TRAITER LES INFORMATIONS LOGIQUES

I/ INTRODUCTION

On entend par ***traitement d'information logique***, l'ensemble des opérations destinées à générer un ordre ou une commande à partir de la prise en compte d'un ensemble de consignes claires et ne présentant aucune ambiguïté.

Mise en application :

Un store automatique s'ouvre lorsque :

•il est fermé **ET** qu'il n'y a pas de VENT **ET** que
- •l'utilisateur presse sur le bouton d'ouverture
- •**OU** lorsqu'il y a du soleil

Il se referme :

•lorsque qu'il est ouvert **ET** que
- •l'utilisateur presse sur le bouton de fermeture
- •**OU** qu'il y a du vent

vent

pressions mécaniques

soleil

Traiter les informations logiques... afin de provoquer l'ouverture et la fermeture automatique du store

OUVRIR

FERMER

•Le capteur de vent restitue une information de type ***Présence vent*** ou ***absence de vent***

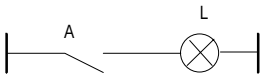
•Le capteur de luminosité et sa structure de mise en forme restitue une information ***luminosité suffisante*** ou non

•Les boutons du boîtier de commande sont appuyés ou non

Ces trois informations (consignes) sont des ***informations logiques*** (claires et sans ambiguïtés). Les commandes de sortie sont elles aussi des informations logiques.

II/ DÉFINITIONS

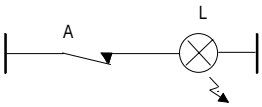
On nomme **VARIABLE BINAIRE** tout phénomène qui ne peut prendre que deux états :
- •**L'état logique 0** peut être associé à une affirmation fausse : absence de tension, actionneur non commandé, etc...
- •**L'état logique 1** peut être associé à une affirmation vraie : tension présente, actionneur commandé, présence d'un phénomène, etc...



Trouvez l'état logique des variables binaires ***A*** et ***L***.

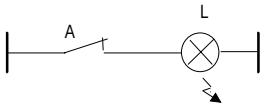
A est un interrupteur ouvert au repos et ***L*** est une lampe.

A= ___ L=_____



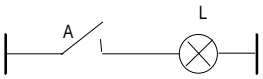
A=____ L=_____

Pour les schémas ci-contre, ***A*** est un interrupteur fermé au repos.

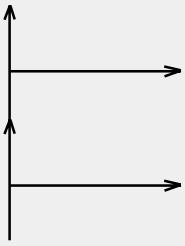


A= ___ L=___

A=____ L=_____



Tension logique : c'est une tension qui ne peut prendre que deux valeurs différentes.



Tension analogique : c'est une tension qui peut prendre une multitude de valeurs.

On appelle **NIVEAU LOGIQUE,** en électronique ou en automatisme, une tension correspondant à un état logique.

- Ainsi la tension la plus élevée d'un circuit logique est généralement associée à l'état logique "1" : On dira qu'il s'agit du niveau logique "1" (NL1).
- Par opposition la tension la plus faible (le 0v souvent) est appelée niveau logique "0" (NL0).

On appelle **OPERATEUR LOGIQUE** un opérateur mathématique (*mis à jour par le mathématicien Georges BOOLE 1815-1864*) qui permet de lier des variables binaires, en vue de décrire avec plus de précision un problème. En principe il n'existe que 3 opérateurs de base :

- ET
- OU
- NON

Grâce à ces trois opérateurs il est possible de décrire un problème simple par des équations.

L'opérateur *ET* (*AND* en anglais) est représenté dans une équation par le caractère *point* "." : **A *ET* B s'écrit A . B** et se lit **A ET B**

L'opérateur *OU* (*OR* en anglais) est formalisé par le caractère plus "+" : **A *OU* B s'écrit A + B** et se lit **A OU B**

L'opérateur *NON* se représente en surlignant la variable binaire ainsi **NON A s'écrit $\overline{A}$** et se lit **A barre.** Quelques fois **A barre** s'écrit également ***/*A**

Par extension, le terme opérateur logique a été associé à des composants électroniques capables de réaliser ces opérations logiques. Ces circuits sont également appelés **PORTES LOGIQUES.**

Fonction combinatoire : correspond à un traitement logique tel que l'information de sortie (logique) est une combinaison linéaire des informations d'entrée. Exemple : *Pour que le coffre fort s'ouvre, les 4 boutons doivent être sur la bonne position*

Fonction séquentielle : correspond à un traitement logique tel que l'information de sortie (logique) dépend à la fois d'un état précédent (effet de mémoire) et des informations d'entrée. Exemple : *Pour que le coffre fort s'ouvre, le bouton doit se mettre dans des positions précises en respectant un ordre chronologique.*

III/ DESCRIPTION D'UN PHÉNOMÈNE LOGIQUE

Il existe plusieurs manières de décrire un phénomène logique :

III.1/ Par l'équation

Chaque variable de sortie est définie par une équation comprenant les variables binaires d'entrée et les opérateurs.

