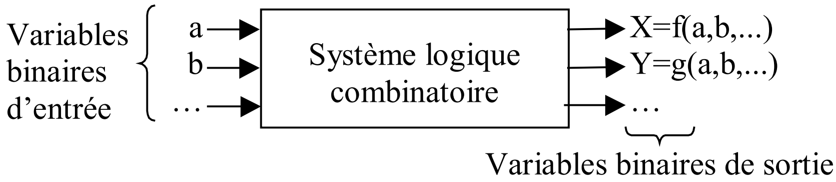
3. La fonction logique combinatoire

1. Introduction

- **Système logique combinatoire**



Caractériser un système logique combinatoire consiste à obtenir une représentation des fonctions de ce système. Ces fonctions expriment les **relations** entre les variables de sortie et les variables d'entrée. Elles peuvent se représenter sous forme de:

- table de vérité,
- représentation décimale (somme de états ou produit des états)
- expression canonique,
- tableau de KARNAUGH
- représentation électrique
- diagramme à contacts (Ladder)
- schéma logique (logigramme)
- …

- **Table de vérité** : Une fonction logique combinatoire $F(x_1,\dots,x_n)$ de n variables binaires peut être définie par la valeur 0 ou 1 pour **chacun** des 2^n états d'entrée possibles. Cette liste de couples comprenant un état d'entrée et l'état de sortie correspondant est représentée sous forme d'un tableau appelé table de vérité de la fonction F.

  Exemple**:** Considérons un système logique détectant parmi ses 3 variables d'entrée si on a un nombre impair de variables à 1. On appelle F la fonction réalisée par ce système.

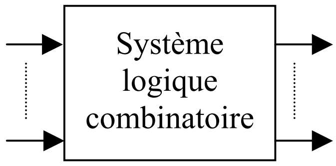
a	**b**	**c**	**F**
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

3 variables d'entrée
=
2^3 combinaisons

Table de vérité de F

2. Fonctions de n variables

n variables d'entrée
=
2^n combinaisons possibles



Pour une combinaison d'entrée, la fonction peut prendre deux valeurs {0,1}
=
2^{2^n} fonctions possibles

- **Fonctions d'une variable (n=1)**

La table de vérité suivante représente les 4 fonctions possibles d'une variable a. Une variable binaire peut prendre 2 valeurs. Pour chaque état d'entrée, la fonction peut prendre deux valeurs. Il existe donc $2^2 = 4$ fonctions d'une variable :

a	F
0	0
1	0

Fonction nulle
F = 0

a	F
0	0
1	1

Fonction identité
F = a

a	F
0	1
1	0

Fonction complément
F = $\overline{a}$

A	F
0	1
1	1

Fonction vraie
F = 1

- **Fonctions de deux variables (n=2)**

La table de vérité suivante représente toutes les fonctions possibles de deux variables a et b. Il existe $2^4 = 16$ fonctions de deux variables :

a	b	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}
0	**0**	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	**1**	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	**0**	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	**1**	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Ecriture des fonctions :

$F_1 = 0$ $F_4 = a$ $F_6 = b$ $F_{16} = 0$ $F_{11} = \overline{b}$

« NON a » ou « complément de a » (***NOT a***) : $F_{13} = \overline{a}$

ET (***AND***) : $F_2 = a\,.\,b$ OU (***OR***) : $F_8 = a + b$

NON-ET (***NAND***) : $F_{15} = \overline{a.b}$ NON-OU (***NOR***) : $F_9 = \overline{a + b}$

OU-Ex (***XOR***) : $F_7 = a \oplus b$ NON-OU-Ex (***NXOR***) : $F_{10} = \overline{a \oplus b}$

3. Expressions canoniques

Définitions: On appelle **minterme** de *n* variables, un produit logique de ces variables. Avec *n* variables, on a 2^n mintermes (autant de combinaisons possibles de *n* éléments binaires).
On appelle **maxterme** de *n* variables, une somme logique de ces variables. Avec *n* variables, on a 2^n maxtermes (autant de combinaisons possibles de *n* éléments binaires).

