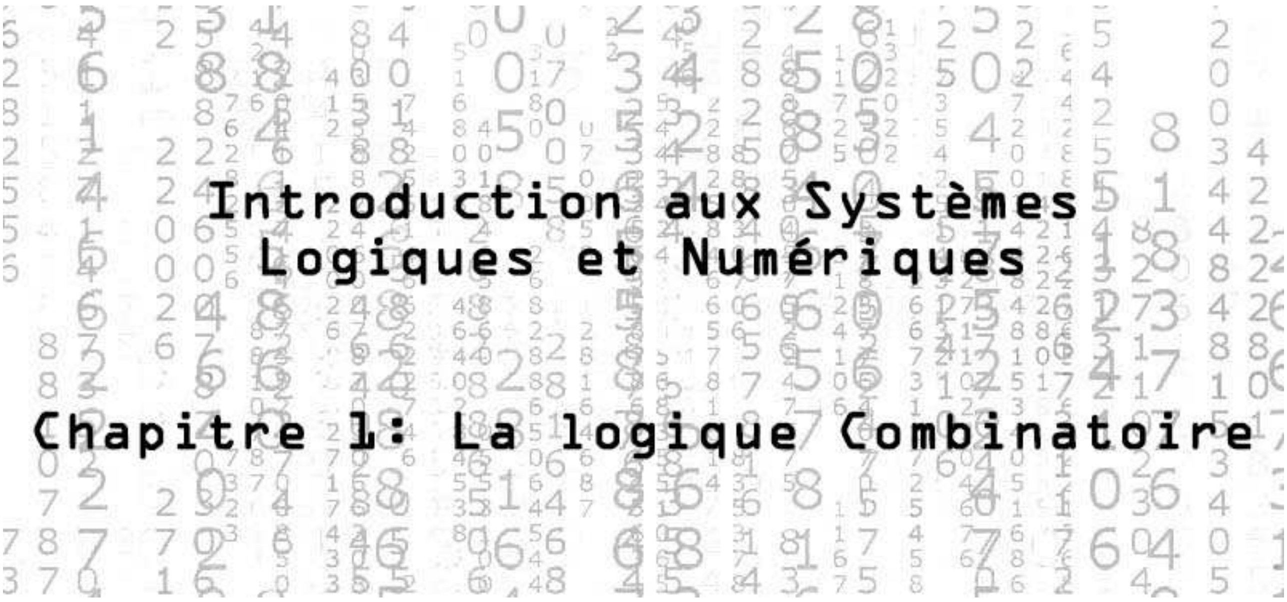


V. Laurain

vincent.laurain@univ-lorraine.fr

Table des matières

1 Différents types de signaux 5
1.1 Les variables analogiques 5
1.2 Les variables numériques 5
1.3 Les variables logiques ou binaires 5
1.4 Les systèmes 5
1.5 But du cours 7

2 Les représentations utilisées en logique combinatoire 8
2.1 La représentation Electrotechnicienne : Le schéma contact 8
2.2 La représentation Electronicienne : Le logigramme 8
2.3 La représentation mathématique: équations et table de vérité 9
2.3.1 Les équations logiques 9
2.3.2 Les tables de vérité 9

3 L'algèbre de Boole 11
3.1 Un peu d'histoire 11
3.2 Les éléments essentiels de l'algèbre de Boole. 11
3.2.1 L'ensemble de Boole. 11
3.2.2 Les opérateurs de Boole. 12
3.3 Description des différentes opérations dans toutes les représentations logiques 12
3.3.1 L'égalité 12
3.3.2 L'Union 13
3.3.3 L'Intersection 13
3.3.4 La négation ou complémentarité 14
3.3.5 Commentaires 14
3.4 Exercices de conversions entre représentations 15
3.5 Transformer une table de vérité en équation: Equation canonique disjonctive 16
3.5.1 Exercices 18

4 Simplification par l'Algèbre de Boole 18
4.1 Propriétés de l'Algèbre de Boole 18
4.2 Théorèmes de base 20
4.3 Autres théorèmes utiles 20
4.4 Théorèmes de De Morgan 21
4.4.1 Utilisation de De Morgan: L'équation Canonique conjonctive 21
4.5 Exercices de simplification 22

5 Conclusions Préliminaires 24

6 Simplification graphique d'équations logiques: Le tableau de Karnaugh 24
6.1 Le code gray 25
6.2 Les tableaux de Karnaugh 26
6.2.1 Construction d'un tableau de Karnaugh 26
6.2.2 Remplissage d'un tableau de Karnaugh à l'aide d'une table de Vérité 27
6.2.3 Remplissage d'un tableau de Karnaugh à l'aide d'une équation 28
6.3 Simplifier à l'aide des tableaux de Karnaugh 29
6.3.1 Les implicants 29
6.3.2 Etape 1: Trouver les implicants premiers 30
6.3.3 Etape 2: Trouver les implicants premiers essentiels 32

6.3.4 Trouver l'équation minimale de sortie 33
6.4 Résumé de la méthode de Karnaugh 35

7 Synthèse de problèmes logiques **36**
7.1 Problèmes avec cas impossibles ou facultatifs 38
7.2 Exercice de synthèse de problèmes logiques 39

8 Conclusion **40**

Liste d'exercices

1.1 Exercice (Système analogiques) . . . 5
1.2 Exercice (Système logique) . . . 6
1.3 Exercice (combinatoire ou séquentiel?) . . . 6
1.4 Exercice (Exemple d'un four) . . . 7
3.1 Exercice (Interprétation de phrases logiques) . . . 15
3.2 Exercice (Convertir à partir d'une équation) . . . 15
3.3 Exercice (Convertir à partir d'un schéma bloc) . . . 15
3.4 Exercice (Convertir à partir d'un logigramme) . . . 16
3.5 Exercice (Forme canonique ou pas?) . . . 18
3.6 Exercice (Conversion à partir de tables de vérité) . . . 18
4.1 Exercice (Demonstration par l'algèbre de Boole) . . . 20
4.2 Exercice (Simplification par l'Algèbre de Boole) . . . 23
6.1 Exercice (Combinaisons adjacentes) . . . 24
6.2 Exercice (Remplissage de tableaux de Karnaugh) . . . 28
6.3 Exercice (Remplissage de tableaux de Karnaugh à partir d'équation) . . . 29
6.4 Exercice (Valeur d'un implicant) . . . 30
6.5 Exercice (Implicants premiers) . . . 31
6.6 Exercice (implicants premiers essentiels) . . . 33
6.7 Exercice (Simplification à l'aide de Karnaugh) . . . 35
7.1 Exercice (Problème: Chauffage d'une cuve) . . . 38
7.2 Exercice (Problèmes non résolus) . . . 39

1 Différents types de signaux

1.1 Les variables analogiques

Il s'agit de variables **continues** (position, hauteur, température). Ces variables peuvent prendre toutes les valeurs possibles : entre deux valeurs, il existe toujours une autre valeur. Problème : ces variables sont impossibles à stocker. Par exemple, is est impossible de stocker du nombre π étant donné l'infinité du nombre de décimales.

1.2 Les variables numériques

Les variables numériques ne peuvent prendre qu'un nombre discret de valeurs : elles sont basées sur des nombres, qui sont des combinaisons de chiffres (au moins dans la base décimale que nous utilisons tous les jours). Elles sont une approximation des variables analogiques $\pi \approx 3.1415$. Ce sacrifice de précision permet en revanche de stocker de l'information. Il est par exemple possible de stocker tous les nombres entre 0 et 1, avec une précision de 1/1000 (il y en a 1001 : 0 0.001, 0.002,........0.999, 1).
L'ensemble de la technologie humaine repose sur des systèmes numériques. Le terme de "numérique" s'est largement transformé dans le langage commun et lorsqu'on parle de "numérique", on pense beaucoup plus aux technologies numériques binaires (on y reviendra).

1.3 Les variables logiques ou binaires

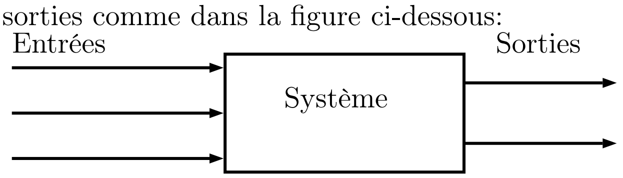
Une variable logique, ou binaire est une variable numérique ne pouvant prendre que deux valeurs : 0 ou 1. Elle représente une variable qui répond à une question logique qui implique la réponse : Vrai ou Faux, Actif ou Non-actif, Présent ou Absent.

Les variables binaires ont pris énormément d'importance avec le pilotage de systèmes électroniques reposant **sur des capteurs et actionneurs logiques** (Porte fermée ou ouverte? Ticket valide ou invalide? Voyant allumé ou éteint? interrupteur enclenché ou non?) **et bien entendu, avec la naissance de l'informatique** où toute l'information était stockée magnétiquement (magnétisé ou non-magnétisé) et où le processeur repose sur des transistors en mode saturé (laisse passer le courant, ou coupe le courant).

Les variables binaires se combinent pour former des variables numériques binaires (de la même façon que des chiffres se combinent en nombres). Il y a d'ailleurs une équivalence entre tous les systèmes de numération: ce qui peut être fait en décimal peut également être fait en binaire.

1.4 Les systèmes

Un système, est un mécanisme qui, en fonction de variables d'entrée, modifie des variables de sortie. En général, un système est représenté par un système avec ,à gauche, les entrées et à droite, les sorties comme dans la figure ci-dessous:



Exercice 1.1 (Système analogiques) *:*

- *Le thermomètre à mercure: Quelles sont ses entrées et ses sorties? (Entrée Température, sortie: hauteur de mercure)*
- *Corps humain: Quelles sont ses entrées et ses sorties? (Transforme de la nourriture en énergie mécanique.)*

■

Un système logique, c'est simplement un système pour lequel, **les entrées et les sorties sont des variables logiques.**

Exercice 1.2 (Système logique) *:*

- *Un neurone: Quelles sont ses entrées et ses sorties? (En fonction des impulsions électriques des neurones connectés (0 ou 1), produit un signal électrique (0 ou 1))*
- *Le transistor en régime saturé: Si un courant est présent dans la base, alors le courant est autorisé à passer entre le collecteur et l'émetteur*

■

On peut souligner le fait que la structuration d'un ensemble de systèmes logiques simples peut déboucher sur un fonctionnement très perfectionné (neurones → cerveau, transistor→ processeur). Il existe deux grandes familles de systèmes :

- Les Systèmes combinatoires : la valeur de la sortie ne dépend que des valeurs en entrée uniquement.
- Les Systèmes séquentiels : la valeur de sortie dépend des entrées mais également de variables internes au système (mémoire). Ces systèmes ne seront pas étudiés dans ce cours.

