

## La logique booléenne

Beaucoup de systèmes automatisés fonctionnent en utilisant des organes et des fonctions binaires. Ces organes et fonctions binaires ne peuvent être que dans deux états possibles. Par exemple, un détecteur de niveau peut être immergé ou submergé. Un voyant peut être allumé ou éteint.

Par convention, on représente par la valeur logique « 0 » l'un de ces états et par la valeur logique « 1 » l'autre état. La valeur logique « 0 » correspond à un organe binaire (ou une fonction binaire) dans un état dit « *non-activé* », « *non-actionné* » ou « *inactif* » (exemple : un voyant inactif est éteint). La valeur logique « 1 » correspond à un organe binaire (ou une fonction binaire) dans un état dit « *activé* », « *actionné* » ou « *actif* » (exemple : un voyant actif est allumé). Dans le chapitre précédent la notion de présence (niveau logique = 1) ou d'absence (niveau logique = 0) d'une grandeur physique était envisagée sous une forme logique.

La mathématique des fonctions binaires est appelée *l'algèbre booléenne* et elle fut inventée par M. **Georges Boole** en 1847. Dans un écrit nommé « The Mathematical Analysis of Logic », il définit trois opérateurs de base ainsi qu'une foule de règles et de postulats. Ainsi, toutes les fonctions binaires (dites aussi logiques) sont des relations entre des entrées et des sorties logiques composées d'opérateurs de base et sur lesquelles on peut appliquer diverses règles d'algèbre.

Les fonctions logiques (ou binaires) peuvent être représentées de diverses façons. La plus élémentaire consiste à dresser une table répertoriant toutes les combinaisons de valeurs logiques des variables soumises à des opérateurs. Ces tables sont nommées « *tables de vérité* ». Une autre représentation possible est d'écrire les fonctions logiques sous forme d'équations. Ces « *équations logiques* » facilitent la simplification des fonctions logiques grâce aux règles de l'algèbre booléenne.

Le chapitre qui suit a pour but de vous introduire à tous ces aspects et de vous permettre de comprendre les mécanismes mis en œuvre pour solutionner les systèmes logiques.

## 2-1) Les trois opérateurs de base

Les fonctions logiques reposent sur trois opérateurs de base. Ce sont les fonctions logiques : « NON » (en anglais « NOT »), « ET » (en anglais « AND ») et « OU » (en anglais « OR »). Nous les présenterons en montrant leur équation et leur table de vérité.

### 2-1-A) La fonction logique « NON »

Soit une variable booléenne nommée  $A$ . La fonction logique  $\text{NON}(A)$ , appelé **complément de  $A$** , sera notée  $\bar{A}$  (lire  $A$  barre). Le résultat de  $\text{NON}(A)$  sera également une variable booléenne. La table ci-contre est la table de vérité de cette fonction.

Fonction $F = \text{NON}(A)$	
A	F
0	1
1	0

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = \bar{A}$ .

### 2-1-B) La fonction logique « ET »

Soit deux variables booléennes nommées  $A$  et  $B$ . Le résultat de la fonction logique  $A$  ET  $B$  sera également une variable booléenne. Le tableau de droite montre la table de vérité de cette fonction. Comme vous pouvez le constater, la fonction logique ET n'active la sortie que lorsque toutes les entrées sont actives.

Fonction $F = A$ ET $B$		
A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = A \bullet B$ . Le symbole du ET est semblable à celui du produit.

### 2-1-C) La fonction logique « OU »

Soit deux variables booléennes nommées  $A$  et  $B$ . Le résultat de la fonction logique  $A$  OU  $B$  sera également une variable booléenne. La table de vérité de cette fonction est montrée à droite. Comme vous pouvez le constater, la fonction logique OU donne une valeur de sortie égale à 1 dès qu'une des entrées est à 1.

Fonction F = A OU B		
A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = A + B$ . Le symbole du OU est semblable à celui de la somme.

*Ces trois fonctions de bases sont le fondement de l'algèbre booléenne. Toutes autres fonctions logiques peuvent s'exprimer à partir de ces trois fonctions élémentaires (exemple :  $F = (A \bullet B) + \bar{C}$ ).*

### 2-2) Les autres opérateurs logiques

En plus des opérateurs de base, il existe d'autres opérateurs de deux variables et nous présenterons ici les plus importants.

#### 2-2-A) La fonction logique « NON-ET »

Soit deux variables booléennes nommées  $A$  et  $B$ . Le résultat de la fonction logique NON( $A$  ET  $B$ ) sera également une variable booléenne. La figure de droite montre la table de vérité de cette fonction. Comme vous pouvez le constater, la sortie de la fonction logique NON-ET (en anglais NAND) à un comportement inverse à celle de la fonction logique ET.

Fonction "NON-ET"		
A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = \overline{A \bullet B}$ . Remarquez l'ajout de la barre au-dessus de la fonction et traduisant ainsi l'inversion du résultat du ET.

### 2-2-B) La fonction logique « NON-OU »

Soit deux variables booléennes nommées  $A$  et  $B$ . Le résultat de la fonction logique NON( $A$  OU  $B$ ) sera également une variable booléenne. La table ci-contre montre la table de vérité de cette fonction. Comme vous pouvez le constater, la sortie de la fonction logique NON-OU (en anglais NOR) à un comportement inverse à celle de la fonction logique OU.

Fonction "NON-OU"		
A	B	F
0	0	1
0	1	0
1	0	0
1	1	0

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = \overline{A + B}$ . Remarquez l'ajout de la barre au-dessus de la fonction et traduisant ainsi l'inversion du résultat du OU.

Ces deux fonctions (NON-ET et NON-OU) sont très utilisées en électronique<sup>1</sup>, car elles représentent des éléments de connexions universels. Toute fonction logique peut en effet être écrite exclusivement à partir de l'une ou l'autre de ces fonctions.

Un des avantages de ces éléments de connexion universel est de permettre l'implantation de n'importe quelle fonction logique à l'aide d'un seul type de circuit électronique. Il y a donc **standardisation** sur un seul type de circuit électronique. Ainsi, il est possible d'en acheter une grande quantité pour bénéficier d'un prix de vente avantageux. On ne stocke qu'un seul type de circuit en prévision d'éventuelles pannes.

### 2-2-C) La fonction logique « OU-EXCLUSIF »

Soit deux variables booléennes nommées  $A$  et  $B$ . Le résultat de la fonction logique  $A$  OU-EXCLUSIF  $B$  sera également une variable booléenne. La table de vérité de droite montre le comportement de cette fonction. Comme vous pouvez le constater, la sortie de la fonction logique OU-EXCLUSIF (en anglais EXOR) ne donne un niveau logique 1 que si l'une des entrées est à un niveau logique 1.

Fonction "OU-EXCL"		
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

<sup>1</sup> Ce sera vu en GPA-325 – Introduction à l'électronique

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = A \oplus B$ . Cette expression peut aussi être écrite sous une autre forme :  $F = A \oplus B = A\bar{B} + \bar{A}B$ .

### 2-2-D) La fonction logique « NON-OU-EXCLUSIF »

Soit deux variables booléennes nommées  $A$  et  $B$ . Le résultat de la fonction logique NON(A OU-EXCLUSIF  $B$ ) sera également une variable booléenne. La figure de droite montre la table de vérité de cette fonction. Comme vous pouvez le constater, la sortie de la fonction logique NON-OU-EXCLUSIF (en anglais ENXOR) est l'inverse de celle obtenue avec la fonction logique OU-EXCLUSIF.

Funct "NON-OU-EXC"		
A	B	F
0	0	1
0	1	0
1	0	0
1	1	1

Au niveau algébrique, l'équation correspondant à cette table de vérité est  $F = \overline{A \oplus B}$ . Cette expression peut aussi être écrite sous une autre forme :  $F = \overline{A \oplus B} = AB + \bar{A}\bar{B}$ .

Les fonctions vues jusqu'à présent sont les plus courantes de l'algèbre booléenne. Toutefois, il est bon de savoir que d'autres opérations sont définies, nommées et utilisées en technique numérique. Il y a, en effet, seize fonctions possibles pour deux variables booléennes. Deux variables permettent quatre combinaisons ( $2^2$ ) d'entrées et ces quatre combinaisons donnent seize ( $2^4$ ) combinaisons différentes pour la fonction. Pour retrouver ces seize fonctions, il suffit d'écrire tous les nombres entiers de 0 à 15 dans l'ordre binaire naturel. La table 2-1 montrée en page suivante montre ces 16 fonctions identifiées de  $F0$  à  $F15$ .

Plusieurs de ces fonctions sont très simples. Par exemple,  $F0$  est égale à 0. Toutes ces fonctions peuvent être exprimées au moyen des opérateurs élémentaires définis précédemment. La table 2-2 (toujours en page suivante) montre les expressions des seize fonctions. On y reconnaît le NON ( $F3$  ou  $F5$ ), le ET ( $F8$ ), le OU ( $F14$ ), le NON-ET ( $F7$ ), le NON-OU ( $F1$ ), le OU-EXCLUSIF ( $F6$ ) et le NON-OU-EXCLUSIF ( $F9$ ).

A	B	F0	F1	F2	F3	F4	F5	F6	F7
0	0	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0

A	B	F8	F9	F10	F11	F12	F13	F14	F15
0	0	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1

Table 2-1 - Les seize fonctions possibles associant deux variables d'entrées

$F0 = 0$	$F4 = A\bar{B}$	$F8 = AB$	$F12 = A$
$F1 = \bar{A}\bar{B} = \overline{A+B}$	$F5 = \bar{B}$	$F9 = AB + \bar{A}\bar{B} = \overline{A \oplus B}$	$F13 = A + \bar{B}$
$F2 = \bar{A}B$	$F6 = \bar{A}B + A\bar{B} = A \oplus B$	$F10 = B$	$F14 = A + B$
$F3 = \bar{A}$	$F7 = \bar{A} + \bar{B} = \overline{AB}$	$F11 = \bar{A} + B$	$F15 = 1$

Table 2-2 - Les seize expressions logiques de deux variables d'entrées

Avec trois variables d'entrées, il faudrait traiter 256 fonctions possibles. En effet, trois variables permettent huit combinaisons ( $2^3$ ) d'entrées et ces huit combinaisons donnent 256 ( $2^8$ ) combinaisons différentes pour la fonction. Il est évident qu'il devient non rentable d'énumérer la liste des expressions logiques possibles. D'autres approches seront donc nécessaires et nous verrons plus loin comment déduire l'équation logique à partir d'une table de vérité.

### 2-3) Fonctions logiques matérialisées avec des interrupteurs

Comme il était mentionné au chapitre 1, un interrupteur est un organe binaire pouvant être actionné (lorsque l'opérateur appui dessus), ou non-actionné (lorsque l'opérateur n'y touche pas). Selon le type d'interrupteur, il peut être actionné par un opérateur (bouton de commande), par un mécanisme (interrupteur de fin de course), par la détection de la présence d'une grandeur physique (détecteur de niveau, de température, ...).

## La logique booléenne

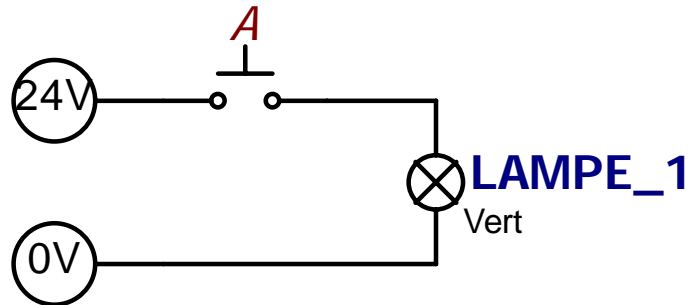
Dans sa position de repos, un interrupteur peut laisser passer ou non le courant électrique. Si l'interrupteur doit être actionné pour qu'il laisse passer le courant, nous avons un interrupteur dit « normalement ouvert ». Par contre, si en actionnant l'interrupteur le courant est coupé (ou ne passe plus), nous avons un interrupteur dit « normalement fermé »

Une ampoule (ou un voyant) est un organe binaire qui peut être actionné (lorsque le courant passe et que l'ampoule brille), ou non-actionné (lorsque le courant ne passe plus et que l'ampoule est éteinte).