Exemples : Pour 2 variables a et b, on a :
→ 4 mintermes : $\overline{a}\,\overline{b}$, $\overline{a}\,b$, $a\,\overline{b}$, $a\,b$
→ 4 maxtermes : $\overline{a}+\overline{b}$, $\overline{a}+b$, $a+\overline{b}$, $a+b$

A partir de la table de vérité d'une fonction logique, il est possible d'obtenir deux expressions algébriques (écrites sous 2 formes différentes mais équivalentes) correspondantes à la fonction. Ceux sont les expressions canoniques de la fonction:

- La 1^{er} forme canonique (Somme de produits algébriques ou somme de mintermes) est obtenue en considérant les états d'entrée pour lesquels la fonction vaut 1.
- La 2^{nde} forme canonique (Produits de sommes algébriques ou produit de maxtermes) est obtenue en considérant les états d'entrée pour lesquels la fonction vaut 0.

Fonction d'une variable

Soit F(a) une fonction logique de la variable a. Soient F(1) la valeur de la fonction quand a = 1 et F(0) sa valeur quand a = 0. Alors F(a) s'écrit de la façon suivante :

- 1^{er} forme canonique : $F(a) = a.\,F(1) + \overline{a}\,.\,F(0)$
- 2^{nde} forme canonique : $F(a) = (\,a + F(0)\,)\,.\,(\,\overline{a} + F(1)\,)$

Fonction de 2 variables

Soit F(a,b) une fonction logique de deux variables a et b. Soient F(0,0), F(0,1), F(1,0) et F(1,1) les 4 valeurs que peut prendre la fonction pour les états d'entrée correspondants. Alors F(a,b) s'écrit de la façon suivante :

- 1^{er} forme canonique : $F(a,b) = \overline{a}\,\overline{b}.F(0,0) + \overline{a}\,b.F(0,1) + a\,\overline{b}.F(1,0) + a\,b.F(1,1)$
- 2^{nde} forme canonique :

$$F(a,b) = (a+b+F(0,0)).(a+\overline{b}+F(0,1)).(\overline{a}+b+F(1,0)).(\overline{a}+\overline{b}+F(1,1))$$

- **Fonction de n variables (théorème de Shannon)**

Soit une fonction de n variables (x_1 … x_n) où les nombres F(0,0,…0,0), F(0,0,…0,1), … F(1,1…,1,1) représentent les 2^n valeurs que peut prendre la fonction pour les 2^n états correspondants. La fonction F s'écrit alors (1er forme canonique) :

$$F\left(x_n,...,x_1\right)=\overline{x_n}\,\overline{x_{n-1}}...\overline{x_2}\,\overline{x_1}.F(0,...,0)+\overline{x_n}\,\overline{x_{n-1}}...\overline{x_2}\,x_1.F(0,...,1)+...+x_n x_{n-1}...x_2 x_1.F(1,...,1)$$

- **Exemple :**

Soit un système caractérisé par la fonction F(a,b) suivante définie par sa table de vérité:

a	b	F
0	0	1
0	1	0
1	0	0
1	1	1

1er forme canonique : $F=\overline{a}\,\overline{b}.1+\cancel{\overline{a}b.0}+\cancel{a\overline{b}.0}+a\,b.1 \;=\; \overline{a}\,\overline{b}+a\,b$

2nd forme canonique: $F=(\cancel{a+b+1})(\overline{a}+b+0)(a+\overline{b}+0)(\cancel{\overline{a}+\overline{b}+1}) \;=\; (\overline{a}+b)(a+\overline{b})$

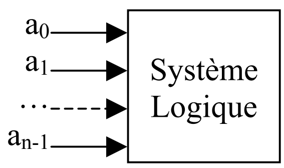
4. Représentation décimale d'une fonction logique

- **Définitions :**

On utilise un codage décimal pour représenter les états des variables binaires d'entrée. Par convention, ce code est l'équivalent décimal du mot codé en binaire naturel :

