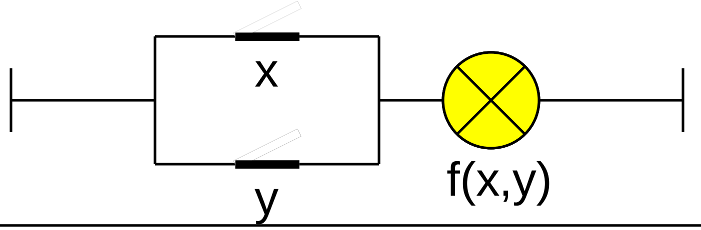
norme IEEE

1.1Fonction Logique

- **Fonction OU (OR)** : somme logique . C'est une fonction f de plusieurs variables équivalente à <u>l'union</u> en théorie des ensembles. Elle prend la valeur 1 si au moins une variable est égale à 1. Soient x et y, deux variables booléennes, f(x,y) s'écrit :

$$\boxed{f(x,y)= x+y}$$

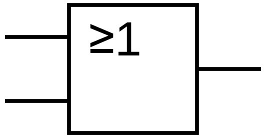
<u>système binaire :</u>
Interrupteurs branchés en parallèle



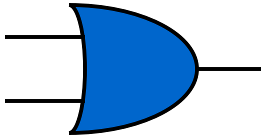
<u>table de vérité :</u>

x	y	f(x,y)
0	0	0
0	1	1
1	0	1
1	1	1

<u>symbole :</u>



norme CEI



norme IEEE

C- Propriétés de l'algebre de Boole

Fonctions	OU	ET	Commentaires
2 variables	A+B=B+A	A.B=B.A	Commutativité
3 variables	A+(B+C)=(A+B)+C =A+B+C	A.(B.C)=(A.B).C =A.B.C	Associativité
	A+B.C=(A+B).(A+C)	A.(B+C)=A.B+A.C	Distributivité

Fonctions	OU	ET	Commentaires
1 variable	A+A=A	A.A=A	Idempotence
	A+1=1	A.0=0	Elément absorbant
	A+0=A	A.1=A	Elément Neutre
	$A+\overline{A}=1$	$A.\overline{A}=0$	Complément
	$\overline{\overline{A}}=A$		Involution

- **Théorèmes**: Pour effectuer tout calcul Booléen, on utilise, en plus des propriétés, un ensemble de théorèmes

Théorèmes	OU	ET
De DEMORGAN	$\overline{A+B} = \overline{A}.\overline{B}$	$\overline{A.B} = \overline{A}+\overline{B}$
	Ce théorème peut être généralisé à plusieurs variables	
	$\overline{A+B+\dots+Z} = \overline{A}.\overline{B}.\dots.\overline{Z}$	$\overline{A.B.\dots.Z} = \overline{A}+\overline{B}+\dots+\overline{Z}$
D'absorption	$A+AB=A$	$A.(A+B)=A$
D'allègement	$A+\overline{A}B=A+B$	$A.(\overline{A}+B)=A.B$
	$A.B+\overline{A}C+BC=AB+\overline{A}C$	

D-Autres opérateurs logique :

- **OU Exclusif** (XOR) : elle prend la valeur 1 si et seulement si le nombre de variables égales à 1 est impair. Soient x et y, deux variables booléennes, f(x,y) s'écrit :

$$\boxed{f(x,y) = x \oplus y = \bar{x}y + x\bar{y}}$$

système binaire :

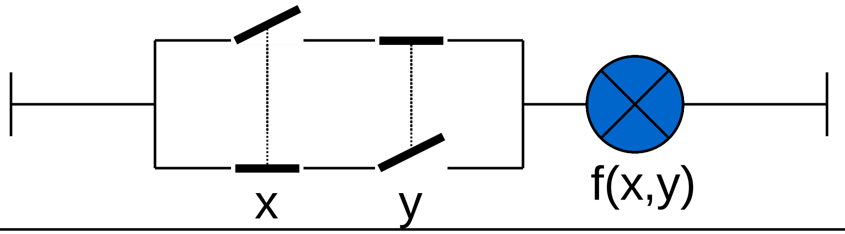
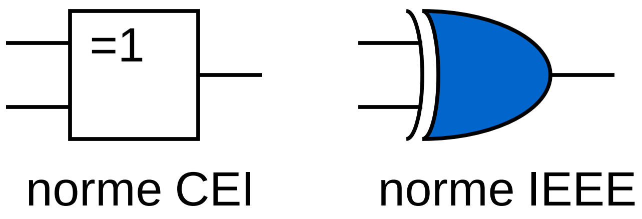


table de vérité :

x	y	f(x,y)
0	0	0
0	1	1
1	0	1
1	1	0

symbole :



- L'opérateur OU exclusif vérifie les propriétés suivantes :

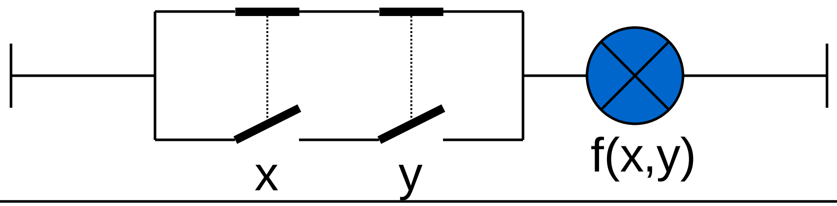
$x \oplus 0 = x$	$x \oplus 1 = \bar{x}$
$x \oplus x = 0$	$x \oplus \bar{x} = 1$
$x \oplus y = y \oplus x$	
$x \oplus (y \oplus z) = (x \oplus y) \oplus z$	
$\overline{x \oplus y} = x \oplus \bar{y} = \bar{x} \oplus y$	