Dans les paragraphes qui suivent, nous verrons comment, en utilisant les interrupteurs comme entrées logiques et une ampoule comme sortie obtenir des fonctions logiques.

### 2-3-A) Fonction logique « OUI »

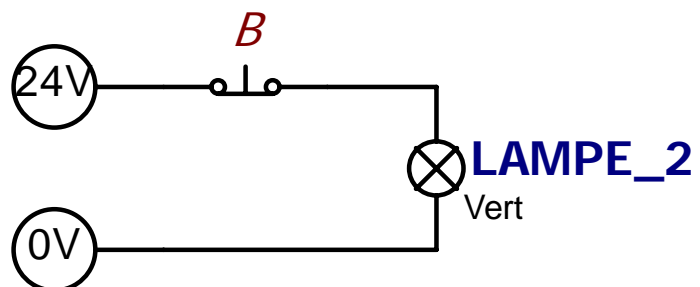
La figure de droite montre la fonction logique OUI. Cette fonction utilise simplement un interrupteur normalement ouvert. Le voyant s'allumera si l'interrupteur est actionné.



La fonction logique de ce montage est :  $Lampe\_1 = A$ .

### 2-3-B) Fonction logique « NON »

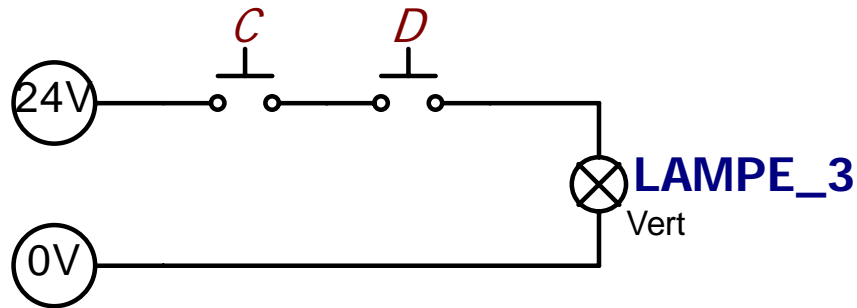
La figure de droite montre la fonction logique NON. Pour obtenir cette fonction, il suffit de brancher un interrupteur normalement fermé. Le voyant s'allume tant que l'interrupteur n'est pas actionné.



La fonction logique de ce montage est :  $Lampe\_2 = \bar{B}$ .

### 2-3-C) Fonction logique « ET »

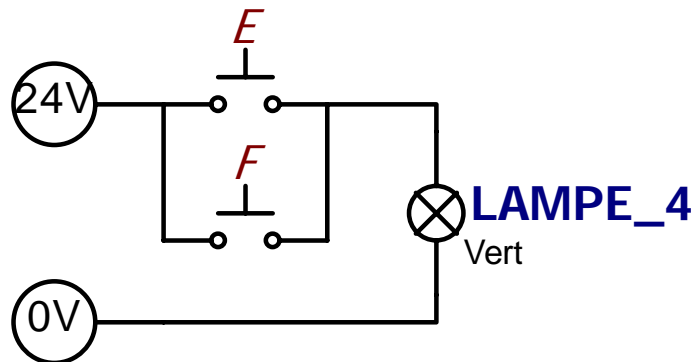
La figure ci-dessous montre la fonction logique ET. Pour obtenir cette fonction, il suffit de brancher deux interrupteurs normalement ouverts en série. Pour que le courant puisse traverser le voyant et l'allume, il faut actionner simultanément les deux interrupteurs.



La fonction logique de ce montage est :  $Lampe\_3 = C \cdot D$ .

### 2-3-D) Fonction logique « OU »

La figure ci-dessous montre la fonction logique OU. Pour obtenir cette fonction, il suffit de brancher deux interrupteurs normalement ouverts en parallèle. Le voyant s'allume dès que l'un des interrupteurs est actionné.

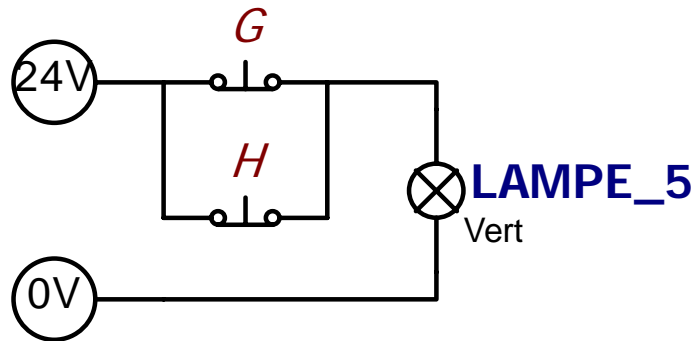


La fonction logique de ce montage est :  $Lampe\_4 = E + F$ .



### 2-3-E) Fonction logique « NON-ET »

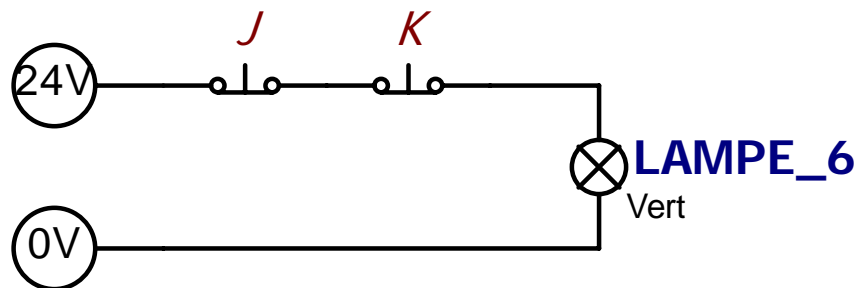
La figure ci-dessous montre la fonction logique NON-ET. Pour obtenir cette fonction, il suffit de brancher deux interrupteurs normalement fermés en parallèle. Le voyant s'éteint seulement si les deux interrupteurs sont actionnés simultanément.



La fonction logique de ce montage est :  $Lampe\_5 = \overline{G \cdot H} = \overline{G} + \overline{H}$ .

### 2-3-F) Fonction logique « NON-OU »

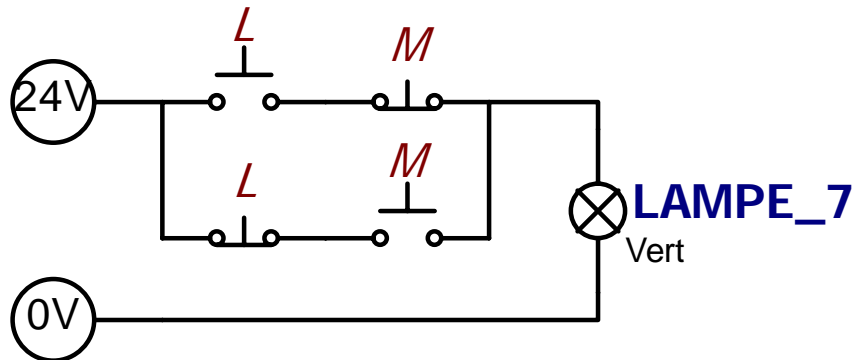
La figure ci-dessous montre la fonction logique NON-OU. Pour obtenir cette fonction, il suffit de brancher deux interrupteurs normalement fermés en série. Le voyant s'éteint dès qu'un des interrupteurs est actionné.



La fonction logique de ce montage est :  $Lampe\_6 = \overline{J + K} = \overline{J} \cdot \overline{K}$ .

### 2-3-G) Fonction logique « OU-EXCLUSIF »

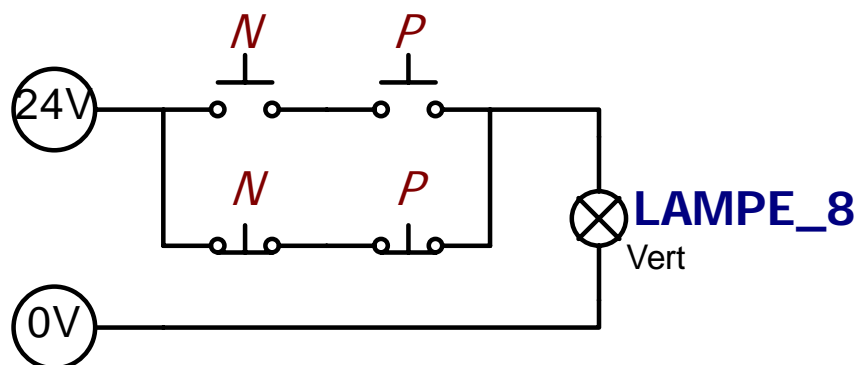
La figure ci-dessous montre la fonction logique OU-EXCLUSIF. Pour obtenir cette fonction, il suffit de brancher, tel que montré, deux interrupteurs ayant deux contacts chacun, l'un normalement ouvert, l'autre normalement fermé. . Le voyant s'allume si et seulement si un seul interrupteur est actionné.



La fonction logique de ce montage est :  $Lampe = L \oplus M = L \cdot \bar{M} + \bar{L} \cdot M$ . Il est à noter que ce montage est appelé un « Three-way » par les électriciens.

### 2-3-H) Fonction logique « NON-OU-EXCLUSIF »

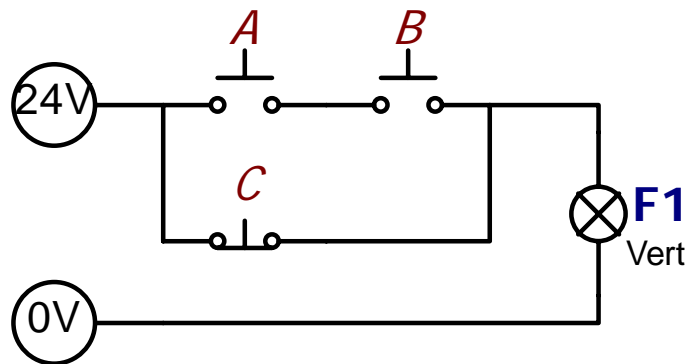
La figure ci-dessous montre la fonction logique NON-OU-EXCLUSIF. Pour obtenir cette fonction, il suffit de brancher, tel que montré, deux interrupteurs ayant deux contacts chacun, l'un normalement ouvert, l'autre normalement fermé. Le voyant s'éteint si et seulement si un seul interrupteur est actionné.



La fonction logique de ce montage est :  $Lampe = \overline{N \oplus P} = \bar{N} \cdot \bar{P} + N \cdot P$ .

## 2-3-l) Fonctions logiques câblées avec des interrupteurs

Les fonctions logiques réalisées avec des interrupteurs sont réalisées de façon fort simple. Toutes les variables associées à des opérateurs logiques « **ET** » correspondent à des interrupteurs **connectés en série**. Toutes les variables associées à des opérateurs « **OU** » correspondent à des interrupteurs **connectés en parallèle**. Une variable complémentée par la fonction **NON** utilise un **interrupteur normalement fermé**. Ainsi, l'équation logique  $F = (A \cdot B) + \bar{C}$  sera représentée par le schéma suivant :



$A$  et  $B$  sont représentés par deux interrupteurs placés en série à cause du ET. Comme ni  $A$ , ni  $B$  sont inversés, ces deux interrupteurs sont normalement ouverts.  $C$  est un interrupteur normalement fermé (à cause du NON) placé en parallèle avec l'ensemble, car la fonction OU est faite avec le résultat de  $A$  ET  $B$ .

