Chapitre 2 : Les circuits

I- Définition :

- Variable binaire : elle est représentée par une grandeur physique qui ne peut prendre que deux états 0 et 1 entre lesquels elle effectue des transitions.
- Fonction binaire : elle dépend d'un certain nombre de variables binaires indépendantes ou d'autres fonctions.
- Algèbre binaire (ou algèbre de Bool)
Elle étudie l'état des variables et des fonctions binaire.
- Analyse binaire : elle étudie les transitions

II- Opérateurs logiques

1) Opérateurs élémentaires (de base) :

a) Egalité :

Deux variables binaires sont égales si les termes de même rang sont égaux

b) Complémentation (inversion) 'Non'

Il correspond au complément de l'état actuel de la variable.

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

c) Le produit logique (Intersection) : "•"

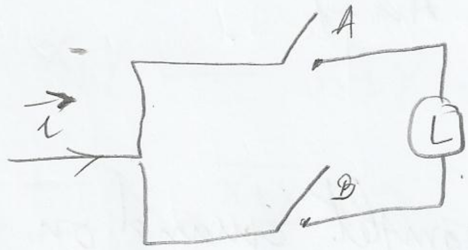
X	Y	$X \cdot Y$
0	0	0
0	1	0
1	0	0
1	1	1

circuit série

Opérateur "et"

d) Produel logique

Il correspond au niveau de la théorie des ensembles à la réunion et correspond à un circuit en parallèle en physique.



X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1

opérateur "OU"

* Théorème de Morgan

Pour inverser une fonction il faut inverser les variables et remplacer les opérateurs ET par OU et inversement.

Preuve :

$$\overline{A_1 . A_2 \cdots A_n} = \overline{A_1} + \overline{A_2} + \cdots + \overline{A_n}$$

$n = 2$ $\overline{A_1 . A_2} = \overline{A_1} + \overline{A_2}$

Supposons au rang n que

$$\overline{A_1 . A_2 \cdots A_n} = \overline{A_1} + \overline{A_2} + \cdots + \overline{A_n}$$

Demontrons que c'est vrai au rang n+1

$$\overline{A_1 . A_2 \cdots A_{n+1}} \overset{?}{=} \overline{A_1} + \overline{A_2} + \cdots + \overline{A_{n+1}}$$

$$\overline{A_1 . A_2 \cdots A_{n+1}} = \overline{(A_1 . A_2 \cdots A_n) . A_{n+1}}$$

$$= \overline{X \cdot A_{n+1}}$$

$$= \overline{X} + \overline{A_{n+1}}$$

$$= \overline{A_1 \cdot A_2 \cdots A_n} + \overline{A_{n+1}}$$

$$= \overline{A_1} + \overline{A_2} + \cdots + \overline{A_n} + \overline{A_{n+1}}$$

2) Opérateurs complets

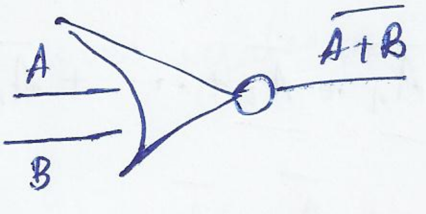
On dit qu'un opérateur est complet quand on peut redéfinir tous les opérateurs de base en l'utilisant

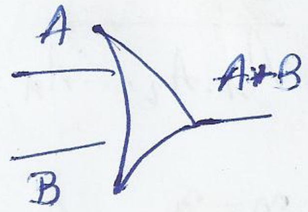
NOR (OU suivi de NON)

NAND (ET suivi de NON)

* NOR

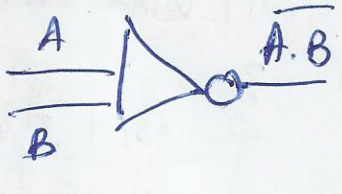
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