1.1Fonction Logique

- **Coincidence ou identité** : elle prend la valeur 1 ssi le nombre de variables égales à 1 est pair. C'est la fonction complémentaire de la fonction Ouxlusive. Soient x et y, deux variables booléennes, f(x,y) s'écrit :

$$\boxed{f(x,y) = \bar{x}\bar{y} + xy = \overline{x \oplus y}}$$

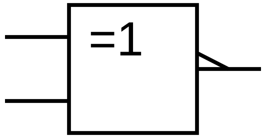
<u>système binaire :</u>



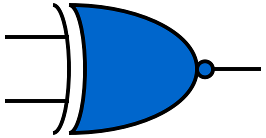
<u>table de vérité :</u>

x	y	$x \oplus y$	f(x,y)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

<u>symbole :</u>



norme CEI



norme IEEE

Fonction Logique

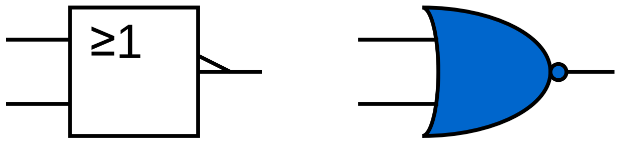
- **NON OU (NOR ou NI)** : elle prend la valeur 1 si toutes les variables sont simultanément égales à 0. C’est aussi un opérateur complet. Soient x et y, deux variables booléennes, f(x,y) s’écrit :

$$\boxed{f(x,y) = \overline{x+y}}$$

table de vérité :

x	y	x + y	f(x,y)
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

symbole :



norme CEI   norme IEEE

1.1Fonction Logique

- **NON ET (NAND ou ON)** : elle prend la valeur 1 si au moins une variable est égale à 0. C’est un opérateur complet car il permet de réaliser les trois opérateurs de base de l’algèbre de Boole. Soient x et y, deux variables booléennes, f(x,y) s’écrit :

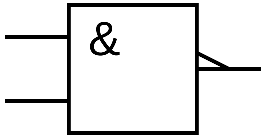
$$\boxed{f(x,y) = \overline{x\ y}}$$

table de vérité :

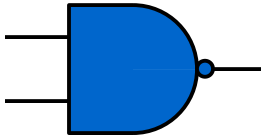
1

x	y	x · y	f(x,y)
0	0	0	1
0	1	0	0
1	0	0	0
1	1	1	0

symbole :



norme CEI



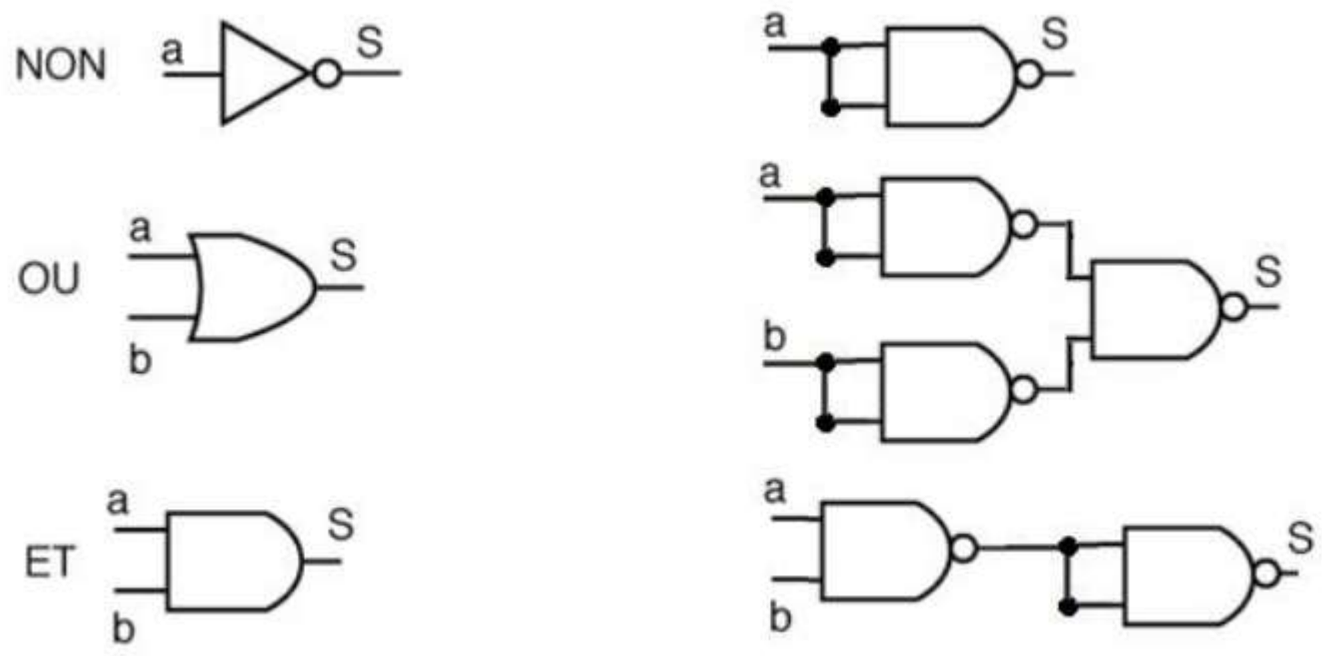
norme IEEE

- **<u>Exercice:</u>**

Les portes logiques NAND et NOR sont appelées universelles, car avec elles seules on peut réaliser toutes les autres portes logiques.

- **1-A** l'aide des portes NAND (à 2 entrées) uniquement réaliser les trois portes logiques de bases:NON, OU, ET