O Exercice – série 1 :

Représentez le schéma avec des interrupteurs correspondant aux équations logiques suivantes (sans simplifier) :

a)  $F1 = ABC + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C$

b)  $F2 = (AB + \bar{A}\bar{B})(BC + \bar{C}D)$

c)  $F3 = (A + B + C)(\bar{A} + \bar{B}\bar{C} + C)$

## 2-4) Fonctions logiques matérialisées par des relais

Un relais (figure 2-1) est un organe binaire qui fut la base de la plupart des systèmes automatisés depuis le milieu des années '40. Il comporte une bobine qui servira à générer un champ magnétique permettant de déplacer une lame mobile ouvrant et fermant des contacts internes.

Donc, lorsqu'un courant électrique traverse la bobine, elle émet un champ magnétique qui change la position d'un contact mobile.

Si aucun courant électrique ne traverse la bobine, le relais est dit « *inactif* » et il y a contact entre la

borne commune E et la borne A (figure 2.1). Le contact correspondant au couple de bornes A-E est fermé, et ce contact est nommé « *contact normalement fermé* ». La borne B n'est pas connectée, donc le contact des bornes B-E est ouvert, ce qui nous amène à dire que le contact est B-E est un « *contact normalement ouvert* ».

Si un courant traverse la bobine, le relais est dit « *actif* » et le contact entre la borne commune E et la borne B se ferme et celui entre la borne E et la borne A s'ouvre.

Pour simplifier la représentation, on dessine (sur les plans électriques) le relais de façon schématique (voir figure 2-2).

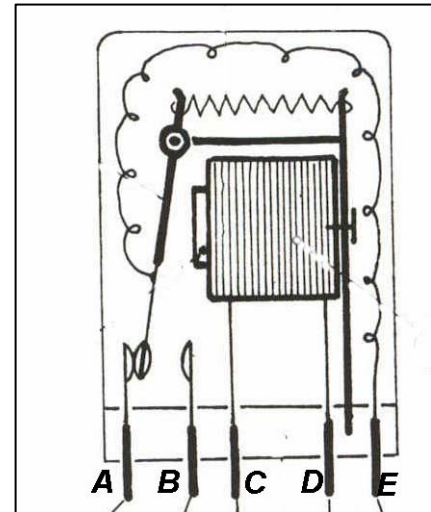


Figure 2-1 - Relais d'automatisme

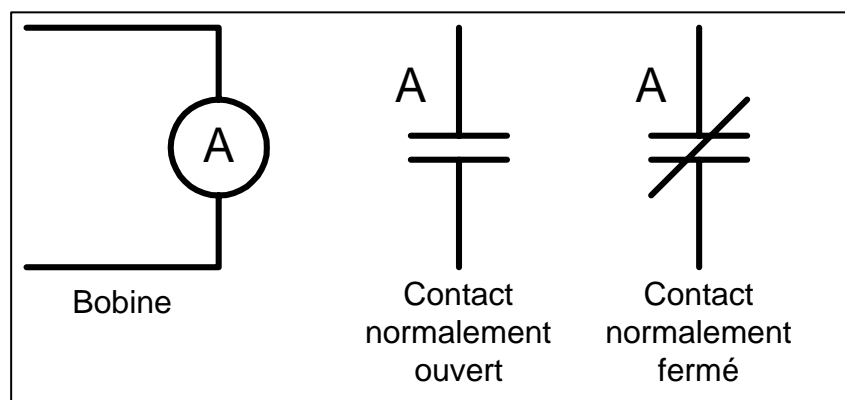
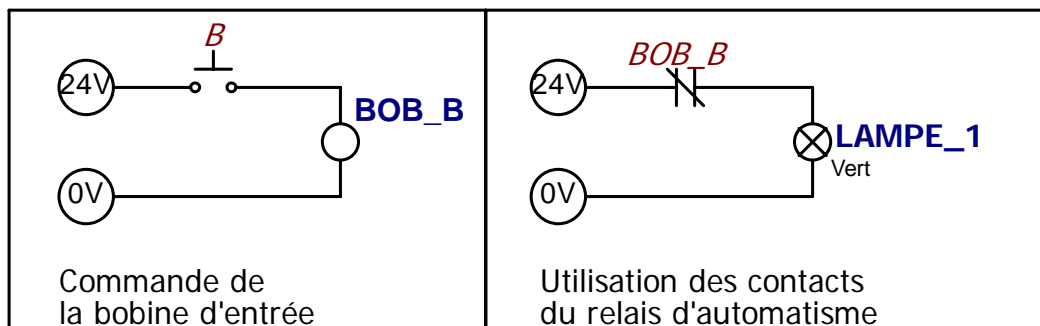


Figure 2-2 - Schéma d'un relais d'automatisme

Tout comme pour les interrupteurs, ce sera par la mise en série ou en parallèle des contacts des relais que nous pourrons obtenir des fonctions logiques.

### 2-4-A) Fonction logique « NON »

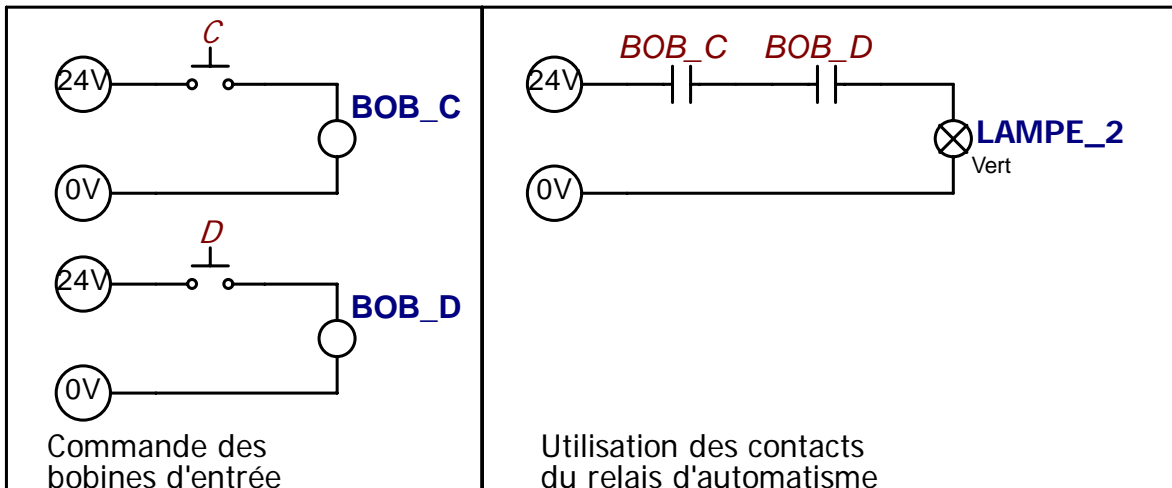
La figure ci-dessous montre la fonction logique NON. Pour obtenir cette fonction, il suffit de brancher le contact normalement fermé du relais *BOB\_B*. L'interrupteur normalement ouvert *B* permet d'activer ou non la bobine du relais *BOB\_B*. Ainsi, si l'interrupteur n'est pas actionné, la bobine du relais n'émet pas de champ magnétique. Donc, le contact normalement fermé reste fermé et l'ampoule sera allumée. Si l'interrupteur est actionné, la bobine du relais émet un champ magnétique, ce qui change la position du contact mobile qui ouvre le contact normalement fermé et éteint l'ampoule.



La fonction logique de ce montage est :  $Lampe_1 = \bar{B}$ .

### 2-4-B) Fonction logique « ET »

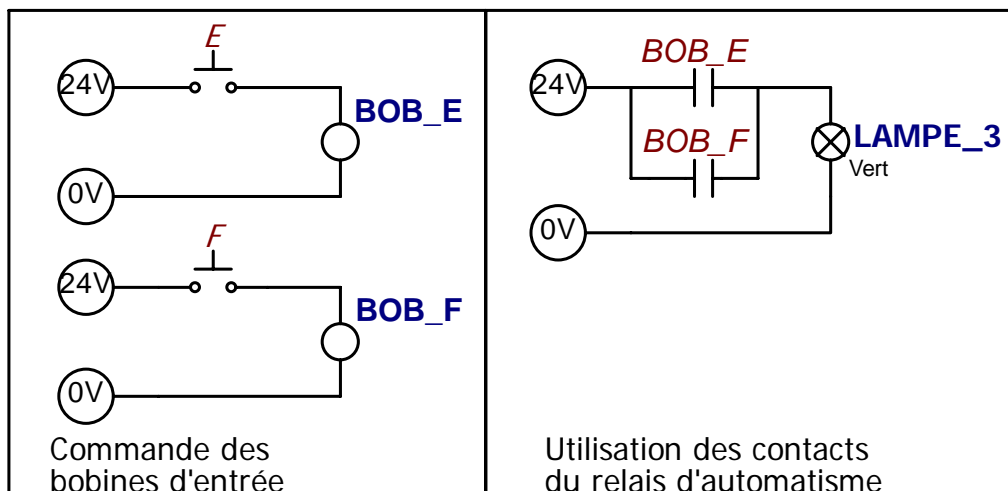
La figure en haut de page suivante montre la fonction logique ET. Pour obtenir cette fonction, il suffit de brancher deux contacts de relais normalement ouverts en série. Pour que le courant traverse l'ampoule, il faut activer simultanément les deux relais.



La fonction logique de ce montage est :  $Lampe\_2 = C \cdot D$ .

### 2-4-C) Fonction logique « OU »

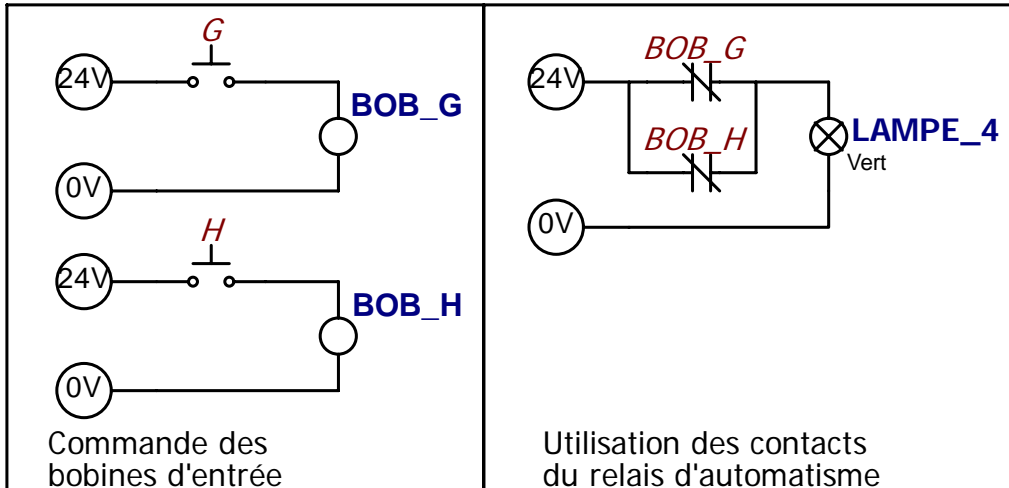
La figure ci-dessous montre la fonction logique OU. Pour obtenir cette fonction, il suffit de brancher deux contacts de relais normalement ouvert en parallèle. Pour que le courant allume l'ampoule, il faut actionner un des relais.



La fonction logique de ce montage est :  $Lampe\_3 = E + F$ .

