ELEN0040 - Electronique numérique

FRED SENNY

~~Patricia ROUSSEAUX~~

2017-2018

Année académique ~~2014 - 2015~~

CHAPITRE 2

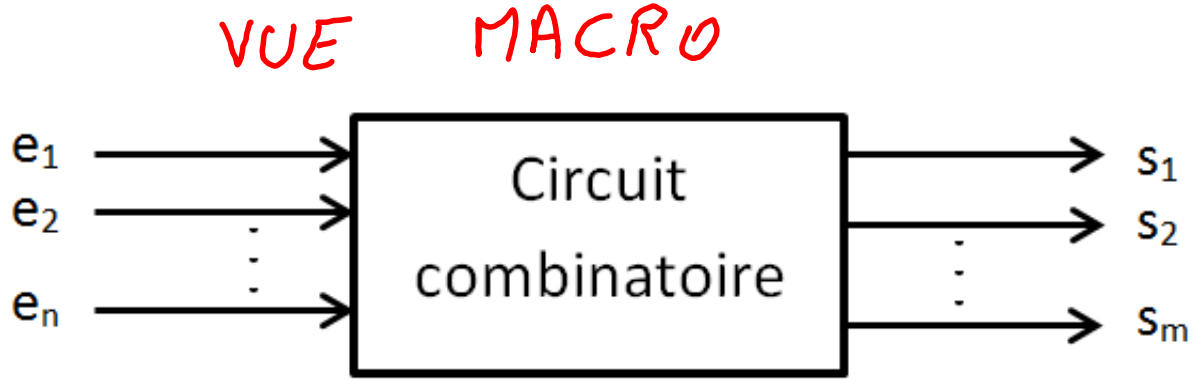
Conception de circuits combinatoires

MENU

1 Circuits combinatoires
1.1 Définitions
1.2 Design d'un circuit

2 Simplification de fonctions
2.1 Simplification par table de Karnaugh
Tables de Karnaugh, 2 à 4 variables
Impliquants premiers
Fonction non complètement spécifiée - Don't care
Simplification sous forme d'un produit de sommes
Circuits à 2 niveaux
2.2 Circuits à plus de deux niveaux

Circuit combinatoire

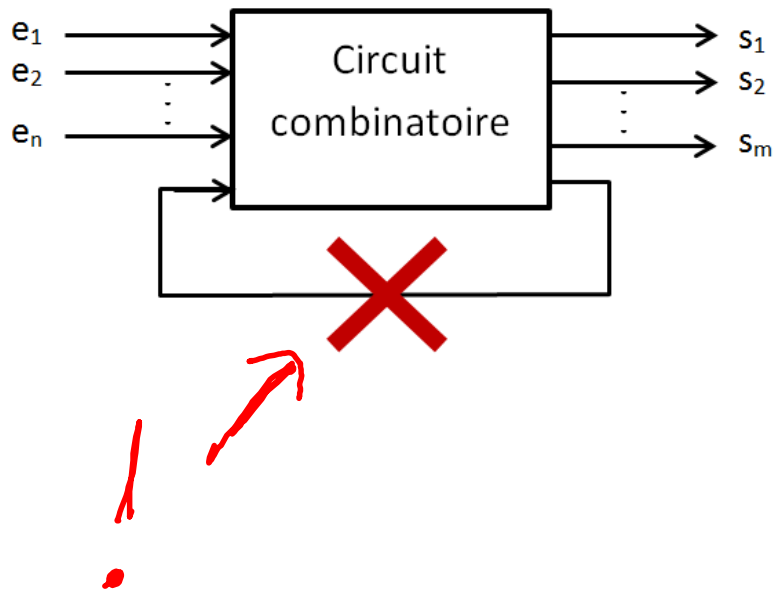


Une circuit combinatoire :

- est activé par n signaux d'entrée numériques $= n$ variables booléennes $e_1, \cdots, e_n$
- fournit m signaux de sortie numériques $= m$ variables booléennes $s_1, \cdots, s_m$
- est constitué de m circuits logiques, chacun réalisant une fonction logique $s_j = F(e_1, \cdots, e_n)$, $j = 1, \cdots, m$

Absence de “mémoire”