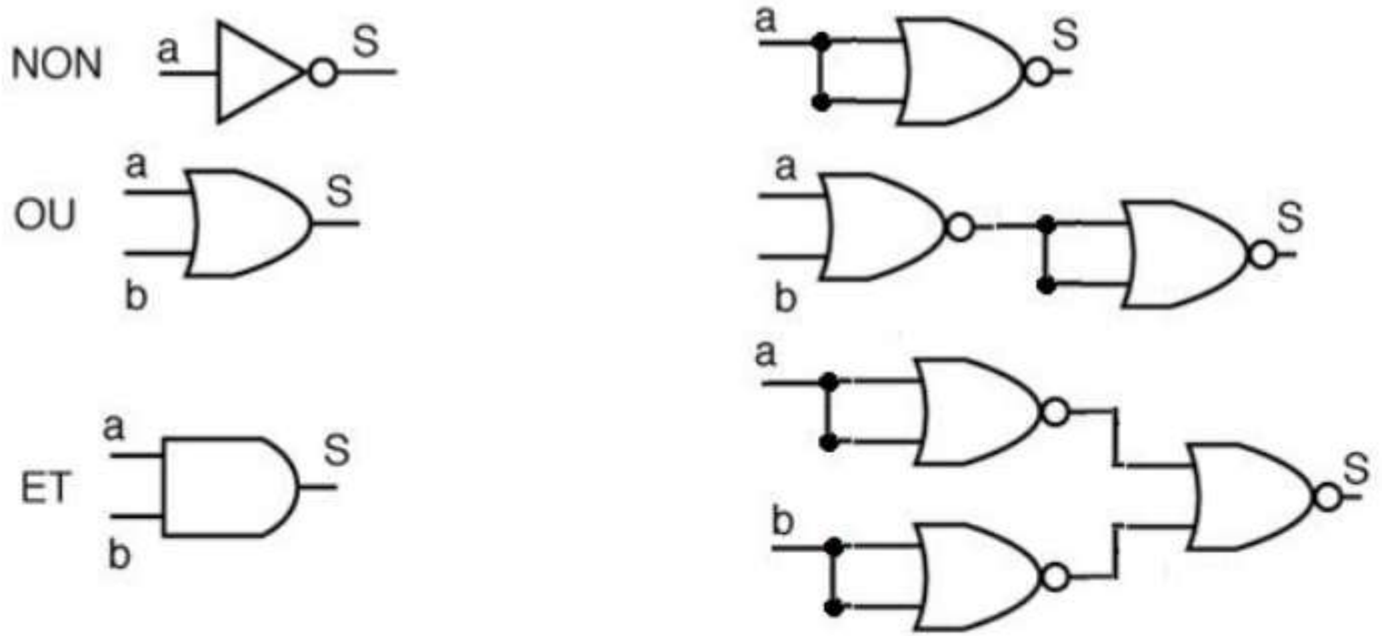
- **Solution**:
- 1-Construction des portes NON, OU, ET à l'aide de portes NAND



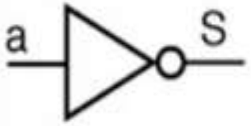
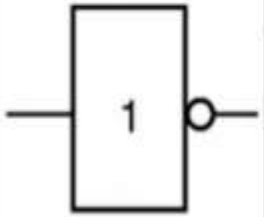
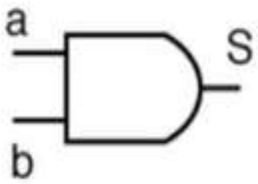
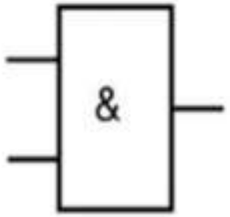
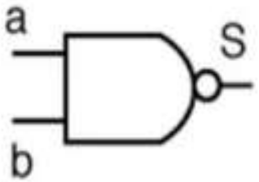
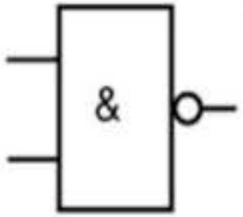
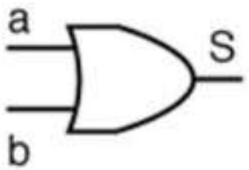
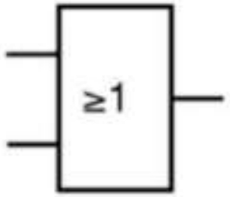
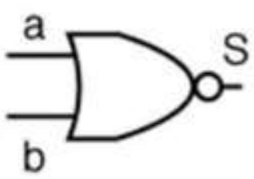
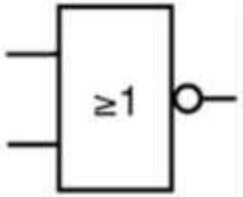
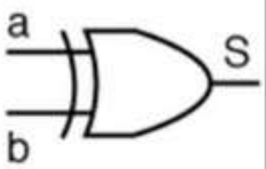
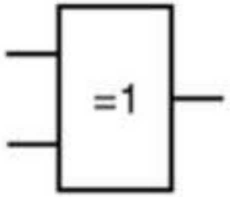
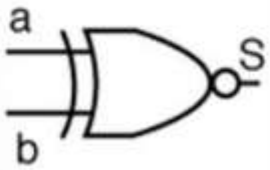
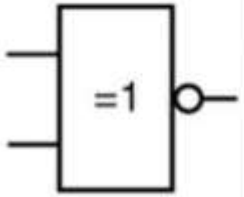
2-A l'aide des portes NOR uniquement réaliser les trois portes logiques de bases:NON, OU, ET

<u>Solution</u>:

2- Construction des portes NON, OU, ET à l'aide de portes NOR



Résumé des identités booléennes de base

FONCTION	SYMBOLES International	SYMBOLES Français	TABLES DE VERITE a	b	S
NON			0 1		1 0
ET			0 0 1 1	0 1 0 1	0 0 0 1
NAND			0 0 1 1	0 1 0 1	1 1 1 0
OU			0 0 1 1	0 1 0 1	0 1 1 1
NOR			0 0 1 1	0 1 0 1	1 0 0 0
OU Exclusif			0 0 1 1	0 1 0 1	0 1 1 0
NOR Exclusif			0 0 1 1	0 1 0 1	1 0 0

REPRESENTATION ET SIMPLIFICATION DES FONCTIONS LOGIQUES

OBJECTIFS

- Représentation algébrique d’une fonction logique
- Comprendre la simplification algébrique d’une fonction logique
- Simplification par la methode de Karnaugh

REPRESENTATION D’UNE FONCTION LOGIQUE

•Une fonction logique est une combinaison de variables binaires reliées par les opérateurs ET, OU et NON. Elle peut être représentée par une écriture algébrique ou une table de vérité ou un tableau de KARNAUGH ou un logigramme.