### 2-4-D) Fonction logique « NON-ET »

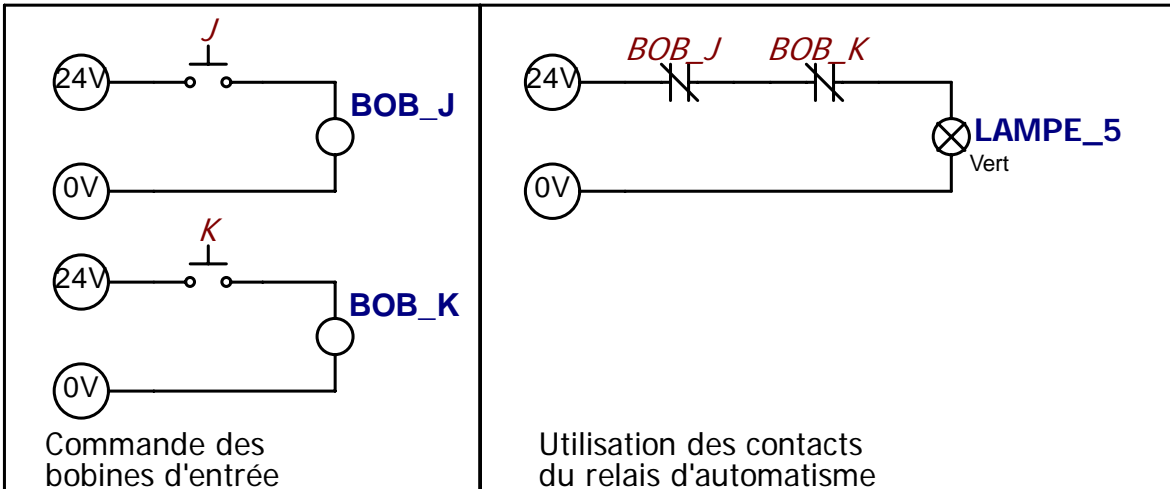
La figure ci-dessous montre la fonction logique NON-ET. Pour obtenir cette fonction, il suffit de brancher deux contacts de relais normalement fermés en parallèle. Pour que le courant cesse de se rendre à l'ampoule et quelle s'éteigne, il faut actionner les deux relais simultanément.



La fonction logique de ce montage est :  $Lampe_4 = \overline{GH} = \overline{G} + \overline{H}$ .

### 2-4-E) Fonction logique « NON-OU »

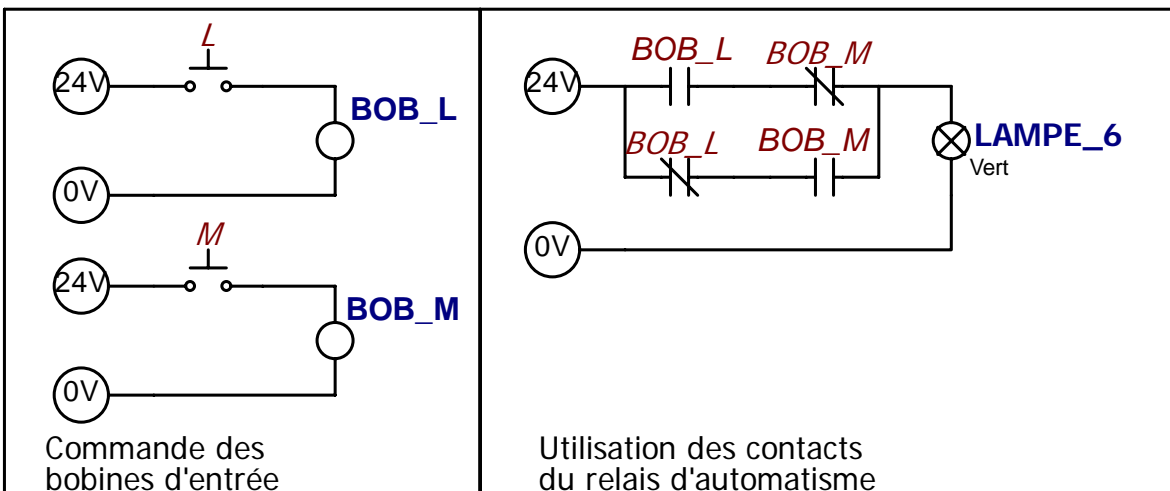
La figure en haut de la page suivante montre la fonction logique NON-OU. Pour obtenir cette fonction, il suffit de brancher deux contacts de relais normalement fermés en série. Pour que le courant cesse de se rendre à l'ampoule et qu'elle s'éteigne, il faut actionner l'un ou l'autre des relais.



La fonction logique de ce montage est :  $Lampe\_5 = \overline{J + K} = \overline{J} \cdot \overline{K}$ .

#### 2-4-F) Fonction logique « OU-EXCLUSIF »

La figure ci-dessous montre la fonction logique OU-EXCLUSIF. Pour obtenir cette fonction, il suffit de brancher, tel que montré, deux relais ayant deux contacts chacun, l'un normalement ouvert, l'autre normalement fermé. Pour que le courant puisse allumer l'ampoule, il ne faut actionner qu'un des relais.

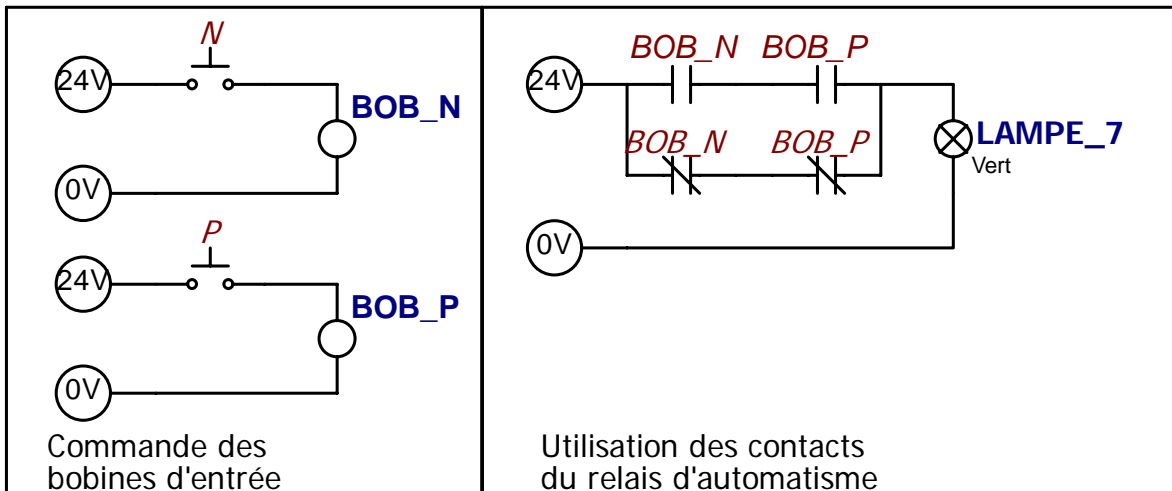


La fonction logique de ce montage est :  $Lampe\_6 = L \oplus M = L \cdot \overline{M} + \overline{L} \cdot M$ .



### 2-4-G) Fonction logique « NON-OU-EXCLUSIF »

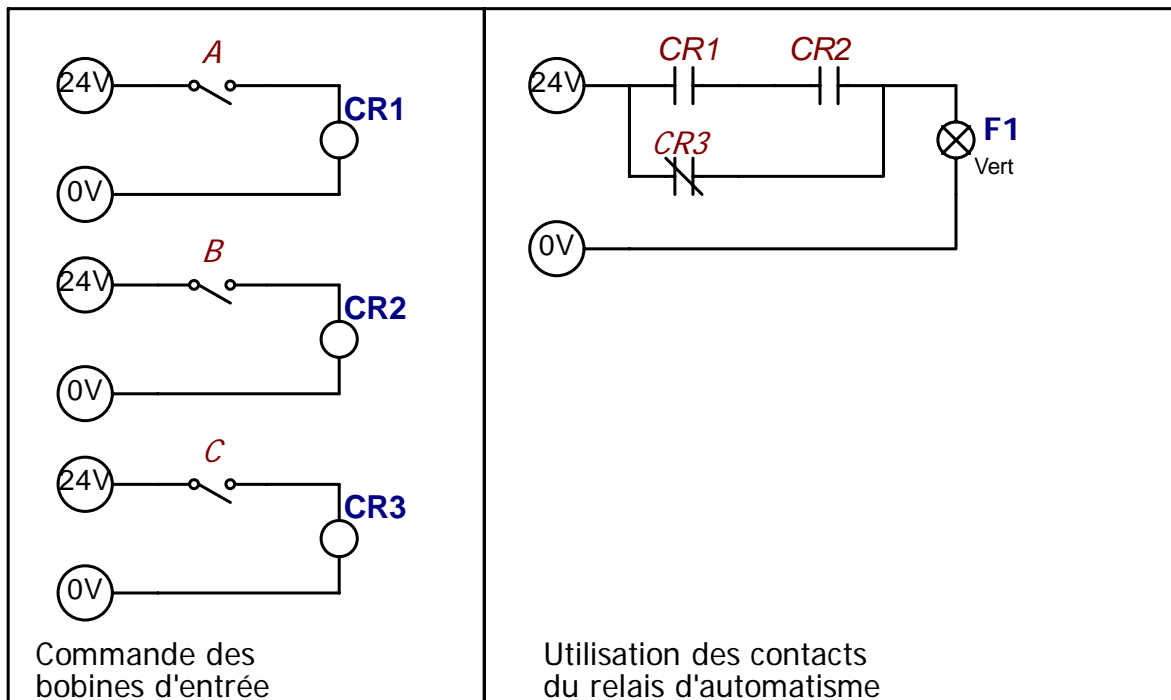
La figure ci-dessous montre la fonction logique NON-OU-EXCLUSIF. Pour obtenir cette fonction, il suffit de brancher, tel que montré, deux relais ayant deux contacts chacun, l'un normalement ouvert, l'autre normalement fermé. Pour que le courant cesse de se rendre à l'ampoule et quelle s'éteigne, il ne faut actionner qu'un des relais.



La fonction logique de ce montage est :  $Lampe = \overline{M} \oplus \overline{N} = \overline{NM} + NM$ .

## 2-4-H) Fonctions logiques câblées avec des interrupteurs

Comme mentionné précédemment, les fonctions logiques réalisées avec des relais le sont de façon similaire à celles réalisées par des interrupteurs. Ainsi, l'équation suivante :  $F1 = (A \cdot B) + \bar{C}$  sera représentée par le schéma suivant :



0 Exercice – série 1 (suite) :

Représentez le schéma avec des relais correspondant aux équations logiques suivantes (sans simplifier) :

- d)  $F1 = ABC + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C$
- e)  $F2 = (AB + \bar{A}\bar{B})(BC + \bar{C}D)$
- f)  $F3 = (A + B + C)(\bar{A} + B\bar{C} + C)$

***En pratique le schéma de commande des bobines d'entrée n'est jamais représenté en logique. Ils ne figurent que sur les plans électriques.***

## 2-5) Les règles et postulats de l'algèbre booléenne

Lorsque l'on désire réaliser le montage d'une fonction logique, il est nécessaire de la simplifier pour faciliter l'analyse et pour minimiser le câblage. C'est exactement ce que l'on fait en algèbre avec des équations. Par exemple, supposons qu'il faut faire le calcul suivant :

$$y = \int \left( \frac{x^2 + 2x + 1}{(x+1)^2} \right) dx$$

La solution devient claire si l'équation est simplifiée, car le calcul revient à faire l'intégrale de 1 et alors la solution est  $y = x + C$ .

La même approche est privilégiée en logique booléenne, mais pour y arriver, il est important de bien comprendre les règles de base de l'algèbre booléenne.