Reprenons l'exemple du store automatique :

OUVRIR=FERME . (BP_OUVRIR + SOLEIL). /VENT

FERMER= _____

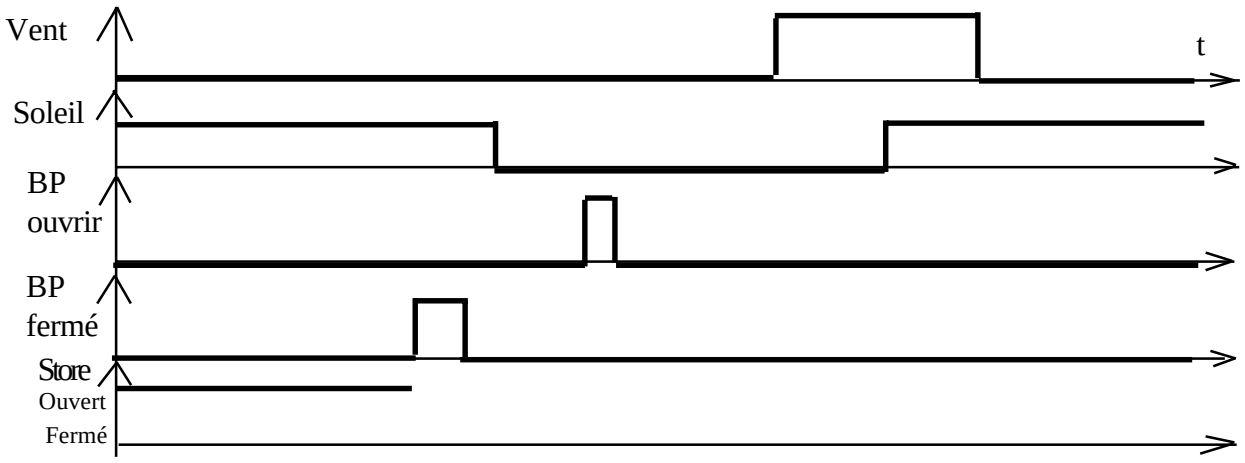
Remarques :

•Les équations s'appliquent mal pour des fonctions séquentielles (comment écrit-on état précédent ?)

•Certains programmateurs de circuits logiques programmables utilisent les équations logiques.

III.2/ Par un chronogramme

Un chronogramme est une représentation graphique en fonction du temps des variables binaire.



Remarque :

•Un chronogramme ne reprend pas forcément toutes les combinaisons logiques

•il convient aussi bien pour les fonctions combinatoires et séquentielles

III.3/ Par table de vérité

Une table de vérité est un tableau reprenant l'ensemble des combinaisons possibles des variables binaires d'entrée et décrivant les états logiques des variables de sortie. L'évolution des lignes doit respecter le code binaire naturel (pour ce code voir le paragraphe IV).

Sur une même ligne, les variables sont reliées entre elles par une fonction ***ET*** alors que l'expression d'une ligne est reliée à l'expression d'une autre ligne par la fonction ***OU***.

Vent	Soleil	BP ouvrir	Store ouvert	Ouvrir Store
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	

IV/ LES CODES

IV.1/ Code binaire naturel

Le code binaire naturel respecte la numération en binaire. Comme le système décimal, le système binaire se dit à *poids positionnels* car la valeur de chaque chiffre dépend de son rang. La valeur correspond au chiffre multiplié par 2^n où *n* est le rang du chiffre: $\mathbf{V=a_n.2^n+a_{n-1}.2^{n-1}+...+a_2.2^2+a_1.2^1+a_0.2^0}$

Poids	3	2	1	0
Valeur	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5				
6				
7				
8				

9				
10				
11				
12				
13				
14				
15				

Chaque chiffre binaire en anglais est appelé *binary digit* d'où la contraction ***bit***.

Le bit de poids le plus faible est appelé **LSB** et le bit de poids le plus fort **MSB**.

IV.2/ Code binaire réfléchi ou code de Gray

En automatisme deux variables binaires ne peuvent pas changer en même temps. De ce fait, le code binaire ne peut pas convenir pour une description chronologique exacte. On utilise alors le code de Gray, qui est un code non pondéré (les chiffres ne sont pas affectés à un poids) et pour lequel seul un chiffre change d'état à la fois.