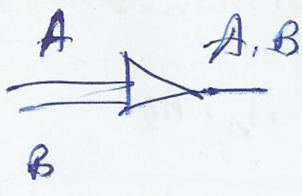




* NAND

A	B	$\overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0





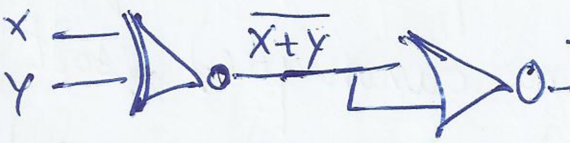
NOR

$$X \longrightarrow \overline{X} = \overline{X+X}$$

NON

OU

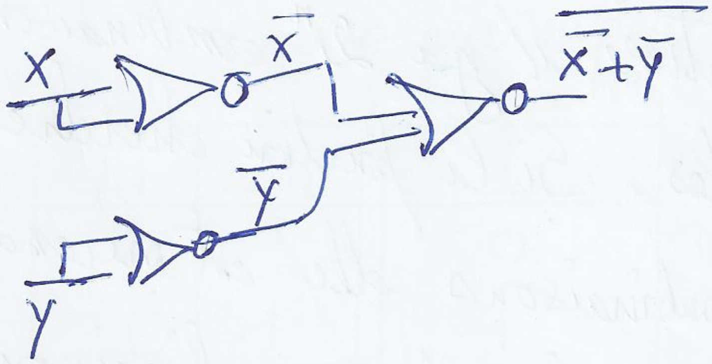
$$x, y \text{ — } x+y = \overline{(\overline{x+y})}$$



$$\overline{\overline{x+y}+\overline{x+y}} = \overline{\overline{x+y}} = x+y$$

ET

$$x, y \text{ — } x.y = \overline{\overline{xy}} = \overline{\overline{x}+\overline{y}}$$



A faire ! Pour le NAND

III - Synthèse d'un circuit :

La synthèse d'un circuit destiné à réaliser une fonction binaire donnée comprend 3 étapes

1ère étape : Écriture de l'expression de la fonction binaire

2è étape : Simplification de l'expression précédente en vue d'obtenir un circuit économique

3ème étape : Passage de l'écriture symbolique au schéma électrique du circuit.

a) 1ère étape : Formes canoniques :

Toute fonction binaire peut s'exprimer soit par un somme de produit (1ère forme canonique), soit par un produit de sommes (2e forme canonique).
Ces deux expressions représentent les 2 formes canoniques de la fonction.
Pour n variables d'entrées il y a 2^n combinaisons distinctes de variables. Si la fonction cherchée est à 1 (vrai) ou p combinaisons elle est nécessairement égale à 0 pour les $2^n - p$ combinaisons restantes.
En générale p et $2^n - p$ sont distinctes et l'une des formes canoniques est plus simple parce qu'elle a moins de termes.

* Pour la 1ère forme canonique, on a un somme de produit, le nombre de produits est déterminé par le nombre d'états à 1 de la fonction.

Pour chaque produit on prend la variable telle quelle si elle est à 1 et on la complémente sinon.

* Pour la 2e forme canonique, on a un produit de produels, le nombre de produels est déterminé par le nombre ~~(de variable)~~ d'états à 0 de la fonction.

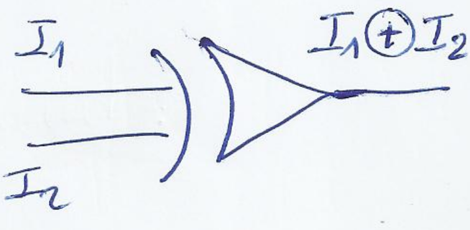
Pour chaque produel on prend la variable telle quelle si elle est à 0 et on la complémente dans le cas contraire.

Exemple : Circuit électrique d'une lampe avec deux interrupteurs.

I_1	I_2	L
0	0	0
0	1	1
1	0	1
1	1	0

1ère forme

$$L = \bar{I}_1 I_2 + I_1 \bar{I}_2$$



2e forme : produit de produel

$$L = (I_1 + I_2)(\bar{I}_1 + \bar{I}_2)$$

$$= \underbrace{I_1\bar{I}_1}_{0} + \underbrace{I_1\bar{I}_2 + I_2\bar{I}_1} + \underbrace{I_2\bar{I}_2}_{0}$$

A faire !
Synthèse circuit
— Additionneur
— Distributeur de café

b) 2e étape : Simplification

La simplification d'une fonction doit prendre en charge deux aspects :

- Eliminer tous les termes redondants (un terme est redondant quand il ne change pas la valeur numérique de la fonction
- Tenir compte des ~~technologies~~ contraintes technologiques