### 2-5-A) Les règles (ou propriétés) de l'algèbre booléenne

Le tableau suivant donne la liste des règles de l'algèbre booléenne :

<b>Fermeture</b>	Si $A$ et $B$ sont des variables booléennes, alors $A+B$ , $AB$ sont aussi des variables booléennes
<b>Commutativité</b>	$A+B = B+A$ $A \bullet B = B \bullet A$
<b>Associativité</b>	$A+(B+C) = (A+B)+C$ $A \bullet (B \bullet C) = (A \bullet B) \bullet C$
<b>Distributivité</b>	ET/OU $A(B+C) = AB+AC$ OU/ET $A+(BC) = (A+B)(A+C)$
<b>Idempotence</b>	$A+A = A$ $A \bullet A = A$
<b>Complémentarité</b>	$A+\bar{A} = 1$ $A \bullet \bar{A} = 0$
<b>Identités remarquables</b>	$1+A = 1$ $1 \bullet A = A$ $0+A = A$ $0 \bullet A = 0$
<b>Distributivité interne</b>	$A+(B+C) = (A+B)+(A+C)$ $A \bullet (B \bullet C) = (A \bullet B) \bullet (A \bullet C)$

On retrouve aussi les 8 postulats définis par Boole :

<b>Postulat 1</b>	$0 \bullet 0 = 0$	<b>Postulat 2</b>	$1 + 1 = 1$
<b>Postulat 3</b>	$0 \bullet 1 = 1 \bullet 0 = 0$	<b>Postulat 4</b>	$0 + 1 = 1 + 0 = 1$
<b>Postulat 5</b>	$1 \bullet 1 = 1$	<b>Postulat 6</b>	$0 + 0 = 0$
<b>Postulat 7</b>	$\bar{0} = 1$	<b>Postulat 8</b>	$\bar{1} = 0$

Voici enfin 22 théorèmes dont certains ont été définis par Boole :

<b>#1</b>	$0 \bullet A = 0$	<b>#2</b>	$1 + A = 1$
<b>#3</b>	$1 \bullet A = A$	<b>#4</b>	$0 + A = A$
<b>#5</b>	$A \bullet A = A$	<b>#6</b>	$A + A = A$
<b>#7</b>	$A \bullet B = B \bullet A$	<b>#8</b>	$A + B = B + A$
<b>#9</b>	$(A \bullet B) \bullet C = A \bullet (B \bullet C)$	<b>#10</b>	$(A + B) + C = A + (B + C)$
<b>#11</b>	$A \bullet \bar{A} = 0$	<b>#12</b>	$A + \bar{A} = 1$
<b>#13</b>	$A \bullet (B + C) = (A \bullet B) + (A \bullet C)$	<b>#14</b>	$A + (B \bullet C) = (A + B)(A + C)$
<b>#15</b>	$\overline{A \bullet B \bullet C \bullet \dots \bullet Z} = \bar{A} + \bar{B} + \bar{C} + \dots + \bar{Z}$	<b>#16</b>	$\overline{A + B + C + \dots + Z} = \bar{A} \bar{B} \bar{C} \dots \bar{Z}$
<b>#17</b>	$A + AB = A$	<b>#18</b>	$A(A + B) = A$
<b>#19</b>	$A + \bar{A}B = A + B$	<b>#20</b>	$A(\bar{A} + B) = AB$
<b>#21</b>	$(A + B)(B + C)(\bar{A} + C) = (A + B)(\bar{A} + C)$	<b>#22</b>	$AB + BC + \bar{A}C = AB + \bar{A}C$

À partir de tous ces éléments, il est possible de procéder à la simplification des fonctions logiques.

Par exemple, voici l'équation suivante :  $F = ABC\bar{C} + A\bar{B}C + A\bar{B}\bar{C} + \bar{A}B\bar{C}$  qui est à simplifier.

En appliquant le théorème #8 sur le dernier terme, on peut le décaler de deux positions et on trouve :  $F = ABC\bar{C} + \bar{A}B\bar{C} + A\bar{B}C + A\bar{B}\bar{C}$ .

En appliquant ensuite le théorème #13 sur les deux premiers et les deux derniers termes, on trouve :  $F = B\bar{C}(A + \bar{A}) + A\bar{B}(C + \bar{C})$ .

Puis, en appliquant le théorème #12 sur les termes entre parenthèses, on trouve :  $F = B\bar{C}(1) + A\bar{B}(1)$ .

Enfin, en appliquant le théorème #3 sur chaque terme, on trouve :  $F = B\bar{C} + A\bar{B}$ .

Donc l'équation simplifiée de F est :  $F = B\bar{C} + A\bar{B}$ .

La simplification d'une fonction logique à un effet significatif sur le câblage, mais n'est malheureusement pas toujours évidente à obtenir. D'autres méthodes, plus faciles, seront abordées ultérieurement pour simplifier les fonctions logiques.

O Exercice – série 1 (suite) :

Simplifier les équations logiques suivantes et les représenter avec relais :

g)  $F1 = ABC + \overline{A}\overline{B}C + A\overline{B}\overline{C} + A\overline{B}C$

h)  $F2 = (AB + \overline{A}\overline{B})(BC + \overline{C}D)$

i)  $F3 = (A + B + C)(\overline{A} + \overline{B}\overline{C} + C)$

### 2-5-B) Théorème de De Morgan

Le théorème de De Morgan est fort important en automatisation car il permet de réaliser des fonctions logiques qui ne seraient pas faisables en utilisant des relais. Il comporte deux parties.

- Premièrement, le complément d'une somme (ou) est égal au produit (et) des compléments. Donc,  $\overline{A + B + C} = \overline{A}\overline{B}\overline{C}$ .

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

- Deuxièmement, le complément d'un produit (et) est égal à la somme (ou) des compléments. Donc,  $\overline{ABC} = \overline{A} + \overline{B} + \overline{C}$ .

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

Les deux tableaux de la page précédente donnent des démonstrations éloquentes sur le théorème de De Morgan.

## 2-6) Représentations des fonctions logiques

Une fonction logique peut être représentée par une équation logique. Elle peut aussi être représentée par une table de vérité. La présente section montrera comment utiliser les tables de vérité. Elle montrera aussi les quatre formes canoniques qui existent lorsqu'une fonction logique est représentée par une équation. Enfin, la transformation d'une fonction logique représentée par une table de vérité vers une représentation par une équation logique (ou vice versa) sera présentée.

### 2-6-A) Les tables de vérité

Une table de vérité est une table qui énumère pour toutes les combinaisons possibles d'entrées, les sorties correspondantes à chacune de ces combinaisons. Cette représentation est la première que l'on construit lors de l'analyse d'un problème de logique.

La taille d'une table de vérité dépend du nombre d'entrées et de sorties. Le nombre de colonnes est égal à la somme du nombre d'entrées et du nombre de sorties. Le nombre de lignes est égal au nombre de combinaisons possibles d'entrées qui est 2 exposant le nombre d'entrées.

Ainsi, une table de vérité ayant trois entrées et deux sorties aura cinq (3+2) colonnes et huit (2<sup>3</sup>) lignes. De même, une table ayant cinq entrées et une sortie aura six (5+1) colonnes et trente-deux (2<sup>5</sup>) lignes.

Il est important de noter que chaque ligne de la table de vérité correspond à une équation logique en fonctions de toutes les entrées. Ainsi, pour une table de 3 entrées, on aurait :

A	B	C	Équation
0	0	0	$\overline{A}\overline{B}\overline{C}$
0	0	1	$\overline{A}\overline{B}C$
0	1	0	$\overline{A}B\overline{C}$
0	1	1	$\overline{A}BC$
1	0	0	$A\overline{B}\overline{C}$
1	0	1	$A\overline{B}C$
1	1	0	$AB\overline{C}$
1	1	1	$ABC$

Il est à remarquer l'ordre dans lequel sont disposés les huit combinaisons possibles des trois entrées. En effet l'ordre est : 000, 001, 010, 011, 100, 101, 110 et 111. Or cette représentation est appelée numérotation binaire et correspond à la numérotation décimale suivante : 0, 1, 2, 3, 4, 5, 6 et 7. Cet ordre de numérotation est intentionnellement choisi afin d'éviter d'oublier une combinaison.

## 2-6-B) Construction d'une table de vérité

À partir d'un problème de logique donné, il faut construire une table de vérité pour être en mesure de trouver les équations logiques de ce problème. Il suffit de prendre un exemple simple pour voir comment construire la table de vérité.

Voici l'exemple<sup>2</sup>. Jean et Robert prévoient faire une partie de tennis le samedi après-midi ; s'il fait beau, ils pourront utiliser le court en plein air qui est toujours disponible ; s'il pleut, ils pourront jouer sur le tennis couvert à condition que le terrain soit libre ; s'ils ne peuvent pas jouer, ils iront au cinéma ; toutefois Robert n'est pas certain de pouvoir disposer de son après-midi ; si Robert n'est pas libre et suivant qu'il pleuvra ou non, Jean ira au cinéma ou fera de la voile.

Il faut établir la table de vérité des activités possibles de Jean, tennis ( $T$ ), voile ( $V$ ), cinéma ( $C$ ) en fonction des variables suivantes  $b$  ( $b = 1$ , il fait beau),  $l$  ( $l = 1$ , le tennis couvert est libre),  $d$  ( $d = 1$ , Robert est disponible).

Il y a donc trois entrées ( $b, l, d$ ) et trois sorties ( $T, V, C$ ), ce qui donne le tableau suivant :

$b$	$l$	$d$	$T$	$V$	$C$
0	0	0	0	0	1
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	1	0	0
1	1	0	0	1	0
1	1	1	1	0	0

Le remplissage se fait en fonction de la description du problème. Si nous décrivons ce qui se passe pour chaque combinaison d'entrées, nous avons :

<sup>2</sup> Exemple tiré du livre "Automatismes à Séquences" de Maurice Milsant, éditions Eyrolles

## La logique booléenne

- 1<sup>ère</sup> ligne, s'il ne fait pas beau, que le tennis intérieur n'est pas libre et que Robert n'est pas disponible, Jean ira au cinéma ;
- 2<sup>ième</sup> ligne, s'il ne fait pas beau, que le tennis intérieur n'est pas libre et que Robert est disponible, Jean ira au cinéma ;
- 3<sup>ième</sup> ligne, s'il ne fait pas beau, que le tennis intérieur est libre et que Robert n'est pas disponible, Jean ira au cinéma ;
- 4<sup>ième</sup> ligne, s'il ne fait pas beau, que le tennis intérieur est libre et que Robert est disponible, Jean ira jouer au tennis ;
- 5<sup>ième</sup> ligne, s'il fait beau, que le tennis intérieur n'est pas libre et que Robert n'est pas disponible, Jean ira faire de la voile ;
- 6<sup>ième</sup> ligne, s'il fait beau, que le tennis intérieur n'est pas libre et que Robert est disponible, Jean ira jouer au tennis ;
- 7<sup>ième</sup> ligne, s'il fait beau, que le tennis intérieur est libre et que Robert n'est pas disponible, Jean ira faire de la voile ;
- Dernière ligne, s'il fait beau, que le tennis intérieur est libre et que Robert est disponible, Jean ira jouer au tennis.

---

### 0 Exercice - série 1 (suite) :

Construire la table de vérité du système suivant :

- j) Soit un local ayant trois portes identifiées  $a$ ,  $b$  et  $c$ . À proximité de chacune de ces portes nous trouvons un interrupteur à bascule que les gens manipuleront lorsqu'ils entreront ou sortiront. Ces interrupteurs commandent une ampoule qui éclaire le local. Ainsi, une personne qui entre par la porte «  $a$  » manipulera l'interrupteur «  $a$  » pour allumer l'ampoule et cette même personne sortant par la porte «  $b$  » manipulera l'interrupteur «  $b$  » pour éteindre l'ampoule. Lors de l'inauguration du local,  $a = 0$ ,  $b = 0$ ,  $c = 0$ , et l'ampoule  $L$  est éteinte ( $L = 0$ ).



## 2-6-C) Les quatre formes canoniques des équations logiques

Les équations logiques peuvent être écrites de quatre façons différentes. Ces quatre façons sont appelées les formes canoniques.