Représentation des fonctions logiques

– Convention d'écriture de la table de vérité :

Les variables a, b, et c représente un mot binaire $(a\ b\ c)_2$

f(a,b,c) est une fonction logique de 3 variables

Ordre binaire naturel

	a	b	c	f(a,b,c)
0	0	0	0	f0
1	0	0	1	f1
2	0	1	0	f2
3	0	1	1	f3
4	1	0	0	f4
5	1	0	1	f5
6	1	1	0	f6
7	1	1	1	f7

fi={0, 1}

$(N)_{10}$ est l'équivalent décimal du mot $(a\ b\ c)_2$ avec :
$(N)_{10} = 2^{0}$´ c + 2^{1}´ b + 2^{2}´ a (codage binaire) où :
- c représente le bit le moins significatif (LSB) ou bit de poids faible et
- a représente le bit le plus significatif (MSB) ou bit de poids fort

<u>Exemple</u> : $(1\ 0\ 1)_2 = 2^{0}$ ´ 1 + 2^{1} ´ 0 + 2^{2} ´ 1 = 1 + 0 + 4 = $(5)_{10}$

Représentation algébrique

Une fonction logique peut être représentée sous deux formes canoniques:

S. D. P: □$\Sigma(\Pi)$ somme des produits,

•P. D.S.: □$\Pi(\Sigma)$ produit des sommes,

REPRESENTATION D’UNE FONCTION LOGIQUE

Formes canoniques d’une fonction booléenne

Mintermes :

Un « minterme » de n variables est un **produit** comportant n facteurs, chaque facteur correspondant à une variable **donnée** ou à son **complémentaire.**

exemple :

Exemple: soit a, b, c et d quatre variables booléennes.

- $abcd, \bar{a}b\bar{c}d$ et $ab\bar{c}\bar{d}$ sont trois mintermes construit à partir des variables a, b, c et d.

- $abc, \bar{a}bd$ et $b\bar{c}\bar{d}$ne sont pas des mintermes.

Remarque: A partir de n variables booléennes on peut élaborer 2^nmintermes

REPRESENTATION D’UNE FONCTION LOGIQUE

Formes canoniques d’une fonction booléenne

Maxtermes:

Un «maxterme» de n variables booléennes est une **somme** comportant n termes, chaque terme correspondant à une variable donnée ou à son complémentaire.

exemple :

Exemple: soit a, b, c et d quatre variables booléennes.

- $a+b+c+d, \bar{a}$+b+$\bar{c}$ +d et a+b+$\bar{c}+\bar{d}$ sont trois maxtermes construit à partir des variables a, b, c et d.
- $a+b+c, \bar{a}$ +b+d et b+$\bar{c}+\bar{d}$ne sont pas des maxtermes.

Remarque: A partir de n variables booléennes on peut élaborer 2^nmaxtermes

Formes canoniques d’une fonction booléenne

RQ: le complement de n’importe qu’elle minterme est un maxterme et vis versca

Table de vérité						
Combinaison	**A**	**B**	**C**		**Minterme**	**Maxterme**
0	0	0	0		$\overline{A}\,\overline{B}\,\overline{C}$	$A+B+C$
1	0	0	1		$\overline{A}\,\overline{B}\,C$	$A+B+\overline{C}$
2	0	1	0		$\overline{A}\,B\,\overline{C}$	$A+\overline{B}+C$
3	0	1	1		$\overline{A}\,B\,C$	$A+\overline{B}+\overline{C}$
4	1	0	0		$A\,\overline{B}\,\overline{C}$	$\overline{A}+B+C$
5	1	0	1		$A\,\overline{B}\,C$	$\overline{A}+B+\overline{C}$
6	1	1	0		$A\,B\,\overline{C}$	$\overline{A}+\overline{B}+C$
7	1	1	1		$A\,B\,C$	$\overline{A}+\overline{B}+\overline{C}$

1ère Forme canonique : forme conjonctive

on appelle 1ère forme canonique appellée aussi forme des mintermes si la fonction logique F sécrit sous la forme somme des mintermes

- $F\,(a,b,c....) = \Sigma\, m_i F(m_i)$

exemple :

Exemple: soit a, b, c et d quatre variables booléennes.

- $F(a,b,c,d) = abcd + \bar{a}b\bar{c}d + abc\bar{d}$ sécrit sous la 1ère forme canonique

- $F(a,b,c) = a + bc + ab\bar{c}$ ne sécrit pas sous la 1ère forme canonique

REPRESENTATION D'UNE FONCTION LOGIQUE

Question : est qu'on peut ecrire une fonction non canonique sous la premiere forme canonique?

La reponse est **oui ==>** **deux methodes**

- **Methode algebrique :** consiste a faire le produit de chaque terme par les **variables absentes** sous forme Π(xi + $\overline{xi}$)

<u>exemple</u> :

- $F(a,b,c) = \underset{(b+\bar{b})(c+\bar{c})}{a} + \underset{(a+\bar{a})}{bc} + ab\bar{c}$

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

$$F(a,b,c) = abc + ab\bar{c} + a\bar{b}c + a\bar{b}\bar{c} + \bar{a}bc$$

REPRESENTATION D'UNE FONCTION LOGIQUE

- **Table de verité :** pour chacune des combinaisons de la table de vérité, évaluer l'équation et reporter le résultat dans la table.

exemple :

- $F(a,b,c) = a + bc + ab\bar{c}$

a	b	c	f(a,b,c)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

1ere forme canonique
F(a,b,c) $= \Sigma\, m_i F(m_i)$

$$F(a,b,c) = abc + ab\bar{c} + a\bar{b}c + a\bar{b}\bar{c} + \bar{a}bc$$

2ème Forme canonique : forme disjonctive

on appelle **2ème** forme canonique appellée aussi forme des maxtermes si la fonction logique F sécrit sous la forme produit des maxtermes

- $F\ (a,b,c\ldots.) = \Pi\ (M_{i+} F(M_i)\ )$

<u>exemple</u> :

Exemple: soit a, b, c et d quatre variables booléennes.