Exercice 1.3 (combinatoire ou séquentiel?) *:*

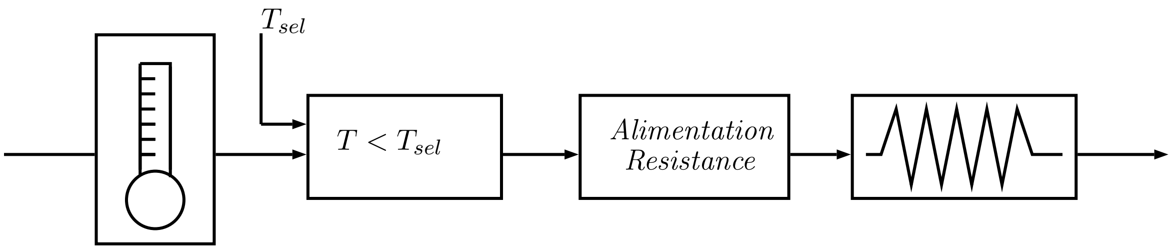
- *Un saturateur électronique.(combinatoire)*
- *Ampoule avec interrupteur simple. (combinatoire)*
- *Ampoule avec relai. (séquentiel)*
- *un Ascenseur (séquentiel, car l'action prise dépend de la position de l'ascenseur (monter ou descendre))*
- *Une voiture, qui prend en entrée une impulsion électrique et la fait avancer d'un mètre et renvoie en sortie la position. Séquentiel: Ḷa position finale dépend aussi de la position initiale … Une même entrée peut donner plusieurs sorties.*

■

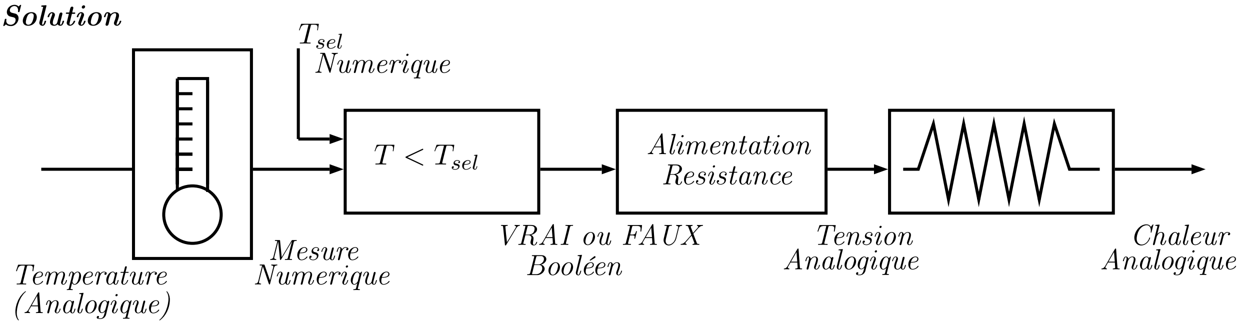
Tout système séquentiel devient combinatoire dès lors qu'on considère les états internes comme des entrées du système. Pour l'exemple de la voiture, si la position initiale (qui est un état du système) est supposée connue et considérée comme une entrée, ce système devient un système combinatoire. Dans ce cours nous nous intéresserons principalement aux systèmes combinatoires.

Exercice 1.4 (Exemple d'un four) *:*

Donnez les noms et types de variables pour ce système combinatoire qui est un thermostat de four.



Solution



■

La discipline qui traite théoriquement de l'étude et de la commande des systèmes séquentiels (appelés systèmes dynamiques lorsque les variables sont analogiques) s'appelle **l'automatique (control theory en anglais)**. La spécialisation de l'automatique aux systèmes logiques s'appelle **"l'automatique à évènements discrets"** aussi appelée **automatisme**.

1.5 But du cours

En conclusion, ce cours s'intéresse à plusieurs aspects des systèmes numériques:

1) L'étude des variables et systèmes logiques combinatoires : Ce chapitre a pour but de formaliser mathématiquement **la Logique** en utilisant des variables binaires. C'est un peu la partie SP du cours.
2) L'étude des variables numériques dans les bases binaires : Si une grosse partie de notre science repose sur des variables numériques décimales, toute l'informatique repose sur des variables numériques binaires. Il devient important alors de comprendre comment . **écrire, compter et calculer des nombres utilisant les variables logiques.** C'est un peu la partie mathématicienne du cours avec certaines généralités sur les changements de base, principalement dédiées aux bases binaires.
3) Le stockage usuel des variables numériques : Une fois le calcul en base binaire assimilé, il est important de comprendre comment stocker des nombres dans un système informatique et les implications sur les différents calculs machine. C'est un peu la partie informaticienne du cours.

A l'heure actuelle, il est impossible à quiconque de travailler dans des domaines comme les télécommunications, l'informatique ou toute technologie numérique sans comprendre ces concepts.

2 Les représentations utilisées en logique combinatoire

Même si les travaux mathématiques majeurs relatifs à la logique se situent au 19ème siècle, leur utilisation se situe près d'un siècle plus tard avec l'apparition des nouvelles technologies : principalement l'automatisme, l'électronique, l'électrotechnique et l'informatique. Chacune a proposé une écriture pour les variables et équations logique. Voyons ici l'ensemble des représentations usuelles et habituelles pour tout système logique.
Il existe, pour toute opération ou système logique, 4 représentations. Elles sont toutes **équivalentes** et **à connaître**:

- Equations
- Table de vérité
- Schéma-Bloc
- Logigrammes

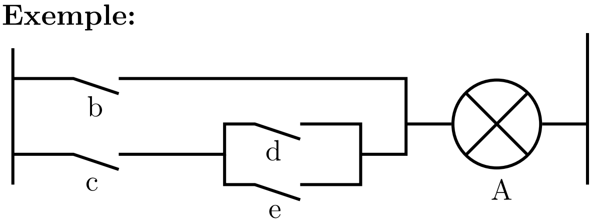
Puisqu'on parle de systèmes logiques, on peut considérer que certaines variables sont des **variables d'entrée** du système, qui se combinent pour définir le résultat des **variables de sortie**. Par nomenclature, les sorties sont souvent indiquées par **des lettres majuscules**, et les entrées par des **minuscules**.

2.1 La représentation Electrotechnicienne : Le schéma contact

Le schéma contact (appelé également LADDER) est issu de l'électrotechnique et de l'automatisme. Dans la représentation d'un système logique en schéma bloc:

- Les entrées sont représentées par des interrupteurs (ouverts "0" ou fermés "1")
- Les sorties sont représentées par des lampes (éteintes "0" ou allumées "1")
- A Gauche se situe une barre de potentiel (5V) et à droite la masse (0V)
- Les opérations sont représentées par des branches en parallèle ou en série

Exemple:

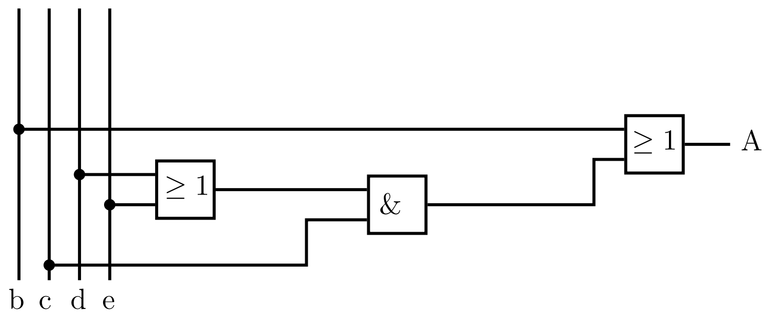


Ici, les entrées sont b, c, d, e et la sortie est A. Pour comprendre le fonctionnement il faut alors se poser la question suivante: **"Quels interrupteurs doivent être enclanchés pour que le courant puisse passer et allumer la lampe de sortie?"**.

2.2 La représentation Electronicienne : Le logigramme

Dans la vision electronicienne des systèmes logique:

- Une variable est représentée par un potentiel sur un fil (5V pour "1" et 0V pour "0").
- Une opération est représentée par un bloc logique simple.



La question à se poser est : **"quel est le potentiel sur le fil de sortie en fonction des potentiels d'entrée?"** Cette vision est inspirée par le fait qu'actuellement, beaucoup de ces blocs sont manufacturés en circuits intégrés réels. Ainsi, le logigramme correspond peu ou prou à un câblage électronique réel sur circuit imprimé.
Enfin, on peut noter que les symboles sont différents qu'il s'agisse de la norme Européenne ou de la norme Américaine. Dans ce cours, nous ne verrons que la norme européenne. Nous vous renvoyons vers la page *wikipédia* pour les normes américaines: *https://fr.wikipedia.org/wiki/Fonction_logique.*

2.3 La représentation mathématique: équations et table de vérité

La représentation mathématique utilise **des équations** et **des tables de vérité**.

2.3.1 Les équations logiques

Les équations logiques utilisent des symboles pour les opérations et des variables. Par exemple :

$$A = b + c.(d + e) \tag{1}$$

est une équation logique, en considérant que A est la sortie et b, c, d, e sont des entrées.

2.3.2 Les tables de vérité

La table de vérité quant à elle représente une table définissant tous les résultats d'une opération. Vous connaissez la table de multiplication , la table d'addition,... D'ailleurs, dans $\mathbb{N}$, on pourrait penser à faire des tables pour beaucoup d'autres opérations.

Exemple:

$C = \max(a, b)$ pour tous les chiffres entre 0 et 9 ($\{0, ...9\} \times \{0, ...9\} \rightarrow \{0, ...9\}$)

a	b	C
0	0	0
0	1	1
$\vdots$	$\vdots$	$\vdots$
0	9	9
1	0	1
$\vdots$	$\vdots$	$\vdots$
1	9	9
2	0	2
$\vdots$	$\vdots$	$\vdots$
9	9	9

Les points importants:

- Les "entrées" sont à gauche, la ou les "sorties" à droite.
- Cette table est limitée à un nombre de lignes. Pourtant elle définit toutes les combinaisons possibles quelque soit l'élément choisi dans l'ensemble de départ (ici {0,...9}). Elle contient 100 lignes: 10 (pour le choix de a) $\times$ 10 (pour le choix de b.
- Pour lister toutes les possibilités, le chiffre b change à toutes les lignes. Le chiffre a change toutes les 10 lignes. **Ceci vient de la base décimale (10 Chiffres).**
- **Si on faisait la table de $D = max(a, b, c)$, elle contiendrait 1000 lignes, c changerait toutes les lignes, b toutes les 10 lignes, et a toutes les 100 lignes.**

■

Et bien, lorsqu'on parle de "table de vérité" on parle de table d'une opération où toutes les entrées et sorties sont des variables logiques: ils valent tous 0 ou 1. Puisque la sortie est "vraie" ou "fausse" on l'appelle table de "vérité". Pour reprendre l'exemple précédent la table de l'opération $C = \max(a, b)$ pour tous les chiffres entre 0 et 1 ($\{0,1\} \times \{0,1\} \to \{0,1\}$) ressemblerait à cela

a	b	C
0	0	0
0	1	1
1	0	1
1	1	1

Cette table n'a que 4 lignes : 2(pour le choix de a) $\times$ 2 (pour le choix de b). Le chiffre b change à toutes les lignes, et le chiffre a change toutes les 2 lignes (et non plus 10 comme en décimal). On peut étendre ce résultat.

Soit une opération avec N entrées, alors:

Les variables d'entrées se situent à gauche, et la sortie à droite.

La table de vérité contiendra 2^N lignes :
2(pour le choix de la variable 1) $\times$ 2(pour le choix de la variable 2) $\times \ldots \times$ 2(pour le choix de la variable N).

La variable 1 change toutes les lignes, la variable 2 change toutes les 2 lignes, la variable 3 toutes les 4 lignes, ..., **la variable N toutes les 2^{N-1} lignes**.

Exemple: Table de vérité à 4 variables

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

- Cette table contient $2^4 = 16$ lignes.
- d change toutes les lignes, c change toutes les 2 lignes, b change toutes les 4 lignes, et a toutes les 8 lignes.

3 L'algèbre de Boole

3.1 Un peu d'histoire

L'algèbre (de l'arabe al-jabr) de Boole (Georges Boole 1815-1864), est une des premières algèbres modernes.

De manière générale, l'algèbre permet d'exprimer les propriétés des opérations et le traitement des équations et aboutit à l'étude des structures algébriques. L'algèbre moderne a largement permis d'unifier de nombreuses théories (étude de fonctions, polynômes, espace vectoriels,...) grâce à l'introduction d'outils comme les matrices. Par exemple, l'algèbre linéaire permet la formalisation des systèmes d'équations linéaires.

L'agèbre de Boole définit tous les outils mathématiques pour une approche algébrique de la **logique**. Comprenez que la logique ne se limite pas aux systèmes numériques. Par exemple:

- "Un animal est un oiseau" **SI ET SEULEMENT SI** l'animal "a des ailes" **ET** "il pond des oeufs"

est une définition logique. Elle a beaucoup d'implication en elle et notamment:

- Si un animal ne pond pas d'oeuf ce n'est pas un oiseau
- Si il n'a pas d'ailes non plus
- Si je parle d'un oiseau je sais qu'il aura des ailes ET qu'il pondra des oeufs
- Ca me permet de savoir qu'une licorne n'est pas un oiseau et qu'un crocodile n'est pas un oiseau.
- Ca me permet de définir que : "Un animal n'est pas un oiseau" **SI ET SEULEMENT SI** l'animal "n'a pas d'ailes" **OU** "il ne pond pas d'oeufs"

L'algèbre de Boole, permet de formaliser mathématiquement tous ces concepts de logique. Rien que ça. Elle trouvera son application pratique près d'un siècle plus tard dans la téléphonie, l'informatique ou l'électronique.