La **première forme canonique** est celle qui présente les équations logiques sous une forme somme de produits. Par exemple l'équation logique  $F = \overline{A}B\overline{C} + A\overline{B}C + A\overline{B}\overline{C} + A\overline{B}C$  est sous cette forme canonique. Cette équation pouvait se simplifier à  $F = A\overline{B} + B\overline{C}$  qui est de forme canonique 1 aussi.

La **seconde forme canonique** est celle qui présente les équations logiques sous une forme produit de sommes. Par exemple, l'équation logique  $F = (A + B + C)(A + B + \overline{C})(A + \overline{B} + \overline{C})(\overline{A} + \overline{B} + \overline{C})$  est de forme canonique 2 et représente le même système logique que l'équation précédente.

Démonstration :

- $F = (AA + AB + A\overline{C} + BA + BB + B\overline{C} + CA + CB + C\overline{C})(A + \overline{B} + \overline{C})(\overline{A} + \overline{B} + \overline{C})$
- $F = (A + AB + A\overline{C} + AB + B + B\overline{C} + AC + BC + 0)(A + \overline{B} + \overline{C})(\overline{A} + \overline{B} + \overline{C})$
- $F = (A + B)(A + \overline{B} + \overline{C})(\overline{A} + \overline{B} + \overline{C})$
- $F = (AA + A\overline{B} + A\overline{C} + BA + B\overline{B} + B\overline{C})(\overline{A} + \overline{B} + \overline{C})$
- $F = (A + B\overline{C})(\overline{A} + \overline{B} + \overline{C})$
- $F = (A\overline{A} + A\overline{B} + A\overline{C} + B\overline{C}\overline{A} + B\overline{C}\overline{B} + B\overline{C}\overline{C})$
- $F = (A\overline{B} + A\overline{C} + B\overline{C}) = A\overline{B} + B\overline{C}$

Donc un même système logique peut être représenté sous ces deux formes canoniques.

La **troisième forme canonique** est celle qui présente les équations logiques sous une forme n'utilisant que des fonctions logiques NON-ET. Cette forme n'est utile que lorsque l'on utilise des circuits électroniques. Cette forme canonique s'obtient directement de la première forme canonique. Il suffit de faire une double négation de l'équation sous forme canonique 1. Par exemple pour l'équation logique  $F = \overline{A}B\overline{C} + A\overline{B}C + A\overline{B}\overline{C} + A\overline{B}C$  le processus est :

- 1) double négation :  $F = \overline{\overline{\overline{A}B\overline{C} + A\overline{B}C + A\overline{B}\overline{C} + A\overline{B}C}}$
- 2) théorème de De Morgan :  $F = \overline{\overline{A}B\overline{C}} \cdot \overline{A\overline{B}C} \cdot \overline{A\overline{B}\overline{C}} \cdot \overline{A\overline{B}C}$

Cette dernière équation est sous forme canonique 3.

La **quatrième forme canonique** est celle qui présente les équations logiques sous une forme n'utilisant que des fonctions logiques NON-OU. Cette forme non plus

n'est utile que lorsque l'on utilise des circuits électroniques. Cette forme canonique s'obtient directement de la seconde forme canonique. Il suffit de faire une double négation de l'équation sous forme canonique 2. Par exemple pour l'équation logique  $F = (A + B + C)(A + B + \bar{C})(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + \bar{C})$  le processus est :

- 1) double négation :  $F = \overline{\overline{(A + B + C)(A + B + \bar{C})(A + \bar{B} + \bar{C})(\bar{A} + \bar{B} + \bar{C})}}$
- 2) théorème de De Morgan :

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

Cette dernière équation est sous forme canonique 4.

### 2-6-D) De la table de vérité aux équations logiques

Étant donné qu'une table de vérité et une équation logique sont deux formes de représentations d'un système logique, il est donc possible de passer de l'une à l'autre. Dans l'élaboration d'un système logique, la première étape est l'obtention de la table de vérité de ce système. La seconde étape est la mise en équation.

Pour mettre une table de vérité en équation, il faut se rappeler qu'à chaque ligne de la table de vérité correspond une équation logique fonction de toutes les entrées de la table (voir en page 2-22). Donc pour obtenir l'équation d'une sortie, il suffit d'écrire une équation ayant comme termes toutes les équations logiques de la table de vérité correspondant aux lignes où la sortie logique est égale à 1.

Reprenons l'exemple des activités de Jean. Nous avons obtenu la table de vérité qui suit :

<i>b</i>	<i>L</i>	<i>d</i>	<i>T</i>	<i>V</i>	<i>C</i>
0	0	0	0	0	1
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	1	0	0
1	1	0	0	1	0
1	1	1	1	0	0

Maintenant, si on désire savoir à quel moment Jean ira au cinéma (sortie C), on pourrait dire qu'il ira au cinéma s'il ne fait pas beau et que le tennis intérieur n'est pas libre et que Robert n'est pas disponible OU s'il ne fait pas beau et que le tennis intérieur n'est pas libre et que Robert est disponible OU s'il ne fait pas beau et que le tennis

intérieur est libre et que Robert n'est pas disponible. Donc en considérant les cas où la sortie est égale à 1 on obtient une somme de produit, ce qui est la forme canonique 1.

En terme d'équation, on aurait donc que  $C = \bar{b}\bar{l}\bar{d} + \bar{b}l\bar{d} + b\bar{l}\bar{d}$  ce qui correspond aux trois équations des trois premières lignes de la table de vérité. Cette équation n'étant pas simplifiée, il faut recourir à un peu d'algèbre comme suit :

- $C = \bar{b}\bar{l}\bar{d} + \bar{b}l\bar{d} + b\bar{l}\bar{d} + bld$
- $C = \bar{b}\bar{l}\bar{d} + \bar{b}l\bar{d} + \bar{b}\bar{l}d + \bar{b}ld$
- $C = \bar{b}l(\bar{d} + d) + \bar{b}\bar{d}(\bar{l} + l)$
- $C = \bar{b}l(1) + \bar{b}\bar{d}(1)$

Donc l'équation simplifiée de C est  $C = \bar{b}l + \bar{b}\bar{d}$ . Donc, les équations tirées directement de la table de vérité ne sont pas simplifiées.

Pour savoir à quel moment Jean aura le pied marin, i. e., qu'il ira faire de la voile (sortie V), on doit analyser la table de vérité. Il ira faire de la voile dans les deux situations suivantes : s'il fait beau et que le tennis intérieur n'est pas libre et que Robert n'est pas disponible OU s'il fait beau et que le tennis intérieur est libre et que Robert n'est pas disponible. Donc, s'il fait beau et que Robert n'est pas là.

L'équation logique est  $V = b\bar{l}\bar{d} + bl\bar{d}$  ce qui se simplifie comme suit :

- $V = b\bar{d}(\bar{l} + l) = b\bar{d}(1) = b\bar{d}$

ce qui ressemble à ce que dit la dernière phrase du paragraphe précédent.

Enfin, Jean ira jouer au tennis avec Robert lorsque l'équation logique suivante sera égale à 1 :  $T = \bar{b}ld + b\bar{l}d + bld$ . L'équation simplifiée est :  $T = ld + bd$ .

***En conclusion, si on désire obtenir les équations logiques sous la forme canonique 1 à partir de la table de vérité :***

- 1) il suffit d'identifier les lignes où les sorties logiques sont égales à 1;***
- 2) écrire chaque terme correspondant aux lignes identifiées en insérant un OU entre chacun.***

---

O Exercice - série 1 (suite) :

À partir de la table de vérité obtenue en j :

- k) Écrire l'équation logique de l'ampoule L.
-

## 2-6-E) De l'équation à la table de vérité

L'opération inverse à celle abordée à la section précédente existe. En effet, nous pouvons passer d'une équation logique à une table de vérité.

La technique est fort simple, il suffit de tester l'équation logique avec toutes les combinaisons d'entrée et de trouver pour chaque combinaison la valeur de la sortie. Un exemple simple montrera ce qu'il en est.

Soit  $F = (A \bullet B) + \bar{C}$ . Pour trouver la table de vérité de F, il faut premièrement connaître le nombre d'entrées. Ici, nous en avons trois, soit A, B et C. Donc la table de vérité devrait avoir normalement quatre (3+1) colonnes et huit ( $2^3$ ) lignes. Pour m'aider, j'ajoute une 5<sup>ème</sup> colonne pour le terme  $(A \bullet B)$ . Ainsi la table résultante sera :

A	B	C	$(A \bullet B)$	F
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

La table de vérité finale sera donc la partie de la table ci-haut mise en gris car le système logique comporte trois entrées et une sortie.

O Exercice – série 1 (suite) :

Faire les tables de vérité des équations logiques suivantes :

l)  $F1 = ABC + \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C$

m)  $F2 = (AB + \bar{A}\bar{B})(BC + \bar{C}D)$

n)  $F3 = (A + B + C)(\bar{A} + \bar{B}\bar{C} + C)$

## 2-7) Techniques de simplification des fonctions logiques

Les tables de vérité permettent de trouver les équations logiques d'un système logique. Malheureusement, ces équations logiques ne sont pas simplifiées et il faut simplifier les équations logiques trouvées à partir de la table de vérité avant de réaliser les fonctions logiques correspondantes avec des relais ou des interrupteurs.

Bien sûr, les règles, postulats et théorèmes de l'algèbre booléenne appliqués à une équation logique mènent à la simplification, mais au prix d'un certain effort. Plus le nombre d'entrées est élevé, plus la simplification est fastidieuse. Deux techniques seront présentées ici pour simplifier les équations logiques de façon plus efficace. Ces deux techniques appelées « *tables de Karnaugh* » et « *tables de Mahoney* » présentent les tables de vérité sous une forme matricielle mettant en évidence les simplifications logiques.

### 2-7-A) Les tables de Karnaugh

La table de Karnaugh est une représentation matricielle de la table de vérité mettant en évidence les simplifications logiques. Pour bien comprendre l'utilité de la table de Karnaugh, il est important de comprendre comment la simplification d'une équation logique est faite.

$$\text{Soit } F_0 = AB + A\bar{B}.$$

Cette fonction se simplifie en mettant A en évidence. Le terme  $(B + \bar{B})$  est égal à 1 et  $F_0 = A$ .

$$\text{Soit } F_1 = ABC + A\bar{B}C + AB\bar{C}.$$

Le processus de simplification est :  $F_1 = ABC + ABC + A\bar{B}C + AB\bar{C}$ , puis  $F_1 = ABC + A\bar{B}C + ABC + AB\bar{C}$ , puis  $F_1 = AC(B + \bar{B}) + AB(C + \bar{C})$ , puis comme les termes  $(B + \bar{B})$  et  $(C + \bar{C})$  sont égaux à 1, alors  $F_1 = AC + AB$ .

Le principe est dès lors évident. Deux termes se simplifient toujours s'ils ne diffèrent que par le fait qu'une variable est présente dans un terme et son inverse dans l'autre terme. Dans  $F_0$ , B et NON B. Dans  $F_1$ , premier et second terme B et NON B ; premier et dernier terme C et NON C.

## La logique booléenne

Voyons maintenant ce qu'il en est avec les tables de vérité. Tout d'abord, celle de  $F0$  :

<b>A</b>	<b>B</b>	<b>F0</b>
0	0	0
0	1	0
1	0	1
1	1	1

Les deux termes sont sur deux lignes successives de la table de vérité. La troisième ligne est identifiée par  $A=1$  et  $B=0$ , la dernière ligne par  $A=1$  et  $B=1$ . Donc, comme seul  $B$  change de valeur et que  $A$  reste toujours égal à 1, cela implique que la solution est  $F0 = A$ .