- $F(a,b,c,d) = (a+b+c+d)\ (\bar{a}+b+\bar{c}+d)(a+b+\bar{c}+\bar{d})$ sécrit sous la **2ème** forme canonique
- $F(a,b,c) = a+bc+ab\bar{c}$ ne s’écrit pas sous la **2ème** forme canonique

– **Extraction d’une équation logique à partir de la table de vérité :**

passage de la table de verité vers la 1ere forme canonique

a	b	c	f(a,b,c)	
0	0	0	1	1
0	0	1	0	
0	1	0	0	
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	0	
1	1	1	0	

<u>Points vrais :</u>

F(0,0,0) = 1 = a b c

F(0,1,1) = 1 = a b c

F(1,0,0) = 1 = a b c

F(1,0,1) = 1 = a b c

<u>Forme normale disjonctive</u> :
Elle ne comprend que les min termes pour lesquels la valeur particulière de la fonction est égale à **1** (points vrais).
Le nombre de termes de la réunion est égale au nombre de 1 de la fonction figurant dans la table de vérité.

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

REPRESENTATION D'UNE FONCTION LOGIQUE

– **Extraction d'une équation logique à partir de la table de vérité :**

passage de la table de verité vers la 2ème forme canonique

a	b	c	f(a,b,c)
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Forme normale conjonctive :
Elle ne comprend que les max termes pour lesquels la valeur particulière de la fonction est égale à **0** (points faux).

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

Représentation des fonctions logiques

- Les formes numériques
  - Chaque combinaison est repérée par un numéro (en général, l'équivalent décimal) afin de condenser l'écriture.
    Exemple précédent :

$$F(a,b,c) = abc + ab\bar{c} + a\bar{b}c + a\bar{b}\bar{c} + \bar{a}bc$$

F(a,b,c) = ∑ (3,4, 5, 6,7)=∑ (011, 100, 101, 110,111)

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

definition

On appelle forme minimale d'une expression logique l'expression sous forme réduite qui comporte :

- Le nombre minimal de terme.
- Le nombre minimal de variable dans chaque terme

On dispose de plusieurs outils de simplification de fonction logique dont on va citer deux :

•methode Algébrique

•methode graphique (tableau de karnaugh)

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

Simplification Algébrique

Dans cette première méthode, on se base essentiellement sur les théorèmes de l'algèbre deBoole pour simplifier les expressions logiques.

Exemples: simplifier les fonctions suivantes

$$F_1 = \bar{a}bc + abc + ab\bar{c} + \bar{a}b\bar{c}$$

$$F_2 = ab + \bar{a}b + a\bar{b}$$

$$F_3 = abc + ab\bar{c} + \bar{a}b\bar{c} + ab\bar{c}$$

Solution :

$$F_1 = \bar{a}bc + abc + ab\bar{c} + \bar{a}b\bar{c} = \bar{a}bc + ab(c + \bar{c}) + \bar{a}b\bar{c}$$

$$F_1 = \bar{a}b(c + \bar{c}) + ab = \bar{a}b + ab = b(a + \bar{a}) = \mathrm{b}$$

$$F_2 = ab + \bar{a}b + a\bar{b}$$

$$F_2 = b(a + \bar{a}) + a\bar{b}$$

$$F_2 = b + \bar{b}a$$

$F_2 = b + a$ (d'après théorème d'allégement)

$$F_3 = abc + ab\bar{c} + \bar{a}b\bar{c} + ab\bar{c}$$

Simplification par le tableau de Karnaugh

Les termes Adjacents

examinons l’expression suivante :

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

- les deux termes possèdent les meme variables
- la seule différence est l’état de la variable B qui change
- si on applique les règles de simplification on Obtient :

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

- ces termes sont dites adjacents

Simplification par le tableau de Karnaugh

Exepmles des termes Adjacents

ces termes sont adjacents:

- $AB+B\bar{A}$
- $ABC+A\bar{B}C$
- $ABCD+AB\bar{C}D$

ces termes ne sont pas adjacents:

- $AB+\bar{A}\bar{B}$
- $ABC+A\bar{B}\bar{C}$
- $ABCD+\bar{A}B\bar{C}D$

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

Description de la table de Karnaugh

•Le tableau de Karnaugh est un outil graphique qui permet de simplifier de manière méthodique une équation logique ou une table de vérité et facilite ainsi la réalisation pratique du circuit correspondant.

•Le tableau de karnaugh comme la table de vérité est instrument qui met en évidence les rapports entre les entrées logiques et la sorie recherchée

•La méthode de karnaugh se base sur la règle de termes adjacents (qui ne diffèrent que par l'état d'une seule variable)

•cette méthode peut s'apliquer aux fonctions logiques de 2, 3, 4, 5 et 6 variables

Nombre de case d'un tableau de Karnaugh

•Le tableau de Karnaugh comprendre autant des cases qu'il 'ya de lignes dans la table de vérité. Il comportent 2^N cases (avec N le nombre de variables)

Réalisation Pratique du tableau de Karnaugh

•Un tableau de Karnaugh est une table de vérité dans lequel chaque case représente un Minterme. Les cases du tableau sont ordonnées suivant un **code binaire réfléchi (le code GRAY)** de tel sorte que pour passer d'une **case** à une **autre** seule **une variable change d'état.**

Nombre de case d’un tableau de Karnaugh

•Le tableau de Karnaugh comprendre autant des cases qu’il ‘ya de lignes dans la table de vérité. Il comportent 2^N cases (avec N le nombre de variables)

Réalisation Pratique du tableau de Karnaugh

•Un tableau de Karnaugh est une table de vérité dans lequel chaque case représente un Minterme. Les cases du tableau sont ordonnées suivant un **code binaire réfléchi (le code GRAY)** de tel sorte que pour passer d'une **case** à une **autre** seule **une variable change d'état.**

Remarque :