On peut simplifier une fonction essentiellement de deux manières

* en utilisant les propriétés des opérateurs

* en utilisant le diagramme ou la Table Karnaugh

Ex : Synthèse circuit d'un additionneur

a_i b_i R_{i-1} S_i R_i

$$S_i = \bar{a}_i \bar{b}_i S_{i-1} + \bar{a}_i b_i \overline{R_{i-1}} + a_i \bar{b}_i \overline{R_{i-1}} + a_i b_i R_{i-1}$$

$$R_i = \bar{a}_i b_i R_{i-1} + a_i \bar{b}_i R_{i-1} + a_i b_i \overline{R_{i-1}} + a_i b_i R_{i-1}$$

$$S_i = a_i \oplus b_i \oplus R_{i-1}$$

$$R_i = (a_i \oplus b_i) R_{i-1} + a_i b_i$$

~~$S_0 = a_0 b$~~

$$S_0 = a_0 \oplus b_0$$

$$R_0 = a_0 b_0$$

Diagramme de Karnaugh:

xy \ kl	00	01	11	10
00	0	0	0	0
01	0	0	0	1
11	0	1	0	0
10	0	1	0	0

(1) (2)

(1) $\forall y$: $x\bar{k}l$

(2) $\bar{x}yk\bar{l}$

$f = x\bar{k}l + \bar{x}yk\bar{l}$

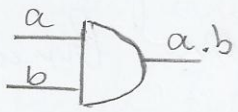
$f(x,y,k,l) = xy\bar{k}l + x\bar{y}\bar{k}l + \bar{x}yk\bar{l}$

1101 1001 0110

Portes logiques

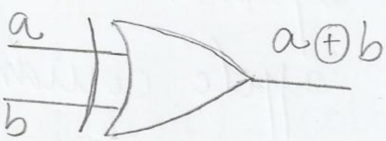
NON (NOT)

ET (AND)

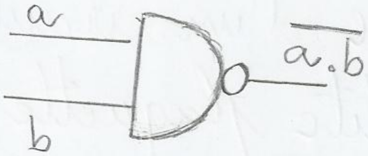


OU (OR)

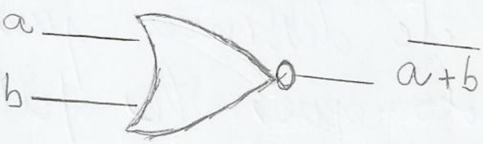
OU-exclusif (XOR)



NON ET (NAND)



NON OU (NOR)



Un diagramme de Karnaugh est une table d'implication logique disposée d'une telle manière que deux termes logiquement adjascents soient adjascents geométriquement.
Deux termes sont adjascents quand ils diffèrent l'un de l'autre par une variable.
Le diagramme de Karnaugh se présente sous la forme d'un carré ou d'un rectangle selon la parité du nbre de variables.
Chaque case est repéré par ses coordonnées placées sur les bords du diagramme selon le code binaire réfléchi.
Ce disposition au niveau du diagramme de Karnaugh fait apparaître ~~au niveau~~ de façon évidente des simplifications de la fonction binaire

c) 3e étape : Circuits intégrés logiques :

Les portes ne sont pas fabriqués ou commercialisées sous leur plus simple expression mais sous la forme de circuit intégré logique appelé couramment circuit intégré ou puce,
La partie active d'un circuit intégré logique est formée d'une petite plaquette de sillicium (la puce) sur laquelle ont été dessinées puis intégrées par divers procédés technologiques les portes qui constituent ce circuit.

La dimension de cette puce est de 5x5 mm
Cette puce de sillicium est ensuite encapsulée dans un boitier rectangulaire en plastic ou en céramique de 5 à 15mm de large sur 20 à 50 mm de long.
Sur les deux long cotés de ce boitier sont disposés de façon symétrique des broches (ou pattes) permettant d'assurer les connections électriques ~~(alim~~ (signaux d'alimentation en électricité, signaux d'entrée, signaux de sortie, etc.
On peut classer les circuits intégrés logiques suivant le nbre de porte qui y sont définies.
C'est ainsi que nous avons:

- le SSI (Small Scale Integration): circuit à faible intégration groupant de ~~une~~ à 10 portes par circuit
- le MSI (Medium Scale Intégration): circuit à moindre intégration groupant de 10 à 100 portes par circuit
- le LSI (Large Scale Integration): circuit à grandre intégration groupant de 100 à 100.000 portes par circuits
- le VLSI (Very Large Scale Integration): circuit à très grande intégration groupant plus de 100.000 portes par circuit.