Passons maintenant à la table de vérité de  $F1$  :

<b>A</b>	<b>B</b>	<b>C</b>	<b>F1</b>
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Les deux dernières lignes de la table de vérité correspondent successivement à  $A=1$ ,  $B=1$  et  $C=0$ , puis à  $A=1$ ,  $B=1$  et  $C=1$ . Conclusion,  $C$  est la seule variable à changer et, comme  $A$  et  $B$  restent au niveau logique 1, cela mène au terme  $AB$ . Donc, ces deux lignes adjacentes de la table de vérité mènent à un terme simplifié.

Pourtant, les 6<sup>ième</sup> et 7<sup>ième</sup> lignes de la table de vérité sont adjacentes et ne font pas apparaître de terme simplifié. Cela vient du fait que deux variables changent d'état ( $B$  et  $C$ ) entre la 6<sup>ième</sup> ligne et la 7<sup>ième</sup> ligne.

De plus, les 6<sup>ième</sup> et 8<sup>ième</sup> lignes ne diffèrent que par la variable  $B$  explique le terme simplifié  $AC$ .

Donc, la table de vérité ne met pas directement en évidence les simplifications possibles car, deux lignes adjacentes ou non peuvent parfois générer un terme simplifié. Tout le génie de Karnaugh est de changer la présentation de la table de vérité pour mettre en évidence les simplifications.

## La logique booléenne

Lorsque le système logique comporte deux variables d'entrées (par exemple A et B), la table de Karnaugh sera une matrice 2x2 comme montrée ci-bas :

	F0		A	0	1
B	0				
1					

Comme la table de Karnaugh est faite de deux lignes et de deux colonnes, les variables d'entrées sont utilisées pour identifier dans quelle case la valeur de la sortie doit apparaître pour chaque combinaison d'entrée. L'une des variables (par exemple A) sert à identifier la colonne où se situe la case à remplir. Si A=0, la case choisie est dans la colonne de gauche et si A=1, la case choisie est dans la colonne de droite. L'autre variable (B) sert à identifier la ligne où se situe la case à remplir. Si B=0, la case choisie est dans la ligne du haut et si B=1, la case choisie est dans la ligne du bas.

Donc, si F0=0 pour A=0 et B=0, il faut inscrire un 0 dans la case située sur la ligne du haut et la colonne de gauche.

La figure qui suit montre la correspondance entre la table de vérité et la table de Karnaugh pour un système logique ayant deux variables d'entrée.

A	B		F0	F0		A	0	1
0	0		●	0		B	0	1
0	1		●	1				
1	0		●	0				
1	1		●	1				

Donc à chaque ligne de la table de vérité correspond une case de la table de Karnaugh. Il y a des gens qui prennent l'habitude de numéroter les cases de la table de Karnaugh pour faciliter le remplissage.

Pour un système logique ayant trois variables d'entrées (A, B et C), la table de Karnaugh sera une matrice 2 (lignes) x 4 (colonnes).

## La logique booléenne

	F1	AB	00	01	11	10
C	0					
1						

Les variables A et B servent à identifier la colonne où se situe la case à remplir. La variable C identifie la ligne de la case à remplir. Il est à noter la façon de numéroter les colonnes. On compte comme suit : 00, 01, 11, et 10, ce qui correspond en décimal à compter 0, 1, 3 et 2. Cette curieuse méthode de compter est nommée « code binaire réfléchi ». Elle fait en sorte que si nous passons d'une colonne à la colonne adjacente, une seule variable à la fois varie. Tout cela pour faire apparaître clairement les simplifications logiques. La colonne de gauche et la colonne de droite varient que d'une seule variable (A) et sont par conséquent considérées comme adjacentes (c'est comme si la table était un cylindre déroulé).

La figure suivante montre la correspondance entre la table de vérité et la table de Karnaugh pour un système logique à trois entrées.

A	B	C	F1	F1	AB	00	01	11	10
0	0	0	●	C	0				
0	0	1	●	1					
0	1	0	●						
0	1	1	●						
1	0	0	●						
1	0	1	●						
1	1	0	●						
1	1	1	●						

Pour un système logique ayant quatre variables d'entrées (A, B, C et D), la table de Karnaugh sera une matrice 4 x 4, telle que montrée en page suivante.

Pour une combinaison logique donnée, correspondant à une case, les variables A et B servent à identifier la colonne et les variables C et D la ligne de cette case. Noter que la première ligne et la dernière ligne sont considérées comme adjacentes car elles ne varient que par la variable C.

Au-delà de quatre variables d'entrées, nous devons faire apparaître une troisième dimension. Ainsi pour un système logique à cinq variables d'entrées (A, B, C, D et E), la table de Karnaugh sera composée de deux tables 4 x 4 superposées.



## La logique booléenne

F2	AB	00	01	11	10
CD					
00					
01					
11					
10					

Il est plus commode de mettre des deux « étages » cote à cote. La table de Karnaugh est montrée ci-dessous.

	A		0		1
F3	BC	00	01	11	10
DE					
00					
01					
11					
10					

La variable A choisi l'étage et les autres la ligne et la colonne de la case correspondante à la combinaison d'entrées choisie. Pour une case donnée, la case de l'étage du dessus ou du dessous est aussi adjacente.

Avec une sixième variable d'entrée nous passerons à une table de Karnaugh à quatre dimensions. Comme l'analyse d'une telle table devient difficile, on aura plutôt recours à la table de Mahoney.

## 2-7-B) La simplification des équations logiques avec les tables de Karnaugh

Le processus de construction et de remplissage d'une table de Karnaugh ayant été défini, il reste à voir comment réussir à obtenir les équations simplifiées à partir de cette table.

La méthode de simplification consiste à **regrouper les 1 adjacents** par association correspondant à des puissances de 2. Il y aura toujours  $2^N$  cases dans une association où N est un entier (on pourra avoir 1,2,4,8,16, ...cases dans une association). Plus l'association est grande, plus le circuit (l'expression algébrique) correspondant est petit.

Pour démontrer ce qui est écrit dans le paragraphe précédent, prenons les deux exemples suivants :  $F0 = ABC + A\bar{B}C + AB\bar{C} + A\bar{B}\bar{C}$  et  $F1 = ABC + A\bar{B}C + AB\bar{C}$ . Le premier exemple comprend quatre termes qui peuvent se simplifier comme suit :

- $F0 = AC(B + \bar{B}) + A\bar{C}(B + \bar{B}) = AC + A\bar{C}$
- $F0 = A(C + \bar{C}) = A$

Donc, quatre termes peuvent être simplifiés en un seul terme. Nous pouvons visualiser cela dans la figure montrée ci-bas où un groupe de quatre termes est encerclé.

F0	AB	00	01	11	10
C	0	0	0	1	1
1	1	0	0	1	1

Le second exemple de trois termes se simplifie comme suit :

- $F1 = AC(B + \bar{B}) + AB(C + \bar{C}) = AC + AB$

Donc, trois termes ne peuvent être simplifiés en un seul terme. Mais, deux termes peuvent se simplifier en un ex. :  $ABC + A\bar{B}C \Rightarrow AC$  et  $ABC + AB\bar{C} \Rightarrow AB$ .

## La logique booléenne

On peut utiliser des termes qui sont déjà couverts par des associations pour en former d'autres impliquant des termes non couverts. Par exemple le terme ABC de l'exemple à trois termes a été utilisé à deux reprises. La figure suivante montre les deux groupes de deux de ce second exemple.

F1	AB			
	00	01	11	10
C				
0	0	0	1	0
1	0	0	1	1

Dans la recherche des 1 adjacents, on doit utiliser les tables comme si elles se refermaient sur elles-mêmes, à la fois horizontalement et verticalement. Ainsi deux 1 situés en bordure du tableau sur une même ligne ou colonne peuvent être considérés comme adjacents. La figure suivante montre un exemple d'un tel groupe.

F2	AB			
	00	01	11	10
C				
0	0	0	0	0
1	1	0	0	1

Les groupes **doivent être** carrés ou rectangulaires. Ils ne peuvent pas être en diagonales. Voici un exemple de ce qu'il ne faut pas faire.

F3	BC			
	00	01	11	10
DE				
00	1	1	0	0
01	0	1	0	0
11	0	1	1	0
10	0	0	0	1

## La logique booléenne

Il est inutile de former des associations n'ayant que des termes qui sont déjà couverts. Ceux-ci ne faussent pas les résultats, mais augmentent inutilement la complexité du circuit (l'expression algébrique). La figure suivante montre un tel exemple.

F4	AB	00	01	11	10
C					
0		0	0	1	1
1		1	0	0	1

Groupe pas utile

Dans certains cas, ce groupe, qui à première vue n'est pas utile, peut être essentiel pour assurer un bon fonctionnement (en électronique)

Maintenant comment trouver l'équation correspondant à un groupe. Il suffit de repérer quelles variables restent constantes au sein du groupe. Par exemple, dans la table de Karnaugh ci-bas, les variables A et C restent constantes (respectivement 1 et 0) dans le groupe et la variable B n'est pas constante.

F5	AB	00	01	11	10
C					
0		0	0	1	1
1		0	0	0	0

Donc, le groupe est indépendant de B, mais dépendant de A et de C. Donc, sachant que  $A=1$  et  $C=0$  dans ce groupe, l'équation du groupe est  $A\bar{C}$ . C'est aussi l'équation de F5, puisqu'il n'y a qu'un seul groupe.

Autre exemple, soit la table suivante :

F6	AB				
	00	01	11	10	
CD	00	0	1	0	0
	01	0	1	0	0
11	0	1	1	0	
10	1	1	1	1	

Diagramme de Karnaugh avec trois groupes de 1s :

- Groupe #1 : une colonne de 1s à AB=01 (cases 00, 01, 11).
- Groupe #2 : une zone de 1s à B=1 et C=1 (cases 01, 11).
- Groupe #3 : une ligne de 1s à D=0 (cases 10).

Le groupe #1 est dépendant des variables A et B (A=0 et B=1), mais indépendant des variables C et D. Donc l'équation du groupe #1 est :  $\bar{A}B$ .

Le groupe #2 est dépendant des variables B et C (B=1 et C=1), mais indépendant des variables A et D. Donc l'équation du groupe #2 est :  $BC$ . Le groupe #3 est dépendant des variables C et D (C=1 et D=0), mais indépendant des variables A et B.

Donc l'équation du groupe #3 est :  $C\bar{D}$ .

Donc l'équation finale pour F6 est :  $F6 = \bar{A}B + BC + C\bar{D}$ .

### 2-7-C) Les tables de Mahoney

La table de Mahoney est une représentation matricielle de la table de vérité fort semblable à la table de Karnaugh. Les différences viennent de la représentation des variables autour de la table et de la structure pour des tables de cinq variables d'entrées ou plus.

Ainsi, lorsque le système logique comporte deux variables d'entrées (par exemple A et B), la table de Mahoney sera une matrice 2x2 comme montrée en haut de page suivante fort semblable à la table de Karnaugh correspondante.

## La logique booléenne

F0	$\bar{A}$	A
B		
0		
1		

Le remplissage de la table de Mahoney se fait de façon identique à celle de la table de Karnaugh.

A	B	F0	F0	$\bar{A}$	A
0	0	●	●	$\bar{B}$	→
0	1	●	●	$\bar{B}$	→
1	0	●	●	B	→
1	1	●	●	B	→

Pour un système logique ayant trois variables d'entrées (A, B et C), la table de Mahoney sera une matrice 2 (lignes) x 4 (colonnes). Elle s'obtient en prenant une table de Mahoney 2x2 et son image miroir.