Décimal	Gray								
0	0	0	0	0		8	1	1	0
1	0	0	0	1		9	1	1	0
2	0	0	1	1		10	1	1	1
3	0	0	1	0		11	1	1	1
4	0	1	1	0		12	1	0	1
5	0	1	1	1		13	1	0	1
6	0	1	0	1		14	1	0	0
7	0	1	0	0		15	1	0	0

On remarque que même le passage de la valeur 0 à la valeur 15 ne provoque qu'un seul changement.

IV.3/ Le code BCD (décimal codé en binaire)

Il s'agit d'un code adapté à la numération décimal. Chaque chiffre décimal est codé en 4 bits en respectant le code binaire naturel. De ce fait, en BCD, on ne compte que de 0 à 9.

Par exemple :

1983 en décimal s'écrit 0001 1001 1000 0011 en BCD

Ce code est particulièrement utilisé pour faciliter l'***interfaçage machine-homme.***

V/ ALGÈBRE DE BOOLE

L'algèbre de Boole permet de résoudre et en particulier de simplifier des problèmes logiques. Il repose sur un ensemble de propriétés qu'il est nécessaire de connaitre.

V.1/ Propriétés

Commutativité	A + B = B+ A	A . B = B . A	
Distributivité	A . (B + C) = A.B + A.C		
Elément neutre	A.1 = A	A+0 = A	
Antisymétrie	A.0 = 0	A+1 = 1	
Identités remarquables	A.A= A	A+A=A	/(/A) = A
	A./A = 0	A+/A = 1	
	A+A.B = A.(1+B) =A		
	A+/A.B = A+B		

Théorèmes de DE MORGAN

$\overline{A.B} = \overline{A}+\overline{B}$ $\overline{A+B} = \overline{A}.\overline{B}$

C'est grâce à ces propriétés que des équations complexes peuvent être simplifiées.

Exemple : Soit la table de vérité suivante :

A	B	C	S
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

L'équation de S est donc :

S=/A.B./C + /A.B.C + A./B.C+A.B./C + A.B.C

Simplifions :

S=B./C.(/A+A) + B.C.(A+/A) + A./B.C

=> S=B./C + B.C + A./B.C

=>S=B.(C+/C) + A./B.C

=>S=B + A./B.C

=>**S=B + AC**

Dans la pratique cette méthode est assez hasardeuse et difficile à utiliser au-delà de 3 variables d'entrées.

V.2/ Simplification par tableau de KARNAUGH

Un tableau de Karnaugh compte 2^n cases, si *n* est le nombre de variables binaires présentent dans l'équation ou la table de vérité. A chaque case correspond un état logique des *n* variables.

Ainsi pour un problème à 3 variables, l'équation des variables dans chaque case se présente de la manière suivante :

On remarque l'utilisation d'un code binaire réfléchi (le changement de colonne n'implique que le changement d'une seule variable à la fois)

Pour une même case, l'opérateur logique entre chaque variable est le ET.

D'une case à l'autre l'opérateur logique est le OU.

Pour notre exemple précédent, le tableau serait complété de la manière suivante :

C'est grâce au regroupement de cases que l'on arrive à simplifier le problème.

C \ AB	0 0	0 1	1 1	1 0
0	/A/B/C	/A B /C	A B /C	A /B /C
1	/A /B C	/A B C	A B C	A /B C

C \ AB	0 0	0 1	1 1	1 0
0	0	1	1	0
1	0	1	1	1

Règles de regroupement :

•Un regroupement ne peut être constitué que d'un nombre de cases de puissance de 2 (2, 4, 8, 16...)

•Les cases regroupées doivent être adjacentes et contenir la valeur 1 (ceci dans le cas ou l'on souhaite connaître l'équation donnant la valeur de sortie à 1). Il est à remarquer que les cases extrêmes du tableau sont adjacentes entre elles (colonne 00 adjacent avec colonne 10, etc...)

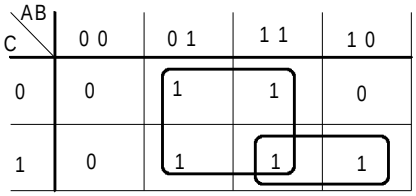
•Les regroupements en diagonal ne sont pas admis

•L'équation d'un regroupement correspond à l'équation logique liant les variables ne changeant pas d'état dans tous le regroupement.