3.2 Les éléments essentiels de l'algèbre de Boole.

Comme toute algèbre, il s'agit de définir **un ensemble** et **des opérations**.

3.2.1 L'ensemble de Boole.

Vous connaissez déjà beaucoup d'ensembles:

- $\mathbb{R}$, $\mathbb{C}$: ensemble analogique
- $\mathbb{N}, \mathbb{Z}, \mathbb{Q}$: ensembles numériques

L'ensemble définit par Boole ne contient que deux valeurs : 0 et 1. il s'écrit donc $B = \{0, 1\}$. C'est un des ensembles les plus simples. Tout élément de B est soit 0, soit 1. Ils représentent les valeurs "FAUX" et "VRAI" respectivement. On appelle ces valeurs des valeurs Booléennes ou plus communément, des Booléens.

3.2.2 Les opérateurs de Boole.

Vous connaissez beaucoup d'opérateurs mathématiques : L'addition, la multiplication, la comparaison, le maximum, la division, ... Dans l'algèbre de Boole, il existe 4, et seulement 4 opérations:

- l'égalité : Elle représenté l'équivalence **SI ET SEULEMENT SI** du langage commun.
- l'union : Elle représente l'opération **OU** du langage commun.
- L'intersection : Elle représente l'opération **ET** du langage commun.
- La négation ou contraire ou complémentarité: Elle représente l'opération **NON** ou **PAS** du langage commun.

Ces opérations constituent les opérations exactement nécessaires et suffisantes pour réaliser TOUT système logique.

3.3 Description des différentes opérations dans toutes les représentations logiques

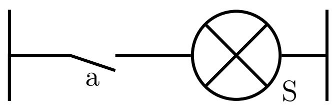
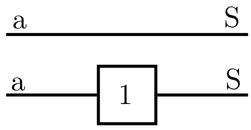
3.3.1 L'égalité

L'égalité correspond à l'équivalence en algèbre de Boole (comme toutes les égalités que vous connaissez d'ailleurs). Elle correspond en langage commun, à la condition "SI ET SEULEMENT SI".

- Soit la proposition S *"Je fais du vélo"*
- Soit la proposition a *"Il fait beau"*

Alors la phrase *"Je fais du vélo"* SI ET SEULEMENT SI *"il fait beau"* se note :

- En equation Boole $S = a$.
- La table de vérité est triviale. S recopie simplement a.
- En Schéma Bloc, Cela correspond à un simple interrupteur.
- En Logigramme, cela correspond à un bloc carré rempli avec un 1. Cette notation n'est pas vraiment utilisée, et on préfèrera la représentation par un simple fil de potentiel.

Equation	Table de Vérité	Schéma contact	Logigramme
$S = a$	a | S 0 | 0 1 | 1		

ATTENTION:

- Le "SI" indique une condition suffisante : *"Je fais du vélo SI il fait beau"*, indique que je fais effectivement du vélo à chaque fois qu'il fait beau mais peut être pas seulement. Je peux faire aussi du vélo quand il pleut. Elle est souvent notée : *"Je fais du vélo"* $\Leftarrow$ *"il fait beau"*
- Le "SEULEMENT SI" indique une condition nécessaire : *"Je fais du vélo SEULEMENT SI il fait beau"* indique que il faut forcement qu'il fasse beau pour que je fasse du vélo. Mais il y a peut être de jours ou il fait beau et ou je ne fais pas de vélo. Elle est souvent notée : *"Je fais du vélo"* $\Rightarrow$ *"il fait beau"*
- Le "SI ET SEULEMENT SI " indique l'équivalence. *"Je fais du vélo SI ET SEULEMENT SI il fait beau"* veut dire que chaque jour où il fait beau, et seulement dans ces moments la, je prend mon vélo. Il n'y a aucune situation où il fait beau et où je ne suis pas sur mon vélo, et aucune situation où je suis sur mon vélo et où il ne fait pas beau. C'est également souvent noté *"Je fais du vélo"* $\Leftrightarrow$ *"il fait beau"*

3.3.2 L'Union

En algèbre de Boole l'Union correspond au besoin d'avoir **au moins** une condition logique vraie parmi plusieurs. Elle correspond en langage commun, à la condition "OU".

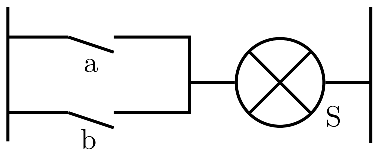
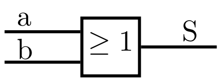
- Soit la proposition S *"Je fais du vélo"*
- Soit la proposition a *"Il fait beau"*
- Soit la proposition b *"C'est dimanche"*

Alors la phrase *"Je fais du vélo"* SI ET SEULEMENT SI *"il fait beau"* **OU** que *"c'est dimanche"* se note :

- En équation Boole $S = a+b$. ATTENTION, le signe $+$ correspond ici à l'Union, pas à l'addition telle que vous le connaissez: $1 + 1 \neq 2$. 2 n'est pas dans l'ensemble de Boole! $1 + 1 = 1$.
- La table de vérité traduit le fait que si **au moins** une des conditions a ou b est vraie, alors la sortie est vraie, sinon elle est fausse: $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$, $1 + 1 = 1$.
- En Schéma Bloc, le courant doit passer lorsque au moins un des interrupteurs est fermé. Ceci est l'équivalent d'un embranchement en parallèle.
- En Logigramme, cela correspond à un bloc carré rempli avec un ≥ 1. Ceci indique " au moins 1 entrée doit avoir un potentiel égal à 1 pour activer la sortie"

<table>
<tr><th>Equation</th><th>Table de Vérité</th><th>Schéma contact</th><th>Logigramme</th></tr>
<tr><td>$S = a + b$</td><td>

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

</td><td>
</td><td>
</td></tr>
</table>

Il y a nouveau équivalence ici. Dès qu'il fait beau, je suis sur mon vélo, et tous les dimanches, je suis sur mon vélo. Si vous me voyez sur mon vélo c'est forcément qu'il fait soit beau (et qu'on est pas dimanche) soit qu'on est dimanche (et il pleut) soit qu'on est dimanche et qu'il fait beau.

3.3.3 L'Intersection

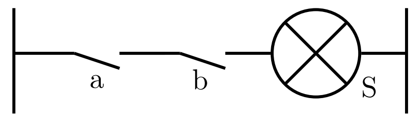
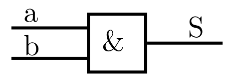
En algèbre de Boole, l'Intersection correspond au besoin d'avoir toutes les conditions logiques nécessaires vraies et ceci **simultanément**. Elle correspond en langage commun, à la condition "ET".

- Soit la proposition S *"Je fais du vélo"*
- Soit la proposition a *"Il fait beau"*
- Soit la proposition b *"C'est dimanche"*

Alors la phrase *"Je fais du vélo"* SI ET SEULEMENT SI *"il fait beau"* **ET** que *"c'est dimanche"* se note :

- En équation Boole $S = a.b$. ATTENTION, le signe . correspond ici à l'Intersection, pas à la multiplication telle que vous le connaissez, même si elle se comporte dans l'ensemble de Boole comme la multiplication.
- La table de vérité traduit le fait que si les deux conditions a et b sont vraies simultanément, alors la sortie est vraie, sinon elle est fausse: $0.0 = 0$, $0.1 = 1$, $1.0 = 1$, $1.1 = 1$.

- En Schéma Bloc, le courant doit passer lorsque les deux interrupteurs sont fermés. Ceci est l'équivalent d'un embranchement en série.
- En Logigramme, cela correspond à un bloc carré rempli avec un &. Ceci indique " les deux entrées doivent avoir un potentiel égal à 1 pour activer la sortie"

Equation	Table de Vérité	Schéma contact	Logigramme
$S = a.b$	a | b || S 0 | 0 || 0 0 | 1 || 0 1 | 0 || 0 1 | 1 || 1		

Il y a toujours équivalence ici. Dès que c'est un dimanche ensoleillé, je suis sur mon vélo. Si vous me voyez sur mon vélo c'est forcément qu'il fait beau et qu'on est dimanche.

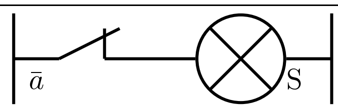
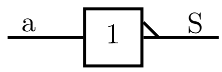
3.3.4 La négation ou complémentarité

En algèbre de Boole, la négation correspond au contraire d'une condition. Elle correspond en langage commun, à la condition "PAS".

- Soit la proposition S *"Je fais du vélo"*
- Soit la proposition a *"Il pleut"*

Alors la phrase *"Je fais du vélo"* SI ET SEULEMENT SI *"il ne pleut"* **PAS** se note :

- En équation Boole $S = \bar{a}$.
- La table de vérité traduit le fait que la sortie est toujours l'opposé de l'entrée.
- En Schéma Bloc, le courant passe lorsqu'on active PAS l'interrupteur. Ainsi, lorsqu'on laisse l'interrupteur au repos, il doit laisser passer le courant. On appelle cela un contact normalement fermé. Si on y touche pas, il est fermé. On le note avec un interrupteur spécial qui s'oriente vers le Haut (contrairement à d'habitude) avec l'ajout d'un petit contact.
- En Logigramme, cela correspond à un bloc carré, rempli avec un 1 suivi d'une petite barre qui indique l'inversion.

Equation	Table de Vérité	Schéma contact	Logigramme
$S = \bar{a}$	a || S 0 || 1 1 || 0		

La notion de contraire est une des plus difficile a appréhender en logique par l'esprit humain. Pensez aux phrases: "Ne voudrais tu pas, ne pas me passer le sel?" De la même façon, quel est le contraire de *"il fait beau OU c'est dimanche"*? L'algèbre de Boole donne une très bonne compréhension des doubles négations ou des négations de combinaisons logiques.

3.3.5 Commentaires

D'autres opérations booléennes existent qui disposent de notations nomenclaturées dans différentes représentations (ET NON, OU NON, OU Exclusif, ET inclusif,...). Cependant, elles peuvent toutes s'exprimer en fonction des opérations de bases, et même si il est bon de les connaître, elles ne sont pas nécessaires.

3.4 Exercices de conversions entre représentations

Ces exercices servent à comprendre une équation logique et maîtriser le passage d'une représentation à l'autre.

Exercice 3.1 (Interprétation de phrases logiques) *:*

Traduisez les phrases suivantes en équation logique, en logigramme et en schéma bloc:

- *Equation logique, en logigramme et en schéma bloc: "Je fais du vélo" SI ET SEULEMENT SI ("il fait BEAU" ET "il ne fait pas chaud ") OU ("c'est dimanche" ET "il fait chaud") ($S = (a.\bar{b}) + (c.b)$)*
- *Equation logique, en logigramme : "Je fais du vélo" SI ET SEULEMENT SI je n'ai PAS la condition "il fait beau ET il fait chaud".*

■

Exercice 3.2 (Convertir à partir d'une équation) *:*

Pour chaque valeur de a,b,c et d donnez la valeur de S sachant que :

- $S_1 = a.b.c + d$
- $S_2 = a.b.(c + d)$
- $S_3 = (a + \bar{b})(c + d)$
- $S_4 = \overline{\overline{b + a}.c.d}$

Pour S_1 S_2 S_3 faire la conversion en schéma Bloc et en Logigramme.
Pourquoi est il difficile de faire S_4 en schéma bloc? (Parce que il y a la négation d'une combinaison logique)

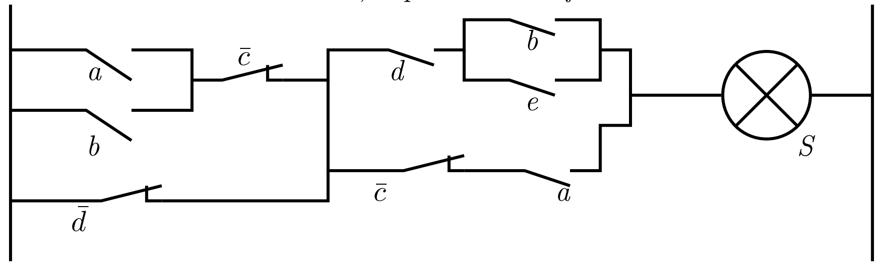
■

Ceci correspond à faire la table de vérité à partir d'une équation. C'est lister toutes les valeurs de S en fonction des entrées. Il est possible de trouver quelques "trucs et astuces" pour aller plus vite.