•on construit le code gray en recopiant les bits de façon symétrique(effet memoire) et en procedant de façon itérative jusquau bit désiré

itteration 1

0
1

itteration 2

0	0
0	1
1	1
1	0

itteration 3

0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

CONSTRUCTION DU TABLEAU DE KARNAUGH

Exemple

- n=2

A \ B	$\overline{B}$(0)	B(1)
$\overline{A}$(0)	00	01
A(1)	10	11

- n=3

A \ BC	$\overline{B}\,\overline{C}$(00)	$\overline{B}C$(01)	BC(11)	$B\overline{C}$(10)
$\overline{A}$(0)	000	001	011	010
A(1)	100	101	111	110

- n=4

AB \ CD	$\overline{C}\,\overline{D}$(00)	$\overline{C}D$(01)	CD(11)	$C\overline{D}$(10)
$\overline{A}\,\overline{B}$(00)	0000	0001	0011	0010
$\overline{A}B$(01)	0100	0101	0111	0110
AB(11)	1100	1101	1111	1110
$A\overline{B}$(10)	1000	1001	1011	1010

S

d \ abc	000	001	011	010	110	111	101	100
0	$\overline{a}\,\overline{b}\,\overline{c}\,\overline{d}$	$\overline{a}\,\overline{b}c\overline{d}$	$\overline{a}bc\overline{d}$	$\overline{a}b\overline{c}\,\overline{d}$	$ab\overline{c}\,\overline{d}$	$abc\overline{d}$	$a\overline{b}c\overline{d}$	$a\overline{b}\,\overline{c}\,\overline{d}$
1	$\overline{a}\,\overline{b}\,\overline{c}d$	$\overline{a}\,\overline{b}cd$	$\overline{a}bcd$	$\overline{a}b\overline{c}d$	$ab\overline{c}d$	$abcd$	$a\overline{b}cd$	$a\overline{b}\,\overline{c}d$

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

- **Passage de la table de vérité au tableau de KARNAUGH**

Une équation logique peut être représentée par une table de vérité ou un tableau de KARNAUGH.

•pour chaque combinaison qui représente un Minterme lui correspont une case dans le tableau qui doit etre mise 1

•pour chaque combinaison qui représente un Maxterme lui correspont une case dans le tableau qui doit etre mise 0

•Lorsqe on remplis le tableau, on doit soit predre les Mintermes ou les Maxtermes

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

- **Passage de la table de vérité au tableau de KARNAUGH**

Soit l'équation : $S = ab + a\bar{b}c + \bar{b}c$

Table de vérité

a	b	c	S
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Tableau de KARNAUGH

S \ ab (c)	00	01	11	10
0	0	0	1	0
1	1	0	1	1

ou bien

S \ bc (a)	00	01	11	10
0	0	1	0	0
1	0	1	1	1

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

- **Passage de la forme canonique à la table de Karnaugh**

- Si la fonction logique est donnée sous la première forme canonique(disjonctive), alors sa représentation est directe : pour chaque terme lui correspond une seule case qui doit être mise à 1.

- Si la fonction logique est donnée sous la deuxième forme canonique(conjonctive), alors sa représentation est directe : pour chaque terme lui correspond une seule case qui doit être mise à 0.

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

- **Passage de la forme canonique à la table de Karnaugh**

Exemple

$$F1(A,B,C) = \sum (1,2,5,7)$$

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

$$F2(A,B,C) = \prod (0,2,3,6)$$

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

Simplification d’équations

La méthode consiste à réaliser **des groupements** de **CASES ADJACENTES** contenant des 1 ou des 0. Un groupement de 1 permet d’obtenir l’équation de**S**, un groupement de 0 permet d’obtenir l’équation $\bar{S}$

Règle

- Le **nombre de cases** d’un groupement doit être égal à **1, 2 ,4 , ...2^n**
- Les **groupements** doivent être les **plus grands possibles**
- **Les groupements** peuvent se **chevaucher** pour être **les plus grands possibles.**
- Dans **chaque groupement** on **ne retient** que les **variables** dont l’état **ne change pas.**
- Pour extraire l’équation de la fonction logique on **ne retient** que **les variables** dont l’état **ne change pas** à l’intérieur d’un **groupement** et on effectue **la somme logique** (OU logique) de toutes les **expressions trouvées**

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

Exemple : regroupement par 2

S: a \ bc	00	01	11	10
0	**0**	**0**	**0**	**0**
1	0	1	1	0

$$a\bar{b}c + abc = ac$$

S: a \ bc	00	01	11	10
0	**1**	**0**	**0**	**0**
1	1	0	0	0

$$\bar{b}\bar{c}$$

S: a \ bc	00	01	11	10
0	**1**	**0**	**0**	**1**
1	0	0	0	0

$$\bar{a}\bar{c}$$

50

- Une case peut appartenir à plusieurs regroupements

- **Les groupements** peuvent se **chevaucher** pour être **les plus grands possibles.**