•Le regroupement peut se faire également par les bords du tableau

Observons les regroupement du problème précédent:

•Les deux cases regroupées entre elles sont adjacentes horizontalement. Leur équation est :



A.B.C + A./B.C=A.C.(B+/B)=A.C

On remarque que l'équation finale de ce regroupement ne comporte que les variables A et C qui ne changent pas d'état dans le regroupement.

•Les quatre cases regroupées sont adjacentes horizontalement et verticalement. Pour ces 4 cases la seule variable qui ne change pas est B. L'équation de ce regroupement est donc B et l'équation finale du problème est : B + A.C

C'est à partir de cette équation que l'on pourra réaliser la structure matérielle.

VI/ RÉFÉRENCES DOCUMENTAIRES

•**Memotech :** Initiation aux sciences de l'ingénieur : page 83 et suivantes

•**Memotech :** Sciences de l'ingénieur page 281

•**Circuits numériques** de Ronald TOCCI Ed. Dunod

•**Scuiences de l'ingénieur** HACHETTE livre de première page 130

vent

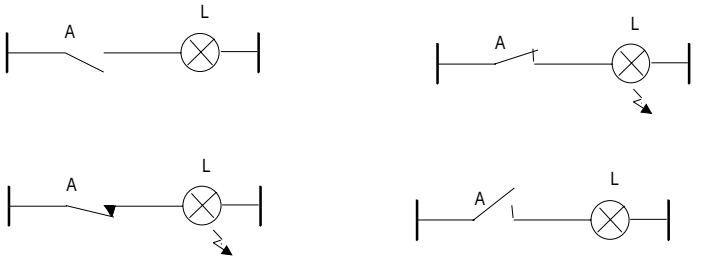
pressions mécaniques

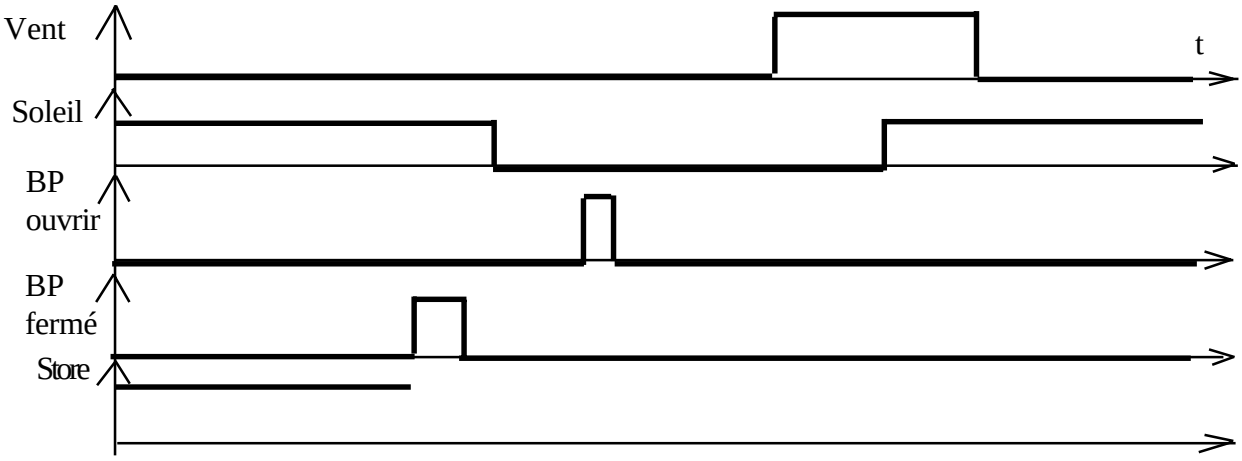
soleil

Traiter les informations logiques... afin de provoquer l'ouverture et la fermeture automatique du store

Ouvrir

Fermer





Décimal	Gray								
0	0	0	0	0		8	1	1	0
1	0	0	0	1		9	1	1	0
2	0	0	1	1		10	1	1	1
3	0	0	1	0		11	1	1	1
4	0	1	1	0		12	1	0	1
5	0	1	1	1		13	1	0	1
6	0	1	0	1		14	1	0	0
7	0	1	0	0		15	1	0	0

Vent	Soleil	BP ouvrir	Store ouvert	Ouvrir Store
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	

Poids	3	2	1	0
Valeur	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				

A	B	C	S
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

C \ AB	0 0	0 1	1 1	1 0
0	/A/B/C	/A B /C	A B /C	A /B /C
1	/A /B C	/A B C	A B C	A /B C

C \ AB	0 0	0 1	1 1	1 0
0	0	1	1	0
1	0	1	1	1