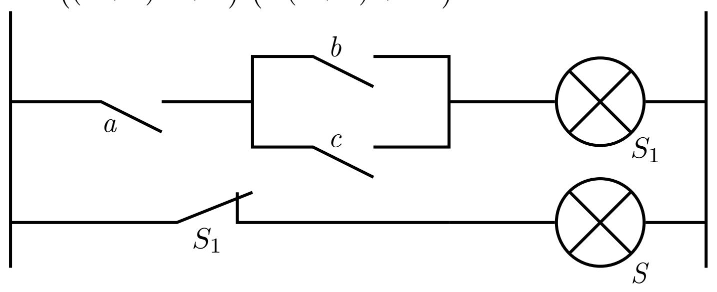
- Dans S_1 dès que a, b, ou c vaut 0, $a.b.c = 0$. Pour toutes les lignes ou $d = 1$ alors S_1 est forcément égal à 1
- Pour S_2 dès que $a = 0$ ou $b = 0$ alors S_2 vaut forcément 0.
- ...

Exercice 3.3 (Convertir à partir d'un schéma bloc) *:*

Soit les schéma blocs suivant, exprimez S en fonction des entrées en équation, puis en Logigramme.



$S = \big((a+b).\bar{c} + \bar{d}\big).\big(d.(b+e) + \bar{c}.a\big)$

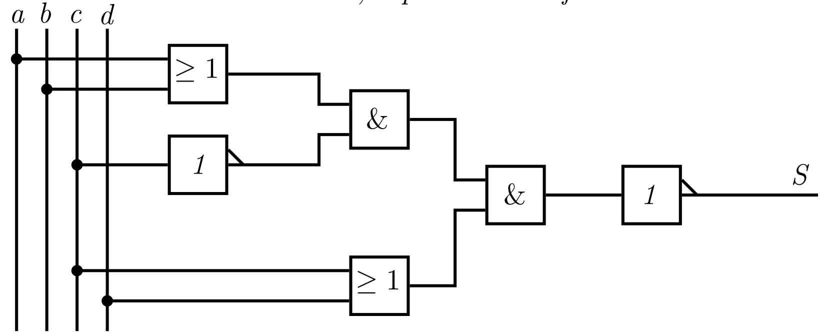


$S = \overline{a.(b+c)}$

■

Exercice 3.4 (Convertir à partir d'un logigramme) *:*

Soit les schéma blocs suivant, exprimez S en fonction des entrées en équation, puis en Schéma Bloc.



$S = \overline{\big((a+b).\bar{c}\big).(c+d)}$

■

3.5 Transformer une table de vérité en équation: Equation canonique disjonctive

A ce point du cours, vous savez normalement partir d'une équation, d'une schéma bloc ou d'un logigramme afin de transformer un écriture en une autre. D'ailleurs, ce passage se fait souvent par le passage intermédiaire en équation logique. Reste donc seulement à savoir partir d'une table de vérité pour la transformer en équation et ainsi pouvoir la transcrire en une autre écriture.
La technique consiste à :

1) Repérer quelles lignes mettent la sortie à "1".
2) Interpréter chacune de ces lignes en équation. Ceci correspond à la définition de **mintermes**.
3) Puisque la table de vérité liste TOUTES les combinaisons possibles d'entrées, alors $S = 1$ SI ET SEULEMENT SI AU MOINS l'une des **conditions de ligne** est remplie. Ainsi pour avoir l'équation de S il suffit de faire **l'union** de tous les **mintermes** mis en équation dans l'étape 2). Il y a ici une réelle égalité. L'équation obtenue par cette méthode s'appelle **l'équation canonique disjonctive** d'un système logique.

Définition:
L'équation canonique disjonctive d'un système logique est une équation composée **d'une Union de mintermes. Un minterme est une intersection contenant TOUTES les variables d'entrées d'un système logique.**

Exemple:

Soit la table de vérité ci-dessous:

1) la sortie s'active aux lignes 2 4 6 7 8 11 12 15
2) Nous allons faire en détails l'exemple de la deuxième ligne () et il faudra faire le même raisonnement pour toutes les lignes de l'étape 1). On peut vérifier alors que **pour la deuxième ligne** $S = 1$ SI $a = 0$ ET $b = 0$ ET $c = 0$ ET $d = 1$. Ceci correspond **pour la deuxième ligne** à $S = 1$ SI $\bar{a}.\bar{b}.\bar{c}.d = 1$. Notez bien ici, le SI qui ne correspond pas à une égalité (SI et seulement SI). Le **minterme** associé est donc $\bar{a}.\bar{b}.\bar{c}.d$. Finalement on met, une fait la NEGATION sur toutes les entrées qui valent 0 et on fait l'intersection avec toutes les entrées.
3) On fait l'Union des **mintermes** et ceci nous donne :$S=\ \bar{a}.\bar{b}.\bar{c}.d\ +\bar{a}.\bar{b}.c.d+\bar{a}.b.\bar{c}.d\ +\bar{a}.b.c.\bar{d}\ +\bar{a}.b.c.d+\ a.\bar{b}.c.\bar{d}+\ a.\bar{b}.c.d+a.b.c.\bar{d}$

a	b	c	d	S	(Etape 1)	minterme associé (Etape 2)
0	0	0	0	0		
0	0	0	1	1	$\leftarrow$	$\bar{a}.\bar{b}.\bar{c}.d$
0	0	1	0	0		
0	0	1	1	1	$\leftarrow$	$\bar{a}.\bar{b}.c.d$
0	1	0	0	0		
0	1	0	1	1	$\leftarrow$	$\bar{a}.b.\bar{c}.d$
0	1	1	0	1	$\leftarrow$	$\bar{a}.b.c.\bar{d}$
0	1	1	1	1	$\leftarrow$	$\bar{a}.b.c.d$
1	0	0	0	0		
1	0	0	1	0		
1	0	1	0	1	$\leftarrow$	$a.\bar{b}.c.\bar{d}$
1	0	1	1	1	$\leftarrow$	$a.\bar{b}.c.d$
1	1	0	0	0		
1	1	0	1	0		
1	1	1	0	1	$\leftarrow$	$a.b.c.\bar{d}$
1	1	1	1	0		

Equation canonique disjonctive obtenue:

$$S = \bar{a}.\bar{b}.\bar{c}.d + \bar{a}.\bar{b}.c.d + \bar{a}.b.\bar{c}.d + \bar{a}.b.c.\bar{d} + \bar{a}.b.c.d + a.\bar{b}.c.\bar{d} + a.\bar{b}.c.d + a.b.c.\bar{d}$$

■

Remarque:

L'équation canonique disjonctive est souvent fastidieuse à lire et est ne correspond pas, en général, à l'équation la plus simple qu'on peut obtenir. Après quelques exercices, nous verrons comment l'Algèbre de Boole permet de manipuler les équations logiques afin de les simplifier.
Cependant, dans de nombreuses analyses de problèmes logique, la table de vérité sera la première chose qui sera obtenue. Ainsi il est crucial de savoir interpréter et transformer une table de vérité en équation.

3.5.1 Exercices

Exercice 3.5 (Forme canonique ou pas?) *:*

Les équations suivantes sont-elles canoniques ou non?

- $S = ab + b\bar{a} + ad$
- $S = abc + \bar{a}bc + ab\bar{c}$
- $S = a + a\bar{b}$

■

Exercice 3.6 (Conversion à partir de tables de vérité) *:*

Trouvez les équations canoniques des tables de vérité suivantes. Transformez-les en logigramme ou en schéma bloc.

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

$S = a.b.c$

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

$S = a.b.c + a\bar{b}.\bar{c} + \bar{a}.\bar{b}.c$

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

$S = \bar{a}.\bar{b}.\bar{c} + a.b.\bar{c}$

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

$S = \bar{a}.b + a.\bar{b} + a.b$
Canonique 1 du OU

■

4 Simplification par l'Algèbre de Boole

Nous allons voir ici certaines règles de l'algèbre de Boole qui permettront de simplifier des équations logiques. Par exemple, il est possible, d'émettre le postulat logique "Je fais du vélo" SI ET SEULEMENT SI " il fait beau" ET "il ne fait PAS beau". On voit que dans ce cas il est plus simple de dire: "Je ne fais jamais de vélo". On voit qu'il y a équivalence entre différentes équations logiques. En électronique, utiliser une équation plus simple signifie utiliser moins de composants et donc économiser.

4.1 Propriétés de l'Algèbre de Boole

Toutes ces propriétés sont à connaitre par coeur.

Commutativité

Comme pour l'addition et la multiplication entre réels, l'Union et l'Intersection sont commutatives en Algèbre de Boole:

- $a + b = b + a$
- $a.b = b.a$

Associativité

Comme pour l'addition et la multiplication entre réels, l'Union et l'Intersection sont associatives en Algèbre de Boole:

- $(a + b) + c = a + (b + c)$
- $(a.b).c = a.(b.c)$

Distributivité

Comme dans l'algèbre des réels, où la multiplication est distributive par rapport à l'addition, en algèbre de Boole , l'intersection est distributive par rapport à l'union:

- $a.(b + c) = (a.b) + (a.c)$

Cependant, il est important de noter que **CONTRAIREMENT** aux réels, où l'addition n'est pas distributive par rapport à la multiplication, en algèbre de Boole , l'union est distributive par rapport à l'intersection:

- $a + (b.c) = (a + b).(a + c)$

On remarquera qu'on a pas l'habitude de manipuler cette dernière loi et que souvent, lorsqu'on bloque dans une simplification d'équation, elle en est la cause.

Involution

L'involution indique simplement que le contraire du contraire de a est a lui-même.

- $\bar{\bar{a}} = a$

Notation

Un peu à la manière de l'algèbre avec des réels, on considèrera que si il n'y pas pas d'opérateur entre 2 variables, il s'agit d'une intersection:

- $a.b$ se note souvent ab

De plus on considèrera que dans le cas ou il n'y a pas de parenthèses, l'intersection est prioritaire sur l'union

- $ab + c$ signifie $(a.b) + c$. Il faudra donc ne pas oublier les parenthèses si l'on souhaite écrire $a(b + c)$.

4.2 Théorèmes de base

Enfin, afin d'opérer des simplifications d'équation, il conviendra de connaître ces théorèmes par coeur et de savoir les reconnaître en situation réelle.

Idempotence:

- $a + a = a$: "Il fait beau" OU "il fait beau" = "il fait beau"
- $a.a = a$: "Il fait beau" ET "il fait beau" = "il fait beau"

Elément neutre:

- $a.1 = a$: "Il fait beau" ET "toujours" = "il fait beau"
- $a + 0 = a$: "Il fait beau" OU "jamais" = "il fait beau"

Elément absorbant:

- $a.0 = 0$: "Il fait beau" ET "jamais" = "jamais"
- $a + 1 = 1$: "Il fait beau" OU "toujours" = "toujours"

Complémentarité:

- $a + \bar{a} = 1$: "Il fait beau" OU "Il ne fait PAS beau" = "toujours"
- $a.\bar{a} = 0$: "Il fait beau" ET "Il ne fait PAS beau" = "jamais"

4.3 Autres théorèmes utiles

Les lois suivantes peuvent se retrouver à partir de celles déjà évoquées mais il est fortement conseillé de les connaître pour se simplifier la vie

- $a + a.b = a$
- $a + \bar{a}b = a + b$
- $a.b + a.\bar{b} = a$
- $(a + b).(a + \bar{b}) = a$
- $a(a + b) = a$

Exercice 4.1 (Demonstration par l'algèbre de Boole) *:*

Démontrez les égalités précédentes. (Aide: distributivité de l'union, distributivité de l'union, factorisation, développement, développement)

■

4.4 Théorèmes de De Morgan

Les théorèmes de Augustus De Morgan (1806-1871) s'intéressent à la formalisation de négation d'équations logiques.