F1	$\bar{A}$	A	A	$\bar{A}$
$\bar{B}$				
B				

Charnière

Cette image miroir de la table de Mahoney est une réplique de la table 2x2 pivotée autour d'une charnière. Remarquez la disposition des variables A et NON-A. Donc la table résultante est une table 2x4. La variable C servira à identifier si une case est à droite ou à gauche de la charnière. Donc la table de Mahoney 2x4 ressemblera à :

## La logique booléenne

F1	$\bar{A}$	A	A	$\bar{A}$
$\bar{B}$				
B				
	$\bar{C}$		C	

Pour un système logique ayant quatre variables d'entrées (A, B, C et D), la table de Mahoney sera une matrice 4 x 4 obtenue de la matrice 2x4.

F2	$\bar{A}$	A	A	$\bar{A}$	
$\bar{B}$					$\bar{D}$
B					D
<hr/>					
B					D
$\bar{B}$					$\bar{D}$
	$\bar{C}$		C		

La moitié du bas est l'image miroir de la moitié du haut, et la variable D servira à identifier si la case est au-dessus ou au-dessous de la charnière.

Pour un système logique ayant cinq variables d'entrées (A, B, C, D et E), la table de Mahoney sera une matrice 4x8 obtenue de la matrice 4x4.

	$\bar{E}$				E			
F3	$\bar{A}$	A	A	$\bar{A}$	$\bar{A}$	A	A	$\bar{A}$
$\bar{B}$								
B								
<hr/>								
B								
$\bar{B}$								
	$\bar{C}$		C		$\bar{C}$		C	

## La logique booléenne

La moitié de droite est l'image miroir de la moitié de gauche, et la variable E servira à identifier si la case est à gauche ou à droite de la charnière du centre.

Pour un système logique ayant six variables d'entrées (A, B, C, D, E et F), la table de Mahoney sera une matrice 8x8 obtenue de la matrice 4x8.

		$\bar{E}$				E			
F4		$\bar{A}$	A	A	$\bar{A}$	$\bar{A}$	A	A	$\bar{A}$
$\bar{B}$									
B									
$\bar{F}$	B								
	$\bar{B}$								
$\bar{B}$									
B									
F	B								
	$\bar{B}$								
		$\bar{C}$	C	C	$\bar{C}$	C	$\bar{C}$	$\bar{C}$	

Avec plus de six variables d'entrée, le processus d'image miroir se poursuit, mais la résolution devient de plus en plus difficile.

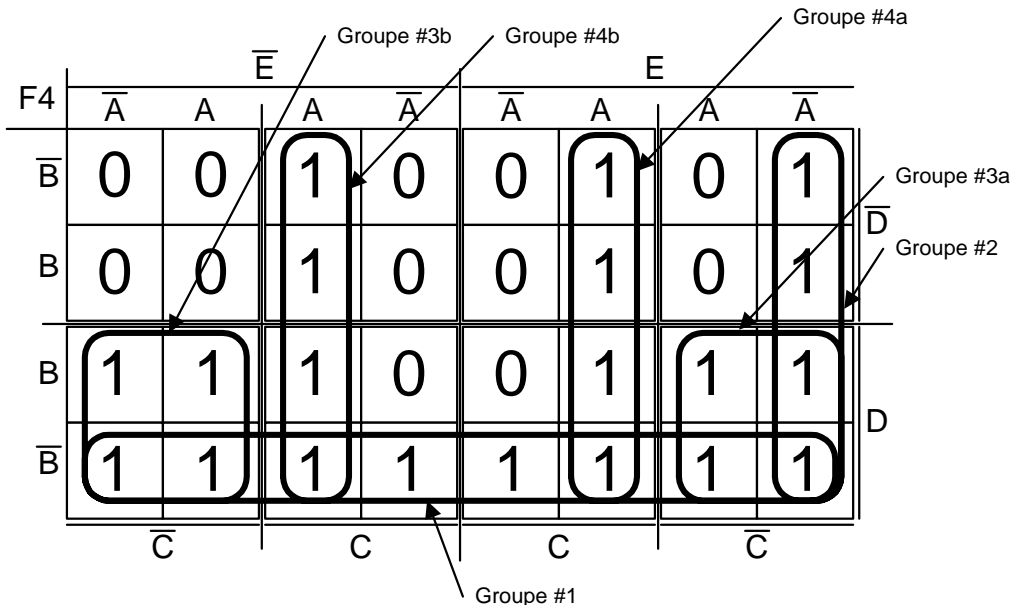
### 2-7-D) La simplification des équations logiques avec les tables de Mahoney

La simplification des tables de Mahoney est très similaire à celle des tables de Karnaugh. Elle repose toujours sur le regroupement des 1 adjacents selon des groupes carrés ou rectangulaires de taille 1, 2, 4, 8, ...

Les différences entre les deux techniques commencent à apparaître avec les tables ayant plus de quatre variables d'entrées car, la structure des tables devient différente.



Par exemple, dans la table 4x8 suivante :



Il semble y avoir six groupes de 1, donc l'équation comporterait six termes. L'équation du groupe #1 est :  $\bar{B}D$ , car B et D sont les seules entrées à rester constante au sein de ce groupe de taille 8. L'équation du groupe #2 est :  $\bar{A}\bar{C}E$ .

Ensuite, l'équation du groupe #3a est :  $\bar{C}DE$  et celle du groupe #3b est :  $\bar{C}D\bar{E}$ . Or ces deux équations peuvent se simplifier, car :  $\bar{C}DE + \bar{C}D\bar{E} = \bar{C}D$ . Ainsi, ces deux groupes de taille 4 sont en fait un seul groupe de taille 8. Cela s'explique par le fait que **ces deux groupes identiques sont placés de façon symétrique** par rapport à la charnière centrale.

Donc les groupes #4a et #4b étant symétriques par rapport à la charnière centrale forment un seul groupe de taille 8 donc l'équation est :  $AC$ .

L'équation de cet exemple est donc  $F4 = \bar{B}D + \bar{C}D + AC + \bar{A}\bar{C}E$ .

### 2-7-E) Les états indifférents (don't care)

Les tables de vérité énumèrent pour toutes les combinaisons d'entrée possibles les valeurs logiques correspondantes des sorties. En théorie tout semble aller de soit, mais en pratique, on peut se retrouver avec des cas bizarres à analyser.

Par exemple, imaginons que nous avons un réservoir équipé de deux détecteurs de niveau. Le premier détecteur « b » placé au bas du réservoir détecte si le réservoir est vide. Si  $b=0$ , le réservoir est vide, si  $b=1$ , le réservoir n'est pas vide. Le second

## La logique booléenne

détecteur « h » placé en haut du réservoir détecte qu'il est plein. Si  $h=0$ , le réservoir n'est pas plein, si  $h=1$ , le réservoir est plein.

Le réservoir peut être rempli par deux pompes « P1 » et « P2 » et le but recherché est de maintenir le réservoir au niveau le plus haut possible, quelle que soit la consommation de liquide. La logique de fonctionnement est la suivante :

- Si le réservoir est plein, les deux pompes sont arrêtées;
- Si le réservoir est vide, les deux pompes sont en marche;
- Si le réservoir n'est ni vide ni plein, seule une pompe marchera et le sélecteur « s » choisira si c'est la pompe P1 ( $s=0$ ) ou la pompe P2 ( $s=1$ ) qui fonctionnera seule.

Le système comporte trois entrées et deux sorties. La table de vérité aura donc cinq colonnes et huit lignes. Mais en y regardant de plus près, on se rend compte que les huit lignes envisagent huit cas possibles et il serait bon de voir quelle est la signification de chaque cas.

<i><b>b</b></i>	<i><b>h</b></i>	<i><b>s</b></i>	<i><b>Description</b></i>
0	0	0	Réservoir vide, faire fonctionner les deux pompes.
0	0	1	Réservoir vide, faire fonctionner les deux pompes.
0	1	0	<b>b indique réservoir vide, mais h indique réservoir plein. Cas impossible.</b>
0	1	1	<b>b indique réservoir vide, mais h indique réservoir plein. Cas impossible.</b>
1	0	0	Réservoir ni vide ni plein, faire fonctionner pompe P1.
1	0	1	Réservoir ni vide ni plein, faire fonctionner pompe P2.
1	1	0	Réservoir plein, donc aucune pompe en fonction.
1	1	1	Réservoir plein, donc aucune pompe en fonction.

Comme le tableau ci-dessus l'indique, deux des huit cas ne sont pas physiquement possibles car, un réservoir ne peut être à la fois plein et vide. Puisqu'il est impossible que ce cas se produise, l'état des sorties P1 et P2 est indifférent. Dans cette situation, que les pompes soient en marche ou non importe peu car elle est impossible.

Pour montrer que *l'état des sorties* importe peu, on place la valeur **X** dans la table de vérité. La table de vérité du système de réservoir est montrée en haut de la page suivante et on constate les X correspondants aux états indifférents car les combinaisons correspondantes des variables d'entrées sont physiquement impossibles.

<i>b</i>	<i>h</i>	<i>s</i>	<i>P1</i>	<i>P2</i>
0	0	0	1	1
0	0	1	1	1
0	1	0	X	X
0	1	1	X	X
1	0	0	1	0
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

Ces états indifférents se retrouvent aussi dans les tables de Karnaugh et de Mahoney tirées des tables de vérité. Par exemple, si nous construisons la table de Karnaugh de la pompe P1, nous aurons :

P1	bh	00	01	11	10
s	0	1	X	0	1
1	1	1	X	0	0

Pour trouver les équations logiques, on forme des groupes de 1. Mais comme les X sont des états indifférents, ils peuvent être considérés soit comme des 0 ou soit comme des 1. Comme il est profitable de faire les plus gros groupes que possible, il y a donc certains avantages à inclure des X pour grossir un groupe. Ainsi, dans la table de Karnaugh, nous aurions un groupe de taille 4 (incluant les deux X) et un groupe de taille 2.

P1	bh	00	01	11	10
s	0	1	X	0	1
1	1	1	X	0	0

Ainsi, la pompe P1 aura comme équation logique de fonctionnement :  $P1 = \bar{b} + \bar{s}h$ . De même, pour la pompe P2, la table de Karnaugh (montrée en haut de la page suivante) mène à  $P2 = \bar{b} + sh$ .

P2	bh	00	01	11	10
s	0	1	X	0	0
	1	1	X	0	1

Les équations pour les pompes P1 et P2 sont donc plus simple grâce aux états indifférents. En contre partie, il faut quand même s'assurer que le système logique résultant reste sécuritaire. Si c'est le cas on conserve les équations que l'on vient de trouver, sinon il faut refaire l'analyse en forçant les états indifférents en cause à avoir la valeur logique 0.

Par exemple, avec le système de pompage, on se rend compte que si le détecteur b reste à 0 parce qu'un fil est coupé, une pompe se mettra en marche, mais cela n'est pas causé par l'utilisation des états indifférents, mais par un mauvais choix de détecteur pour le niveau bas. Si le détecteur h reste à 0 parce qu'un fil est coupé, une pompe se mettra en marche encore à cause d'un mauvais choix de détecteur.

Donc, le système est non sécuritaire et il est préférable de considérer les X comme des 0 et de refaire les équations pour P1 et P2. Toutefois, l'analyse du système indique qu'il serait très préférable de changer les niveaux logiques générés par b et h pour améliorer la sécurité.

Les états indifférents utilisés en entrée permettent de simplifier les tables de vérité. Par exemple, dans le système de réservoir, l'état de « s » ne nous importe que lorsque le réservoir est ni vide et ni plein. Dans les autres cas, l'état de « s » ne nous importe peu. Ainsi, la table de vérité du système de pompage aurait pu être faite comme suit :

<i>b</i>	<i>h</i>	<i>s</i>	<i>P1</i>	<i>P2</i>
0	0	X	1	1
0	1	X	X	X
1	0	0	1	0
1	0	1	0	1
1	1	X	0	0

Cette table plus compacte couvre quand même les huit cas possibles car la première ligne correspond aux deux cas suivants : b=0, h=0, s=0 et : b=0, h=0, s=1.