- On s'arrête lorsque il y a plus de **1** en dehors des regroupements

S: a \ bc	00	01	11	10
0	**0**	**0**	**1**	**0**
1	0	1	1	1

S= ac+ab+bc

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

Exemple : regroupement par 4

S: a \ bc	00	01	11	10
0	0	1	1	0
1	0	1	1	0

c

S: ab \ cd	00	01	11	10
00	1	1	1	1
01				
11				
10				

$\bar{a}\bar{b}$

S: ab \ cd	00	01	11	10
00		1		
01		1		
11		1		
10		1		

$\bar{c}d$

S: ab \ cd	00	01	11	10
00	1			1
01				
11				
10	1			1

$\bar{b}\bar{d}$

SIMPLIFICATION DES FONCTIONS BOOLÉENNE

Exemple : regroupement par 8

S

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

$\bar{a}$

S

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

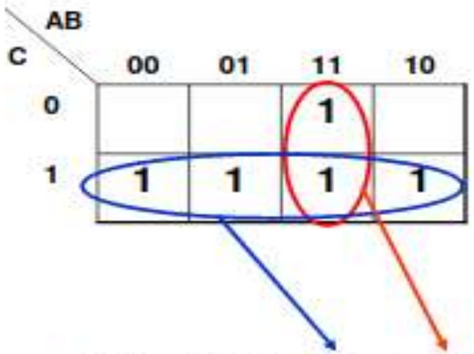
$\bar{c}$

S

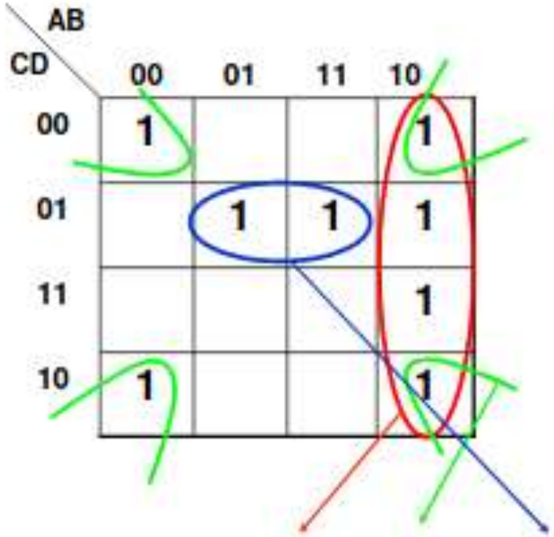
ab \ cd	00	01	11	10
00	1			1
01	1			1
11	1			1
10	1			1

$\bar{d}$

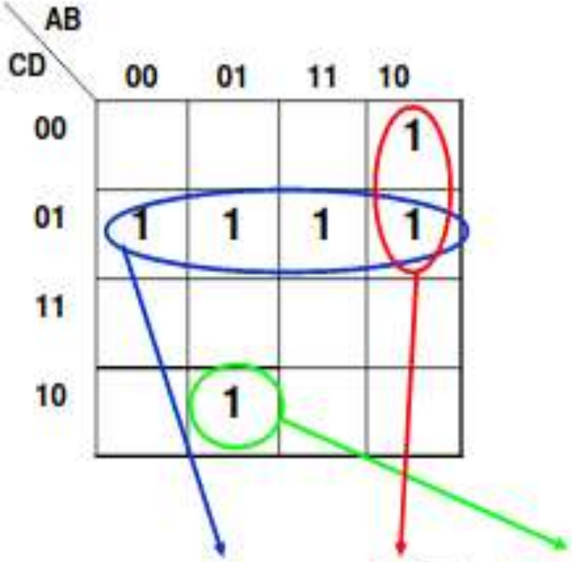
Exemple 1:



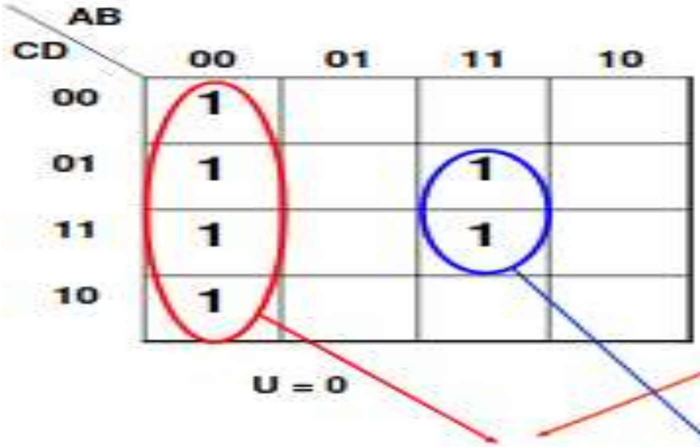
$$F(A,B,C) = C + AB$$

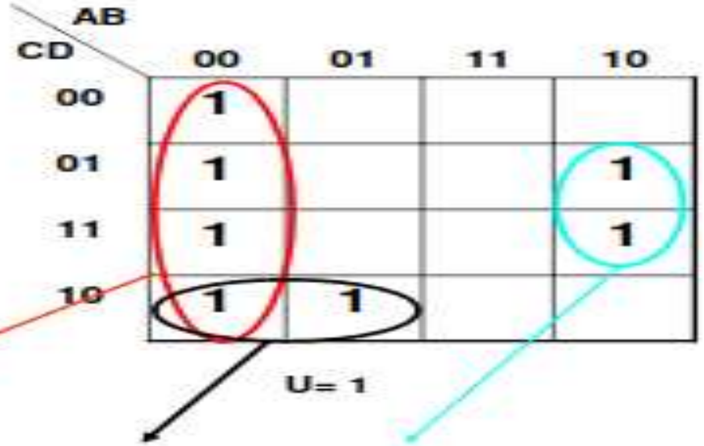


$$F(A,B,C,D) = A\overline{B} + \overline{B}\,\overline{D} + B\overline{C}D$$



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





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

en résumé

pour simplifier une fonction par la table de karnaugh il faut suivre les étapes suivantes :

1. **Remplir** le tableau à partir de la table de vérité ou à partir de la forme canonique.

2. **Faire des regroupements** : des regroupements de 16,8,4,2,1 cases (Les même termespeuvent participer àplusieurs regroupements) .

en résumé

3. Dans un regroupement :

- Qui contient **un seule terme** on peut pas éliminer de variables.
- Qui contient **deux termes** on peut éliminer **une variable** (celle qui change d'état).
- Qui contient **4 termes** on peut éliminer **2 variables**.
- Qui contient **8 termes** on peut éliminer **3 variables**.
- Qui contient **16 termes** on peut éliminer 4 variables.

4. L'expression logique finale est la réunion (la somme) des groupements après simplification et élimination des variables qui changent d'état.

Simplification de fonctions logiques

– Cas des fonctions incomplètement définies :

- Certaines combinaisons ne peuvent jamais exister.

- la valeur de la fonction n’a pas d’importance pour certaines combinaisons de variables.

- La valeur de la fonction est dite indifférente ou la combinaison interdite. La valeur de la fonction est alors notée Φ ou X et peut prendre indifféremment la valeur 1 ou 0 selon qu’elle sert ou non à la simplification.

Simplification de fonctions logiques

– Cas des fonctions incomplètement définies :

- Examinons l'exemple suivant :

Une serrure de sécurité s'ouvre en fonction de quatre clés A, B, C D. Le fonctionnement de la serrure est définie comme suite :   S(A,B,C,D)= 1 si au moins deux clés sont utilisées   S(A,B,C,D)= 0 sinon   Les clés A et D ne peuvent pas être utilisées en même temps.