- Soit la proposition S *"Je fais du vélo"*
- Soit la proposition a *"Il fait beau"*
- Soit la proposition b *"C'est dimanche"*

Et le postulat logique "Je fais du vélo" SI ET SEULEMENT SI "il fait beau" OU " C'est dimanche" qui peut se formuler $S = a + b$. La question qu'on peut se poser est, "quand est-ce que je ne fais PAS de vélo" ? On cherche donc ici le contraire de S, qui est donc $\bar{S}$, et donc le contraire de $a + b$, qui est $\overline{a+b}$.
La réponse est: "Je ne fais pas de vélo" SI ET SEULEMENT SI "il ne fait PAS beau" ET " ce n'est PAS dimanche" qui se formule $\bar{S} = \bar{a}.\bar{b}$. Il s'agit du premier théorème de De Morgan:

- $\overline{a+b} = \bar{a}.\bar{b}$

De même quel est le contraire de "Je fais du vélo" SI ET SEULEMENT SI "il fait beau" ET " C'est dimanche" qui peut se formuler $S = a.b$? C'est "Je ne fais pas de vélo" SI ET SEULEMENT SI "il ne fait PAS beau" OU " ce n'est PAS dimanche" qui se formule $\bar{S} = \bar{a}.\bar{b}$. Il s'agit du deuxième théorème de De Morgan:

- $\overline{a.b} = \bar{a} + \bar{b}$

Pédagogiquement parlant on peut résumer en disant qu'on peut "couper" une barre de négation si on pense bien à changer tous les symboles d'opérations qui se trouvent en dessous.

ATTENTION: Le piège le plus habituel est d'oublier de considérer les parenthéses implicites. Par exemple, $\overline{a.b+c}$ N'EST PAS $\bar{a} + \bar{b}.\bar{c}$. En effet, ici on a oublié les parenthèses implicites $\overline{a.b+c}$=$\overline{(a.b)+c}$. On voit qu'en utilisant correctement De Morgan, le résultat est $(\bar{a}+\bar{b}).\bar{c}$ car les parenthèses restent!

4.4.1 Utilisation de De Morgan: L'équation Canonique conjonctive

Nous avons vu précédemment que l'on pouvait obtenir directement, à partir d'une table de vérité, une équation canonique disjonctive dans la section 3.5. Cependant, nous avons vu que pour un simple OU, son équation canonique disjonctive est $a + b = \bar{a}b + a\bar{b} + ab$. Cette équation canonique est loin d'être simple. Pourquoi privilégier l'opérateur de type ET dans une forme canonique? Parfois il vaut mieux considérer la forme canonique conjonctive.

Définition:
L'équation canonique conjonctive d'un système logique est une équation composée **d'une Intersection de maxtermes. Un maxterme est une union contenant TOUTES les variables d'entrées d'un système logique.**

Par exemple: $S = (a+b+c).(\bar{a}+\bar{b}+c).(a+\bar{b}+\bar{c})$ est une équation canonique d'un système logique à 3 variables. On voit que $S = a + b$ est une équation canonique conjonctive puisque $a + b$ est un maxterme. Comment obtenir une équation canonique conjonctive pour à partir d'une table de vérité?
C'est très simple:

1) Faire la table de vérité de $\bar{S}$

2) Trouvez l'équation canonique disjonctive pour $\bar{S}$

3) Utilisez De Morgan pour trouver la forme canonique conjonctive de S

Exemple:

Reprenons l'exemple du paragraphe 3.5. Obtenir l'équation de $\bar{S}$ signifie simplement inverser les 0 et les 1 au niveau de la sortie. Ensuite on applique la même stratégie que pour trouver la forme canonique disjonctive.

a	b	c	d	$\bar{S}$	(Etape 1)	minterme associé (Etape 2)
0	0	0	0	1	$\leftarrow$	$\bar{a}.\bar{b}.\bar{c}.\bar{d}$
0	0	0	1	0		
0	0	1	0	1	$\leftarrow$	$\bar{a}.\bar{b}.c.\bar{d}$
0	0	1	1	0		
0	1	0	0	1	$\leftarrow$	$\bar{a}.b.\bar{c}.\bar{d}$
0	1	0	1	0		
0	1	1	0	0		
0	1	1	1	0		
1	0	0	0	1	$\leftarrow$	$a.\bar{b}.\bar{c}.\bar{d}$
1	0	0	1	1	$\leftarrow$	$a.\bar{b}.c.\bar{d}$
1	0	1	0	0		
1	0	1	1	0		
1	1	0	0	1	$\leftarrow$	$a.b.\bar{c}.\bar{d}$
1	1	0	1	1	$\leftarrow$	$a.b.\bar{c}.d$
1	1	1	0	0		
1	1	1	1	1	$\leftarrow$	$a.b.c.d$

Et donc $\bar{S} = \bar{a}.\bar{b}.\bar{c}.\bar{d} + \bar{a}.\bar{b}.c.\bar{d} + \bar{a}.b.\bar{c}.\bar{d} + a.\bar{b}.\bar{c}.\bar{d} + a.\bar{b}.c.\bar{d} + a.b.\bar{c}.\bar{d} + a.b.\bar{c}.d + a.b.c.d.$

Puisque:
$S = \bar{\bar{S}} := \overline{\bar{a}.\bar{b}.\bar{c}.\bar{d} + \bar{a}.\bar{b}.c.\bar{d} + \bar{a}.b.\bar{c}.\bar{d} + a.\bar{b}.\bar{c}.\bar{d} + a.\bar{b}.c.\bar{d} + a.b.\bar{c}.\bar{d} + a.b.\bar{c}.d + a.b.c.d}$

On applique De Morgan et:

$S = (a{+}b{+}c{+}d).(a{+}b{+}\bar{c}{+}d).(a{+}\bar{b}{+}c{+}d).(\bar{a}{+}b{+}c{+}d).(\bar{a}{+}b{+}\bar{c}{+}d).(\bar{a}{+}\bar{b}{+}c{+}d).(\bar{a}{+}\bar{b}{+}c{+}\bar{d}).(\bar{a}{+}\bar{b}{+}\bar{c}{+}\bar{d})$

■

4.5 Exercices de simplification

La simplification ne peut être comprise qu'en pratiquant. Plus vous simplifierez, plus vous saurez simplifier. Il est difficile de déterminer une stratégie gagnante (pour l'instant) mais plusieurs étapes sont souvent nécessaires:

- **Bien connaître les théorèmes.**
- **L'application de De Morgan:** Lorsqu'une expression contient des négations d'expression (par exemple $S = \overline{\overline{a+b}.a} + ab$) il est conseillé d'utiliser De Morgan en premier.
- **Le passage par la forme canonique :** Il est toujours possible de passer par la forme canonique, en ajoutant des termes manquants pour en faire des mintermes: par exemple $S = ab + c$ n'est pas une équation canonique. En sachant que $(a + \bar{a}) = (b + \bar{b}) = (c + \bar{c}) = 1$ alors

$c = 1.1.c = (a+\bar{a})(b+\bar{b})c$ et $ab = ab.1 = ab(c+\bar{c})$. En conclusion $S = ab(c+\bar{c})+(a+\bar{a})(b+\bar{b})c$ et en développant, $S = abc + ab\bar{c} + abc + a\bar{b}c + \bar{a}bc + \bar{a}\bar{b}c$. On voit que le **minterme** abc apparait deux fois. Comme $a + a = a$, $abc + abc = abc$. Donc la forme canonique finale est $S = abc + ab\bar{c} + a\bar{b}c + \bar{a}bc + \bar{a}\bar{b}c$.

- **Le développement:** Parfois il n'est pas nécessaire d'utiliser la forme canonique. Seulement on va privilégier le développement avant tout. Par exemple, plutôt que de travailler avec une expression du type $S = a(b\bar{c} + ac) + abc + c(a + b)$, on va choisir de développer pour pouvoir faire les factorisations qui s'imposent. Ici $S = ab\bar{c}+a.ac+abc+ca+cb = ab\bar{c}+ac+abc+ca+cb$
- **La factorisation:** La factorisation est un élément essentiel. On essayera de toujours factoriser par le plus grand terme possible. Attention, une factorisation utile (qui est d'ailleurs dans les lois utile) est une factorisation du type $a + ab$. Ici il faut retenir que $a = a.1$ et donc $a + ab = a.1 + ab = a(1 + b) = a$. Dans l'exemple précédent, $S = ab\bar{c} + ac + abc + ca + cb$ On voit qu'on peut factoriser beaucoup de termes par ac ou ab:
  - factorisation par ac: $S = ab\bar{c} + ac(1 + 1 + b) + cb = ab\bar{c} + ac + cb$. Ic on peut choisir de factoriser b ou a. On peut faire soit l'une soit l'autre. $S = a(c+b\bar{c})+cb$ ou $S = b(a\bar{c}+c)+ac$. Cela mène à $S = a(c + b) + cb$ ou à $S = b(a + c) + ac$. Dans les deux cas c'est égal à $ab + bc + ac$.
  - Factorisation par ab: $S = ab(\bar{c} + c) + ac + cb = ab + ac + cb$. On voit ici que le même résultat est atteint mais en beaucoup moins d'étapes. Tous les chemins mènent à Rome mais l'oeil avisé reconnaitra souvent le plus court.
- Enfin, lorsqu'un équation est formée d'intersections d'union (par exemple $S = (a+b+c).(a+b)$), **il vaut parfois mieux travailler avec** $\bar{S}$. En effet, notre oeil est moins a l'aise avec les factorisations de l'union par rapport à l'intersection (on en a déjà parlé). Ainsi, l'algèbre de Boole dit directement que l'on peut factoriser $(a+b)$: $S = (a+b+c).(a+b) = (a+b+c).(a+b+0) = (a+b)+(c.0)$. C'est difficile à comprendre car ce genre de factorisation n'est pas vraie avec les réels. Dans de cas, on va préférer passer par $\bar{S} = \overline{(a + b + c).(a + b)} = \bar{a}\bar{b}\bar{c}+\bar{a}\bar{b}$ par De Morgan. Les factorisations et simplifications deviennent alors **évidentes**. $\bar{S} = \bar{a}\bar{b}(1+\bar{c}) = \bar{a}\bar{b}$. Une fois la simplification terminée, on peut réexprimer $S = \bar{\bar{S}} = \overline{\bar{a}\bar{b}} = a + b$ par De Morgan.

Exercice 4.2 (Simplification par l'Algèbre de Boole) *:*

- *Trouvez les formes canoniques disjonctives de* $S_1 = ab + \bar{c}$ *et* $S_2 = a\bar{b} + abc + a\bar{c}$.
- *Montrez que* $ab + \bar{b}c + a\bar{c} = a + \bar{b}c$. *(Passage par l'equation canonique puis factorisation par* $a\bar{b}$ *et* ab*)*
- *Simplifiez l'expression* $S = ab\bar{c}+b(a+\bar{c})+\overline{\bar{a} + b + \bar{a}c}$. *D'abord De Morgan, puis développement, puis factorisation. (Solution:* $S = a + \bar{c}b$
- *Simplifiez l'expression* $S = (\bar{a}b + a\bar{b}).(ab + \bar{a}\bar{b})$. *Développement. Solution* $S = 0$
- *Simplifiez l'expression* $S = \overline{\overline{\overline{a+b}.a}.b}$. *De Morgan. Attention ici, une barre fait office de parenthèses également. Solution:*$S = \bar{b}$
- *Montrez que* $\overline{(a + b)(\bar{a} + c)} = (a+\bar{b}).(\bar{a}+\bar{c})$. *Lorsqu'un démonstration s'avère trop compliquée on peut montrer qu'il y a égalité entre les équations canoniques. Sinon ici, on applique De Morgan à gauche, et on développe à droite. Il reste un terme en trop* $(\bar{b}\bar{c})$ *qui sera absorbé par les deux autres termes si on le développe en mintermes.*
- $S = (a + b + c).(a + \bar{b} + c).(a + \bar{b} + \bar{c})$. *Ici il est possible de tout développer, mais c'est long. On préferera travailler avec* $\bar{S} = \bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}bc$ *puis factoriser par* $\bar{a}b$ *puis par* $\bar{a}$ *menant à* $\bar{S} = \bar{a}.(b + \bar{c})$. *On en conclut que* $S = a + \bar{b}c$.