$N = a_{n-1}.2^{n-1} + a_{n-2}.2^{n-2} + …+ a_1.2^1 + a_0.\,2^0$

où les a_i représentent les variables d'entrées (a_0 : bit de poids faible)



Une fonction logique F peut être définie comme :

→ Une somme des états pour lesquelles elle vaut 1 . On note F = Σ(N_1, …Np)

ou bien

→ Un produit des états pour lesquelles elle vaut 0 . On note F = Π(N_1, …Nq)

- **Exemple :**

Considérons un système caractérisé par une fonction logique F(a,b,c) de 3 variables. Cette fonction détecte parmi ces trois variables si on a un nombre pair de variables à 1.

On pose la variable « a » poids fort : $N = a \cdot 2^2 + b \cdot 2^1 + c \cdot 2^0$

N	a	b	c	F
0	0	0	0	**1**
1	0	0	1	**0**
2	0	1	0	**0**
3	0	1	1	**1**
4	1	0	0	**0**
5	1	0	1	**1**
6	1	1	0	**1**
7	1	1	1	**0**

- Ecriture en somme des états : $F = \Sigma(0,3,5,6)$

→Equivalence avec la forme canonique :

$$F = \Sigma(0,3,5,6) = \overline{a}.\overline{b}.\overline{c} + \overline{a}.b.c + a.\overline{b}.c + a.b.\overline{c}$$

- Ecriture en produit des états : $F = \Pi(1,2,4,7)$

→ Equivalence avec la forme canonique

$$F = \Pi(1,2,4,7) = (a+b+\overline{c}).(a+\overline{b}+c).(\overline{a}+b+c).(\overline{a}+\overline{b}+\overline{c})$$

5. Fonctions incomplètement spécifiées

Il arrive dans certains systèmes logiques combinatoires que la valeur prise par la fonction caractérisant le système ne soit pas spécifiée pour une ou plusieurs des combinaisons des variables d'entrée. C'est à dire qu'on ne connaît pas à priori (ou alors ce n'est pas défini) la valeur 0 ou 1 que prend la fonction pour une ou plusieurs combinaisons d'entrée.

- **Exemple :**

On suppose un train électrique. On dispose de deux commandes “a” (avancer) et “r” (reculer). On regarde l’état roulant ou non du train considéré (si le moteur tourne).

Le système combinatoire associé est le suivant:

a	r	M	Etat Moteur
0	**0**	0	Le moteur ne tourne pas
0	**1**	1	Le moteur tourne
1	**0**	1	Le moteur tourne
1	**1**	?	Aucune information !

Dans la réalité, on s’arrangera pour que la combinaison d’entrée « a=1 et r = 1 » ne soit jamais atteignable : on se limite à un fonctionnement normal (Par exemple un dispositif mécanique ne permet pas l’action simultanée de « a » et « r »)

→ Aucune valeur de sortie est affectée pour cette valeur d’entrée qui ne se produira jamais. On dit dans ce cas que la fonction est incomplètement spécifiée. On note ∅ la valeur associée des états non spécifiés :

$$M(1,1) = \varnothing \quad \text{(indifférent)}$$

→ On pourra alors fixer **arbitrairement** la valeur de M quand « a=1 et r = 1 ». Ici par mesure de sécurité, on peut imposer M=0 ; c’est à dire qu’on accorde la priorité à l’arrêt par rapport à marche dans le cas où les deux boutons sont actionnés.

ATTENTION :
Pour éviter toute ambiguïté entre le symbole ∅ et le zéro informatique 0, on préférera la notation « × » ou « – » pour noter les indifférents dans une table de vérité !

- **Définition** :

Pour une fonction incomplètement spécifiées, il existe :

- la couverture supérieure d’une fonction incomplète : c’est la fonction logique obtenue en remplaçant toutes les valeurs non spécifiées par des 1

- la couverture inférieure d’une fonction incomplète : c’est la fonction logique obtenue en remplaçant toutes les valeurs non spécifiées par des 0.