- On remarque que si la clé A et D sont utilisées en même temps l'état du système n'est pas déterminé.

- Ces cas sont appelés cas impossibles ou interdites → comment représenter ces cas dans la table de vérité ?.

Simplification de fonctions logiques

–Cas des fonctions incomplètement définies :

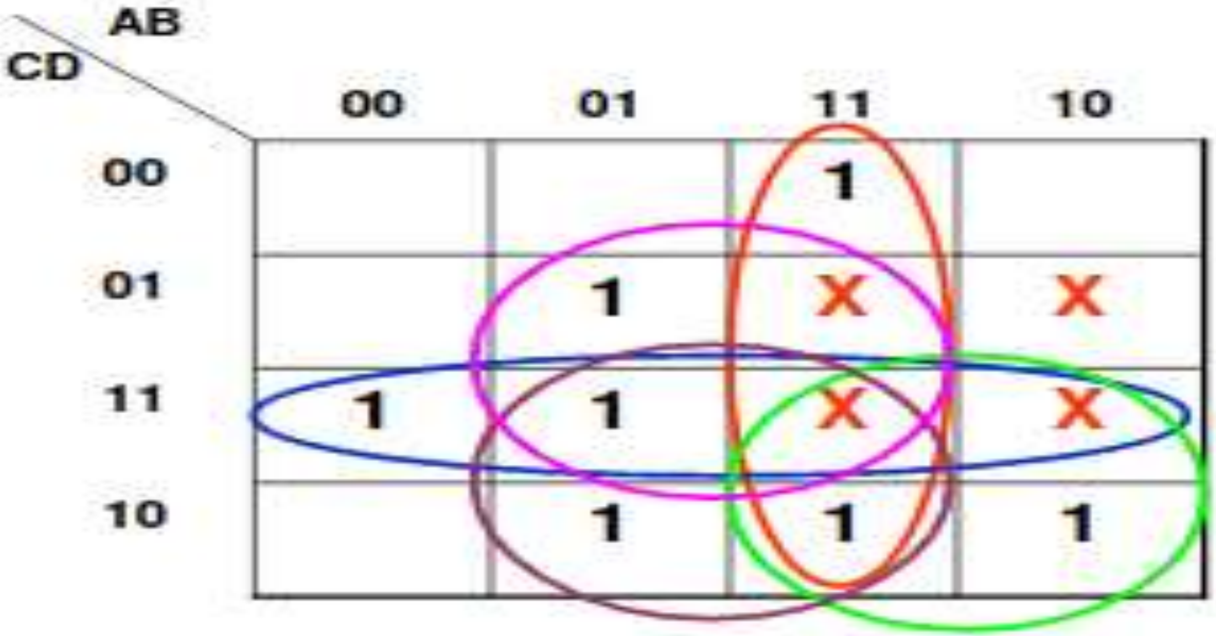
- Pour les cas impossibles ou interdites il faut mettre un X dans la T.V .
- Les cas impossibles sont représentées aussi par des X dans la table de karnaugh

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

A	B	C	D	S
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	X
1	0	1	0	1
1	0	1	1	X
1	1	0	0	1
1	1	0	1	X
1	1	1	0	1
1	1	1	1	X

Simplification de fonctions logiques

– Cas des fonctions incomplètement définies :

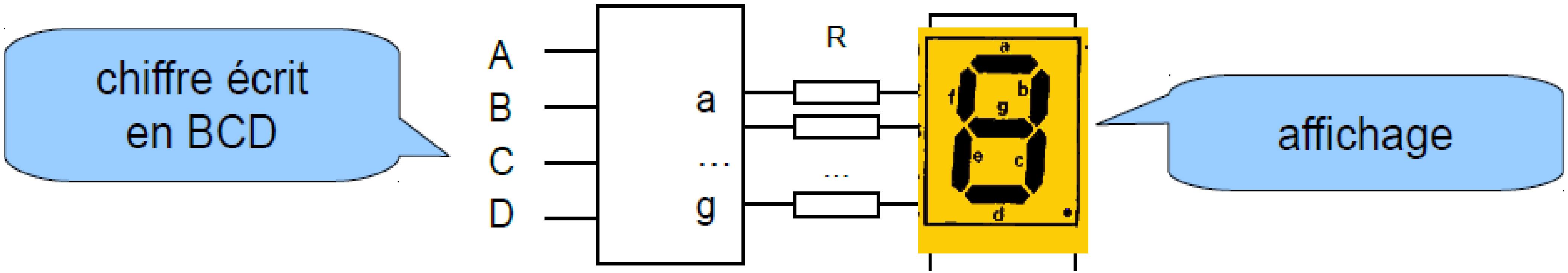


$AB + CD + BD + AC + BC$

Simplification de fonctions logiques

– **Cas des fonctions incomplètement définies :**

- Exemple 1: Afficheur à 7 segments
  - Sert à afficher sur un chiffre allant de 0 à 9.
  - Ce nombre est représenté sur 4 bits ➔ permettant de représenter les nombres allant de 0 à 15
  - Les combinaisons d'entrées de 10 à 15 dont dites des conditions facultatives ou fonction indifférents puisqu'elles ne devraient jamais être présentées



a= R(0,2,3,5,6,7,8,9) + Φ(10,11,12,13,14,15)

b= R(0,1,2,3,4,7,8,9) + Φ(10,11,12,13,14,15)

– **Cas des fonctions incomplètement définies :**

- **Exemple 2**

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

$\bar a\bar c\bar d$

bd

$a\bar b\bar d$

$F = \bar a\bar c\bar d + bd + a\bar b\bar d$

– Cas des fonctions incomplètement définies :

- **Exemple 3**

$$F = b\bar{c}d + bc\bar{d} + ad + b\bar{c}d$$

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

- $\bar{b}cd$
- $b\bar{c}d$
- $bc\bar{d}$
- da