■

5 Conclusions Préliminaires

Jusqu'ici, nous avons vu en détails à quel point l'algèbre de Boole permet de formuler le concept de logique, d'un point de vue mathématique et permet de manipuler, en n'utilisant QUE des objets mathématiques, des expressions logiques. La puissance de cet outil réside dans le fait qu'il est applicable à **tout** type de logique, y compris un raisonnement commun. En ne connaissant que **3 opérations logiques**, il est possible de réaliser TOUT système logique, quelle que soit sa complexité et quelle que soit la technologie associée: Ordinateurs, automates, téléphones,... Tout repose sur l'algèbre de Boole.

Notamment, de manière ludique, on trouvera également une utilisation de l'Algèbre de Boole dans le jeu *Minecraft*, pour lequel il existe toute une technologie de circuits électriques ("redstone" dans le jeu). Il existe de nombreux sites expliquant comment faire des portes automatiques, des additionneurs,... et tout repose à nouveau sur l'algèbre de Boole et des trois opérations de base.
Dans le cadre des opérations logiques il nous reste deux notions à aborder:

- On a vu que la simplification peut être ardue parfois. Il existe une méthode graphique pour la simplification d'équations: Les tableaux de Karnaugh.
- Il nous reste à étudier des problèmes concrets qui utilisent la logique.

6 Simplification graphique d'équations logiques: Le tableau de Karnaugh

Nous avons vu que la simplification d'équations peut être complexe selon le nombre de variables en jeu. Cependant on s'aperçoit que la simplification passe souvent par une "bonne" factorisation:

- $abc + a\bar{b}c = ac(\bar{b} + b) = ac$
- $abc + a\bar{b}c + ab\bar{c} + a\bar{b}\bar{c} = ab + a\bar{b} = a$

Comment trouver LA ou LES BONNES factorisations? La clef réside dans l'écriture des mintermes. Ecrivons un minterme en représentant les variables par la valeur du booléen correspondant pour que le minterme soit VRAI. Par exemple, $a\bar{b}c$ s'écrit 101 ($a = 1, b = 0, c = 1$ implique $a\bar{b}c = 1$).

- Dans l'exemple $abc + a\bar{b}c$ les deux mintermes s'écrivent alors 111 et 101. On voit que un seul bit change entre les deux combinaisons. On les appelle des combinaisons **adjacentes**.

Définition:
Deux combinaisons Booléennes sont dîtes adjacentes, si elles ne diffèrent que d'un seul bit.

Exercice 6.1 (Combinaisons adjacentes) *:*

Quelles sont toutes les combinaisons adjacente à 1011?

■

- Dans l'exemple $abc + a\bar{b}c + ab\bar{c} + a\bar{b}\bar{c}$, les combinaisons sont 111,101,110,100 . On s'aperçoit qu'il est possible de passer d'une combinaison à l'autre en utilisant QUE des combinaisons adjacentes : 111,110,101,100. 111 est adjacente à 110 qui est adjacente à 101 qui est elle-même adjacente à 100.

Ainsi, on peut voir que la simplification dépend largement de l'équation canonique et de si elle contient on non des combinaisons adjacentes qui permettent de la simplifier.
Un outil permettant de visualiser graphiquement toutes les combinaisons adjacentes permettrait de simplifier les équations (Dans une table de vérité, deux lignes consécutives ne sont pas forcément des combinaisons adjacentes).
Il existe une manière de réordonner une table de vérité de façon que chaque ligne corresponde à une combinaison adjacente: **Le code Gray**.

6.1 Le code gray

Le code gray est utilisé dans certains éléments logiques comme la roue codeuse. Il existe une stratégie facile pour ordonner les combinaisons à N bits en code gray:
On commence par mettre les deux valeurs du premier bit.

a	b	c	d
			0
			1

Pour rajouter le deuxième bit, on fait la numérotation habituelle du deuxième bit, et on fait un miroir sur le premier bit:

a	b	c	d	
		0	0	
		0	1	miroir d
		1	1	
		1	0	

Pour rajouter le troisième bit, on fait la numérotation habituelle du troisième bit, et on fait un miroir sur les deux premiers bits:

a	b	c	d	
	0	0	0	
	0	0	1	miroir d
	0	1	1	
	0	1	0	miroir c, d
	1	1	0	
	1	1	1	
	1	0	1	
	1	0	0	

Et ainsi de suite. Voici ce que ca donne pour une table de vérité à 4 bits:

a	b	c	d	
0	0	0	0	
0	0	0	1	miroir d
0	0	1	1	
0	0	1	0	miroir c,d
0	1	1	0	
0	1	1	1	
0	1	0	1	
0	1	0	0	miroir b,c,d
1	1	0	0	
1	1	0	1	
1	1	1	1	
1	1	1	0	
1	0	1	0	
1	0	1	1	
1	0	0	1	
1	0	0	0	

On peut constater que toute ligne adjacente correspond à une combinaison adjacente. Cependant il reste des combinaison adjacentes qui ne sont pas sur des lignes adjacentes (par exemple 1011 et 0011). C'est pour cela que d'habitude on organise la table de vérité en en tableaux: **Les tableaux de Karnaugh!**

6.2 Les tableaux de Karnaugh

Les tableaux de Karnaugh (1953) permettent de simplifier graphiquement et simplement tout équation logique. Dans ce cours, nous ne nous intéresserons qu'à des systèmes logiques à 4 variables maximum. Il est possible d'utiliser Karnaugh pour un plus grand nombre de variables, mais cela devient plus fastidieux.

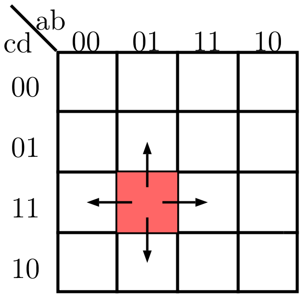
Définition:
Un tableau de Karnaugh est un tableau, où chaque case représente une ligne de la table de vérité. Dans un tableau de Karnaugh:

- Toutes les cases adjacentes à une case "C" (représentant un minterme "M") représentent un minterme "N" qui est adjacent à "M"
- Tous les mintermes "N" adjacents à un minterme "M" se situent dans les cases adjacentes à la case "C" représentant "M".

En d'autres termes, dans un tableau de Karnaugh, toutes les combinaisons adjacentes sont dans des cases adjacentes et ceci permet de trouver directement les bonnes factorisations.

6.2.1 Construction d'un tableau de Karnaugh

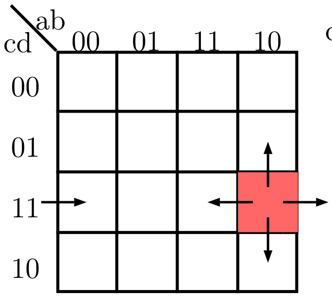
Afin de bien construire un tableau de Karnaugh avec les bonnes cases adjacentes, il faut faire attention à bien positionner les mintermes. C'est la qu'intervient le code gray. Il suffit, dans le cadre de ce cours de connaitre **le code Gray à 2 variables: 00 puis 01 puis 11 puis 10.** Voici des exemples de tableaux de Karnaugh à 4 variables.

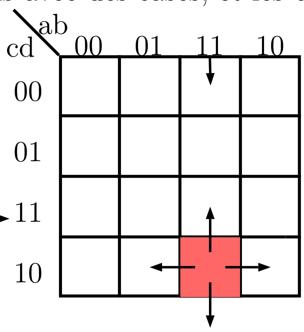


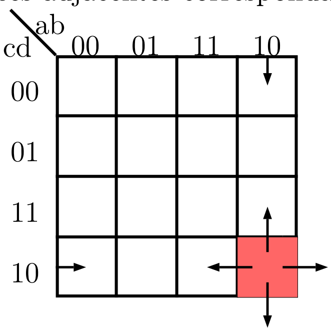
- Chaque colonne représente une combinaison de a et b
- Chaque ligne représente une combinaison de c et d
- La case rouge représente la case où $a = 0$, $b = 1$, $c = 1$ et $d = 1$, soit le minterme $\bar{a}bcd$
- On pourra remarquer que dès que des cases sont adjacentes, alors elles correspondent à des mintermes adjacents. Les cases adjacentes sont notées ici par des flèches.
- Ici peut importe qu'on mette a et b sur les colonnes et c et d sur les lignes. Les variables sont interchangeables.

Maintenant, on peut se demander pourquoi la case $\bar{a}\bar{b}\bar{c}\bar{d}$ (en haut à gauche) n'est pas adjacente à la case $a\bar{b}\bar{c}\bar{d}$ (en haut à droite), alors que ces combinaisons sont adjacentes. **Un tableau de Karnaugh est un tableau cyclique**. Lorsqu'on sort du tableau à droite, on y rentre à gauche, lorsqu'on en sort à gauche on y rentre à droite, lorsqu'on y sort par le haut on y rentre en bas, et lorsqu'on y sort par le bas on y rentre en haut. En fait, les cases $\bar{a}\bar{b}\bar{c}\bar{d}$ et $a\bar{b}\bar{c}\bar{d}$ SONT adjacentes.

Voici quelques illustrations avec des cases, et les cases adjacentes correspondantes:







6.2.2 Remplissage d'un tableau de Karnaugh à l'aide d'une table de Vérité

Une fois que l'on comprend que chaque case correspond à un minterme et que chaque minterme adjacent se trouve dans une case adjacente, il faut remarquer que chaque case d'un tableau de Karnaugh, correspond à une ligne d'un table de vérité.

a	b	c	d	S
0	0	0	0	I
0	0	0	1	II
0	0	1	0	III
0	0	1	1	IV
0	1	0	0	V
0	1	0	1	VI
0	1	1	0	VII
0	1	1	1	VIII
1	0	0	0	IX
1	0	0	1	X
1	0	1	0	XI
1	0	1	1	XII
1	1	0	0	XIII
1	1	0	1	XIV
1	1	1	0	XV
1	1	1	1	XVI

cd \ ab	00	01	11	10
00	I	V	XIII	IX
01	II	VI	XIV	X
11	IV	VIII	XVI	XII
10	III	VII	XV	XI

Dès lors pour remplir un tableau de Karnaugh à partir d'une table de vérité, il suffit de placer la valeur de la sortie dans la case correspondante.

Exemple:

Soit la table de vérité suivante:

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

Un tableau de Karnaugh correspondant est:

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

- Il y a une équivalence complète entre une table de vérité et le tableau de Karnaugh associé.
- Le tableau de Karnaugh permet de visualiser des "groupes" de combinaisons adjacentes et donc facilite grandement la simplification.

Exercice 6.2 (Remplissage de tableaux de Karnaugh) *:*

Pour les tables de vérité suivantes, quels sont les tableaux de Karnaugh associés?

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

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

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

■

6.2.3 Remplissage d'un tableau de Karnaugh à l'aide d'une équation

Il est aussi possible de remplir un tableau de Karnaugh à partir d'une équation. Par exemple, si l'on part d'une expression du type $S = a + \bar{a}b + ab\bar{c} + abcd$ chaque terme représente une ou plusieurs cases.

- On sait qu'un minterme correspond à une case unique. Puisque l'équation contient un minterme, on place un "1" dans la case ou $a = 1$,$b = 1$,$c = 1$ et $d = 1$.
- Le terme $ab\bar{c}$ ne contient que trois variables: il représente donc 2 cases. On place un "1" dans les 2 cases où $a = 1$ $b = 1$ $c = 0$
- Le terme $\bar{a}b$ ne contient que deux variables: il représente donc 4 cases. On place un "1" dans les 4 cases où $a = 0$ et $b = 1$
- Le terme a ne contient qu'une variable: il représente donc 8 cases. On place un "1" dans les 8 cases où $a = 1$
- La où les cases sont restées vides, on place des 0

Exercice 6.3 (Remplissage de tableaux de Karnaugh à partir d'équation) *:*

Pour les équations suivantes, quels sont les tableaux de Karnaugh associés?

- $S = a + b + c\bar{d} + \bar{a}\bar{b}cd$
- $S = abc + a\bar{b} + a\bar{c}d$

■

6.3 Simplifier à l'aide des tableaux de Karnaugh

6.3.1 Les implicants

Simplifier à l'aide des tableaux de Karnaugh est littéralement "un jeu d'enfant" qui consiste simplement à faire des "groupes" de "1". Pourquoi faire des groupes? Prenons un exemple concret. En considérant le tableaux de Karnaugh précédent:

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

- Observez le "groupe" Rouge.
- En son sein, S est toujours égal à 1.
- Si on regarde les mintermes impliqués on peut en conclure que $S = 1$ si $\bar{a}\bar{b}\bar{c}d = 1$ OU $S = 1$ si $\bar{a}b\bar{c}d = 1$ OU $S = 1$ si $\bar{a}\bar{b}cd = 1$ OU $S = 1$ si $\bar{a}bcd = 1$. (Attention c'est un SI pas un SI ET SEULEMENT SI).
- On en conclut que $S = 1$ si $\bar{a}\bar{b}\bar{c}d + \bar{a}b\bar{c}d + \bar{a}\bar{b}cd + \bar{a}bcd = 1$.
- On peut simplifier cette expression avec les bonnes factorisations. $S = 1$ si $\bar{a}d = 1$ (vérifiez-le).

Autrement dit $\bar{a}d = 1$ IMPLIQUE que $S = 1$. Dans ce cas on on dit que $\bar{a}d$ est un **implicant** de S.

> **Définition:**
> On appelle une intersection de variables F, un implicant d'une fonction Booléenne S, si $F = 1$ implique $S = 1$.

En pratique, il est extrèmement facile de connaître un impliquant. Dans les mintermes de l'impliquant rouge, a est toujours égal à 0, d est toujours égal à "1". Les valeurs de c et de b varient et donc n'influent pas, au sein de ce groupe sur la sortie. Ainsi $S = 1$ si $a = 0$ ET $d = 1$ soit $S = 1$ si $\bar{a}d = 1$.

Pour trouver l'implicant correspondant à un groupe dans un tableau de Karnaugh, il suffit de trouver quelles variables ne varient pas et d'en faire l'intersection. Les variables étant égales à 0 seront complémentées (auront une barre).

Exercice 6.4 (Valeur d'un implicant) *:*

Pour les groupes suivants, quels sont les implicants associés?

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

$a\bar{b}cd$

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

$ac\bar{d}$

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

$\bar{a}b$

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

$\bar{a}$

■

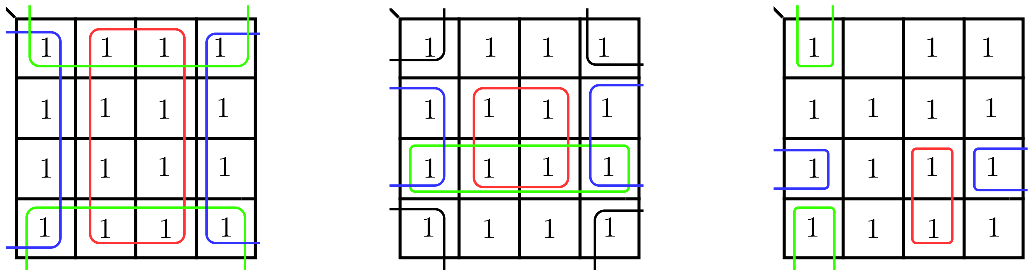
Conclusion, plus un groupe est grand, plus il simplifie l'expression Booléenne !

6.3.2 Etape 1: Trouver les implicants premiers

Le jeu dans un tableau de Karnaugh, est de trouver TOUS les implicants.

Un implicant, dans un tableau de Karnaugh, est forcément un rectangle de taille 2^n (1 2 4 8 16 ...).

Voici quelques exemples d'implicants à 8 case, 4 cases, et 2 cases (16 cases correspond au tableau entier et 1 case est trivial). **Faites toujours attention au coté cyclique du tableau!**



Définition: Un implicant est dit **premier**, si il ne peut pas être recouvert entièrement par des implicants de taille plus grande.

La stratégie est la suivante: Etape 1 trouver tous les implicants premiers

- On commence d'abord par chercher les implicants de plus grande taille ($2^n = 16$)
- Si il reste des "1" non couverts par des implicants, on cherche les implicants de taille ($2^{n-1} = 8$) qui couvrent ces "1"
- Si il reste des "1" non couverts par des implicants, on cherche les implicants de taille ($2^{n-2} = 4$) qui couvrent ces "1"
- Si il reste des "1" non couverts par des implicants, on cherche les implicants de taille ($2^{n-3} = 2$) qui couvrent ces "1"
- Si il reste des "1" non couverts par des implicants, on cherche les implicants de taille ($2^{n-4} = 1$) qui couvrent ces "1"

Exemple:

Soit le Tableau de Karnaugh suivant:

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

Il n'y a pas d'implicant de taille 16. On cherche les implicants premiers de taille 8:

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

Il y en a un : $\bar{a}$

On cherche ensuite les implicants premiers de taille 4:

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

Il y en a deux : cd et cb

On cherche ensuite les implicants premiers de taille 2:

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

Il y en a un : $\bar{b}\bar{c}\bar{d}$

Tous les "1" sont maintenant couverts, nous avons fini l'étape 1.

Exercice 6.5 (Implicants premiers) *:*

Pour les tableaux suivants, trouvez tous les implicants premiers:

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

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

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

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

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

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

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

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

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

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

■

6.3.3 Etape 2: Trouver les implicants premiers essentiels

Définition: Un implicant premier est dit **essentiel**, si il existe en son sein, des cases qui ne sont couvertes QUE par lui-même. Ces case, couvertes par un seul implicant, sont appelées **cases univcouvertes**.

Exemple

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

Il existe deux implicants premiers essentiels en lignes pleines : $\bar{a}\bar{c}$ (en bleu) et abc (en vert)
On peut dire donc que $S = 1$ SI $\bar{a}\bar{c} + abc = 1$. Attention ce n'est pas un SI ET SEULEMENT SI puisqu'il reste d'autres cas où $S = 1$ qui ne sont pas couverts par les implicants premiers essentiels.

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

Ici, tous les implicants premiers sont essentiels: il y d, $\bar{c}\bar{b}$, $\bar{a}bc$
On peut dire donc que $S = 1$ SI $d + \bar{c}\bar{b} + \bar{a}bc = 1$. ici puisque TOUS les "1" sont couvert, on peut transformer le SI en SI ET SEULEMENT SI (il n'y a pas d'autres cas) et donc directement: $S = d + \bar{c}\bar{b} + \bar{a}bc$.

■

Exercice 6.6 (implicants premiers essentiels) *:*

Pour les tableaux suivants, trouvez tous les implicants essentiels:

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

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

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

Indices (il y en 3 dans le premier tableau, 1 dans le deuxième et 4 dans le troisième)

■

6.3.4 Trouver l'équation minimale de sortie

Une équation minimale est par définition, une équation qu'on ne peut plus simplifier.
Une fois que l'on a trouvé les implicants premiers essentiels deux cas de figure se présentent:

A) Les implicants premiers essentiels couvrent TOUS les "1": Alors le SI se transforme en SI ET SEULEMENT SI, et la sortie S est égale à l'union des implicants essentiels premiers.

B) Les implicants premiers essentiels ne couvrent PAS TOUS les "1": Alors il faut garder des implicants premiers qui ne sont pas essentiels. La règle est la suivante:

- On garde les impliquants premiers qui permettent de couvrir TOUS LES "1"
- **On garde les plus grand.**
- **On en garde le moins possible.**

La sortie S est égale à l'union des implicants premiers (essentiels et non essentiels) conservés.

Dans le cas A) la forme minimale est toujours unique. Dans le cas B) il se peut qu'il y ait plusieurs formes minimales. Dans le cas B) il peut y avoir plusieurs solutions minimales. Il faut bien faire attention, les règles de choix imposent que:

- On préfère Deux implicants de 4 cases que deux implicants de 2 case.
- On préfère Deux implicants de 4 cases que un implicant de 4 cases et 1 implicant de 2 case.

Normalement le choix entre 2 implicants de 4 cases, et 1 implicant de 2 cases ne peut pas se présenter. Exemple:

Etape 1:
Impliquants essentiels:

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

Etape 2:
Impliquants essentiels premiers:

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

d, $\bar{c}\bar{b}$, $\bar{a}bc$

Etape 3: cas A)
Tous les "1" sont couverts

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

Solution minimale unique:
$S = d + \bar{c}\bar{b} + \bar{a}bc$

Etape 1:
Impliquants essentiels:

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

Etape 2:
Impliquants essentiels premiers:

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

$\bar{a}\bar{c}$, abc.

Etape 3: cas B)
On peut couvrir les "1" restants avec un seul bloc de deux (toutes les autres solution requièrent au moins deux groupes).

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

Solution minimale unique:
$S = \bar{a}\bar{c} + abc + a\bar{b}d$

Etape 1:
Impliquants essentiels:

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

Etape 2:
Impliquants essentiels premiers:

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

$\bar{a}c$.

Etape 3: cas B)
Il existe 2 possibilités de couvrir le reste des 1 avec les implicants essentiels et dans les deux cas, **on a besoin de 1 bloc de 4 et deux blocs de 2**:
Il y a donc deux solutions minimales:

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

Solution minimale 1:
$S = \bar{a}c + \bar{a}\bar{b} + b\bar{c}d + a\bar{c}\bar{d}$

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

Solution minimale 2:
$S = \bar{a}c + \bar{a}d + ab\bar{c} + \bar{b}\bar{c}\bar{d}$

On voit que dans les deux cas, l'implicant essentiel premier est présent.

6.4 Résumé de la méthode de Karnaugh

Afin de faire une simplification à l'aide du tableau de Karnaugh, il faut absolument savoir:

- **Construire** un tableau (En utilisant le code Gray)
- **Remplir** un tableau à partir d'une table de vérité ou d'un équation
- **Exprimer** des implicants en équation.
- **Trouver** TOUS les implicants premiers
- **Reconnaître** les implicants premiers essentiels
- **Formaliser** l'équation de sortie minimale.

Exercice 6.7 (Simplification à l'aide de Karnaugh) :

Reprenez les exercices 6.2 et 6.3 et donnez les équations minimales associées.

■

7 Synthèse de problèmes logiques

Jusqu'à présent nous avons vu les différentes représentations d'un système logique, et différentes manières de simplifier un système logique en utilisant l'Algèbre de Boole ou des outils graphiques reposant sur l'algèbre de Boole. Les grandes questions en suspend sont:

- Pourquoi n a t'on pas des équations simplifiées à la base?
- Comment met-on tout cela en pratique?

Clairement, cela ne sert à rien d'aller plus loin sans maîtriser les outils précédents (on ne peut pas construire une charpente si on ne sait pas planter un clou). A partir d'ici il est considéré que tous les éléments précédents sont compris et connus.

On a besoin de ces outils lorsqu'on souhaite synthétiser un système de logique combinatoire ou de logique séquentiel. Dans ce cours nous ne parlerons que des problèmes de logique combinatoire. Ces problèmes peuvent s'exprimer comme des besoins pour une application, ou simplement des règles successives que l'on aura besoin de traiter pour produire des solutions.

En général l'énoncé d'un problème logique se fait toujours de sorte qu'on obtient souvent une table de vérité (une liste de cas) ou une équation qui n'est pas simplifiée. Les applications sont nombreuses: Carte électroniques, Automobile, Domotique, Minecraft...
Par exemple dans une maison, on pourrait avoir besoin de fermer automatiquement les volets si "l'ensoleillement est trop fort " ET " il fait trop chaud dans la maison", ou "il fait trop froid" ET "il n'y a pas d'ensolleillement" (pour garder la chaleur). On peut commander les fenêtres qui doivent s'ouvrir si "la température de la maison est chaude" et "l'ensolleillement est faible" sauf si elle est au rez-de-chaussée, ou si "il y a trop de vent"". On voit que nous sommes pour l'instant loin d'une équation simplifiée. Pourtant, ici toutes les variables sont logiques et le système est combinatoire. Comment faire une carte électronique à partir de ces informations?

La méthodologie est ici encore simple:

1) Lister quelles sont les entrées du système et quelles sont les sorties?
2) Produire pour chaque sortie, une table de vérité.
3) Utiliser l'Algèbre de Boole pour donner une expression simplifiée.

En ce qui concerne l'étape 1), il suffit de se demander:

- Qu'est ce qui pilote mon système? ce sont les entrées logiques: Conditions, boutons, capteurs. Dans l'exemple précédent on voit que les entrées sont le vent (trop fort=1, pas fort= 0), le soleil (ensoleillé=1 pas de soleil=0), la température intérieure et extérieur ($T_{int} > T_{ext}$?),etc...
- Sur quoi mon système agit? Ce sont les sorties logiques: actionneurs, conséquences, verbe. Dans l'exemple précédent: Fermer les volets, Ouvrir les volets, ouvrir les fenêtres, fermer les fenêtres (1 elles se ferment, 0 elles s'ouvrent),...

Exemple (Examen 2017):

Un examen se compose de 4 questions. La question Q1 constitue la partie théorique, alors que les questions Q2 Q3 et Q4, constituent la partie pratique.

- La partie théorique est considérée comme validée dès lors qu'il a répondu correctement à Q1.
- La partie pratique est considérée comme validée, dès lors qu'il réussi soit la Q4 (qui est très difficile) soit qu'il réussi les deux questions Q2 et Q3 simultanément.

Un élève peut valider l'examen , être accepté au rattrapage ou être invalidé:

- Il valide l'examen si il valide la partie théorique ET la partie pratique.
- Il est accepté au rattrapage dans deux cas:
    - Soit il valide la partie théorique, mais pas la partie pratique. Il doit dans ce cas tout de même avoir répondu à l'une des deux questions Q2 ou Q3.
    - Soit il ne valide pas la partie théorique, mais il valide la partie pratique.
- Il est invalidé à l'examen dans les autres cas.

Quelles sont les entrées, quelles sont les sorties?

- Les entrées: C'est qui conditionne l'examen: si l'élève répond ou non aux questions. Ce sont les conditions: on peut alors définir 4 conditions Q_1 (=1 si l'élève répond correctement à la question 1), et de même pour Q_2,Q_3 et Q_4.
- Les sorties: Sur quoi vont agir les conditions: Sur le passage ou non de l'élève. On peut alors définir 3 sorties: V (=1 si lélève valide), R (=1 si l'élève va au rattrapage) et I (=1 si l'élève est invalidé.)

Dès lors on sait qu'on aura besoin de 3 tables de vérités à 4 entrées, puisque on a 3 sorties et 4 entrées.

Q_4	Q_3	Q_2	Q_1	V	R	I
0	0	0	0	0	0	1
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	0
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	1	0
1	1	0	1	1	0	0
1	1	1	0	0	1	0
1	1	1	1	1	0	0

Ici le but est de regarder, ligne par ligne dans une configuration particulière si l'élève valide passe au rattrapage ou est invalidé.
Bien souvent, dans ce genre d'exercice il faut se servir de la logique pour vérifier. Ici par exemple ne peut pas valider et être invalidé en même temps. Ceci signifie que qu'il ne peut y avoir qu'une seule sortie égale à 1 sur chaque ligne.

On aurait pu également directement définir des équations pour I, V et R Par exemple si P_T correspond à "valide la partie pratique" et P_P correspond à "valide la partie pratique", d'après l'énoncé, on peut conclure par interprétation que:

- $P_T = Q_1$
- $P_P = Q_4 + Q_2Q_3$
- $V = P_TP_P = Q_1(Q_4 + Q_2Q_3)$
- $R = P_T\overline{P_P} + \overline{P_T}P_P = Q_1.\overline{Q_4 + Q_2Q_3} + \overline{Q_1}(Q_4 + Q_2Q_3)$
- $I = \bar{V}\bar{R} = \overline{Q_1.(Q_4 + Q_2Q_3)}.\overline{Q_1.\overline{Q_4 + Q_2Q_3} + \overline{Q_1}(Q_4 + Q_2Q_3)}$

Qu'on obtienne des équations ou une table de vérité, On peut ensuite utiliser l'Algèbre de Boole ou les tableaux de Karnaugh afin de simplifier ces expressions: Ici on obtiendra commes equations minimales:

- $V = P_T P_P = Q_1 Q_4 + Q_1 Q_2 Q_3$
- $R = \overline{Q_1} Q_4 + Q_2 Q_3 \overline{Q_1} + Q_1 Q_2 \overline{Q_3 Q_4} + Q_1 \overline{Q_2} Q_3 \overline{Q_4}$
- $I = \overline{Q_1 Q_2 Q_4} + \overline{Q_1 Q_3 Q_4} + \overline{Q_2 Q_3 Q_4}$

On pourrait en faire un programme informatique qui en fonction des résultats décide de quels élèvent passent ou non grâce à ces expressions logiques.
Une question subsidiaire était : "Quelles sont les conditions pour qu'un élève soit au moins au rattrapage":
Ici on peut formuler le problème de deux façons:

- "Au moins au rattrapage"$= V + R$
- "Au moins au rattrapaga"$=$ "pas invalidé"$= \bar{I}$

Dans les deux cas on peut utiliser ces expressions pour les simplifier et trouver que:

- $V + R = Q_4 + Q_1 Q_2 + Q_1 Q_3 + Q_2 Q_4$

■

7.1 Problèmes avec cas impossibles ou facultatifs

Il peut arriver dans un problème que, lors du remplissage de la table de vérité, il soit impossible de décider pour certains mintermes, si la sortie est à "1" ou à "0". **On parle de cas indéterminé.** Ceci peut venir d'une impossibilité technique ou d'un cas qu'on ne considère pas comme pouvant arriver, ou tout simplement d'un cas ou c'est égal que la sortie soit à 1 ou à 0.

> Lorsque dans une table de vérité ou un tableau de Karnaugh, la sortie est un cas indéterminé, on indique la sortie avec un "X". Lorsqu'on détermine les implicants premiers dans un tableau de Karnaugh, **les indéterminés** sont considérés comme des "1". Puis, une fois tous les implicants essentiels sont trouvés, **on élimine d'office tous les implicants ne contenant QUE des indéterminés.** Le reste de la simplification par Karnaugh se passe de la même façon que lorsqu'il n'y a pas de cas indéterminé.

L'exercice suivant est un exemple, où il y aura des cas indéterminés (lorsque le niveau de la cuve est sensé être en haut ET en bas en même temps).

Exercice 7.1 (Problème: Chauffage d'une cuve) *:*

Le niveau d'une cuve est contrôlé par 2 capteurs de niveau haut et bas (n_b et n_h respectivement) et 2 capteurs de température haute et basse (t_h et t_b respectivement). Une vanne V permet le remplissage, tant que le niveau haut n'est pas atteint. Une résistance chauffante R assure le chauffage jusqu'à la température maximale.

Une sécurité de fonctionnement interdit le chauffage si le niveau bas est atteint. De même le remplissage est arrêté si la température minimale est atteinte. Dans le cas où le niveau ET la température sont basses alors c'est le remplissage qui est effectué, sans chauffage.

Les capteurs n_b, n_h sont à l'état 1 si le liquide est présent devant le capteur. (Si le liquide passe sous le niveau bas $n_b = 0$) Les capteurs de température t_h (respectivement t_b) sont à l'état 1 si la température du liquide est supérieure à la température haute (respectivement la température basse).

Questions:

- *Quelles sont les entrées et sorties de ce système logique?*
- *Décrire le fonctionnement par une table de vérité.*
- *Déterminer les équations de fonctionnement en utilisant les tableaux de Karnaugh.*

■

7.2 Exercice de synthèse de problèmes logiques

Exercice 7.2 (Problèmes non résolus) *:*

Compteur Digital:

On souhaite réaliser l'affichage d'un compteur digital affichant des chiffres entre 0 et 7. Pour cela on utilise 8 segments nommés de a à g comme suit:

Sachant que le nombre décimal que l'on doit afficher est codé sur 3 bits b_3, b_2 et b_1 (par exemple, la combinaison $b_3 = 1$, $b_2 = 1$ et $b_1 = 0$ affiche le chiffre 6).

- *Quelles sont les entrées et quelles sont les sorties de ce système?*
- *Est-ce un système logique? Pourquoi?*
- *Si oui, écrire la table de vérité et la simplifier pour les segments a et g.*

Fabrique de briques:

Dans une usine de briques, on effectue un contrôle de qualité selon quatre critères: Le poids P, la longueur L, la largeur M, la hauteur H (0 incorrect, 1 correct). Cela permet de classer les briques en trois catégories :

- *QUALITE A le poids P et deux dimensions au moins sont corrects.*
- *QUALITE B le poids seul est incorrect ou, le poids étant correct, deux dimensions au moins sont incorrectes.*
- *QUALITE C : le poids P est incorrect ainsi qu'une ou plusieurs dimensions.*

Questions:

- *Ecrire les équations des fonctions A, B, C.*
- *Simplifier ces fonctions par la méthode de votre choix.*
- *Dessiner le logigramme.*

Phares de voiture:

On dispose. sur une automobile, de quatre commandes indépendantes: C_V pour les veilleuses, C_C pour les deux phares de croisement, C_R pour les deux phares de route, C_A pour les deux phares antibrouillard (valeur 1 au travail, 0 au repos). On note les états des lumières V pour les veilleuses, C pour les feux de croisement, R pour les feux de route, A pour les feux antibrouillard (valeur 1 l'allumage, 0 à l'extinction). Les veilleuses ntant pas comptes comme des phares, il est prcis que:

- *4 phares ne peuvent être allumés simultanément,*
- *les feux de croisement ont priorité sur les feux de route et sur les antibrouillard,*
- *les antibrouillard ont priorité sur les feux de route,*
- *les veilleuses peuvent être allumées seules mais l'allumage des feux de croisement ou des feux de route ou des antibrouillard entraîne obligatoirement l'allumage des veilleuses.*

Donner la table de vérité liant V,C,R,A à C_V ,C_C ,C_F,C_A
Simplifier ces fonctions à l'aide de tableaux de Karnaugh.
■

8 Conclusion

Voici les compétences que vous avez normalement acquises durant ce chapitre:

- Savoir ce qu'est un système et savoir le représenter un système.
- Connaitre les représentations usuelles de la logique combinatoire et savoir passer de l'une à l'autre (Logigramme, Contact, Equation, Table de vérité).
- Connaître de l'algèbre de Boole dans ses différentes représentations : Union, Intersection, Négation.
- Connaître les théorèmes de l'algèbre de Boole et savoir les utiliser pour simplifier.
- Savoir simplifier par Tableau de Karnaugh.
- Savoir synthétiser et résoudre un problème logique.

Dans le chapitre suivant, nous verrons comment passer de variables logiques à des variables numériques binaires: En d'autres termes, comment, à partir de simples variables Booléennes, est-il possible de faire toutes les opérations que nous connaissons, comme compter, additionner, multiplier, ... ? Nous essayerons donc de comprendre comment il est possible de manipuler des nombres qui ne sont pas écrits dans la base que nous connaissons le mieux: La base décimale.