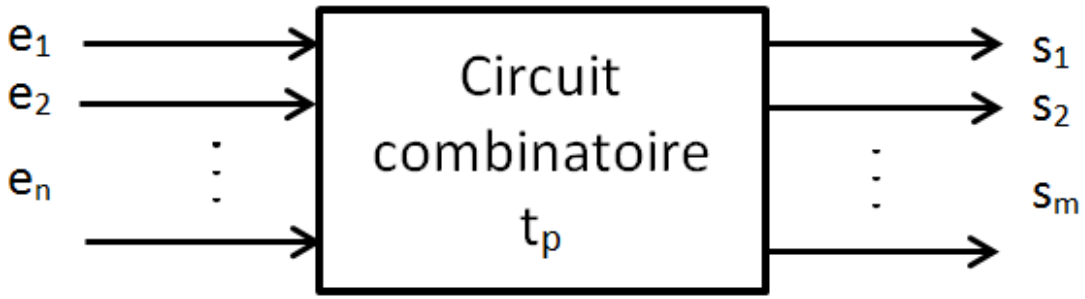
- Dans un circuit combinatoire, il n’existe jamais de boucle de retour des sorties vers l’entrée du circuit
- Cela nécessiterait des éléments “tampon” capables de mémoriser l’état de certaines variables
- Les opérations de mémorisation d’état sont à la base des circuits séquentiels



Délai de traitement

- Les signaux de sortie ne sont pas obtenus instantanément
- Chaque porte du circuit impose un délai
- Le délai total t_p résulte de l'addition des délais associés aux portes rencontrées successivement sur un chemin partant de l'entrée vers la sortie
- Délai du circuit = délai maximum rencontré

CASCADE DOMINO

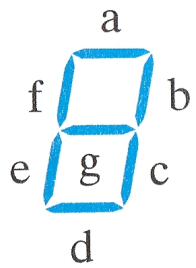


Procédure de design d'un circuit

1. Spécification :
   - décrire ce que le circuit doit réaliser
2. Formulation :
   - définir les variables d'entrée et de sortie
   - établir la table de vérité et les fonctions logiques définissant les sorties
   - recours aux formes canoniques SOM et POM
3. Optimisation :
   - simplification des fonctions
   - satisfaire un compromis entre différents critères : nombre de composants, délai, ...
   - tracer le diagramme logique au moyen des portes primitives AND, OR et NOT
4. Adaptation à la technologie choisie :
   - choisir une technologie pour la réalisation du circuit : exemple CMOS et portes NAND, portes NOR, logique programmable, ...
   - tracer le diagramme du circuit avec les composants correspondants
5. Vérification :
   - vérifier le bon fonctionnement du circuit : souvent via un outil de simulation (voir labos et projet)

Spécification

Exemple : afficheur LED 7 segments



LEDS allumées selon le chiffre à afficher :

- 0 : a, b, c, e, f
- 1 : b, c
- 2 : a, b, g, e, d
- ...

Formulation

Entrées : chiffre à afficher, code BCD (Binary Coded Decimal = decimal codé binaire) : 4 variables binaire A, B, C, D

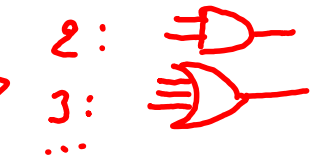
Sorties : une variables binaire pour chaque LED (0=éteint, 1=allumé) : a, b, c, d, e, f, g

Table de vérité :

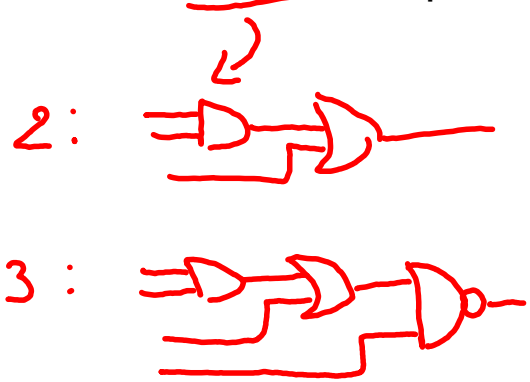
BCD Input				Seven-Segment Decoder						
A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1
All other inputs				0	0	0	0	0	0	0

Optimisation

- But : obtenir l'implémentation la plus simple possible d'une fonction
- simple ? : différents critères, ne pouvant être satisfaits simultanément $\Rightarrow$ compromis
- Critères : définition d'un coût
  - nombre d'entrées totales des portes du circuit, *gate input cost G*
  - G tient à la fois compte du nombre de portes et du nombre d'entrées pour chaque porte
  - le délai introduit par le circuit : lié au nombre de niveaux de portes



VOIR EX APRÈS



Optimisation – Exemple

Considérons la réalisation de la variable a de l'afficheur 7-segments
D'après la table de vérité :

$$a = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}CD + \bar{A}B\bar{C}D +$$
$$+ \bar{A}BC\bar{D} + \bar{A}BCD + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D$$
$$= \sum m(0, 2, 3, 5, 6, 7, 8, 9) \quad (*)$$

Σ de Π
= SOM

On peut par exemple regrouper les mintermes 8 et 9

$$A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D = A\bar{B}\bar{C}.(\bar{D} + D) = A\bar{B}\bar{C}$$

De même pour les mintermes 5 et 7

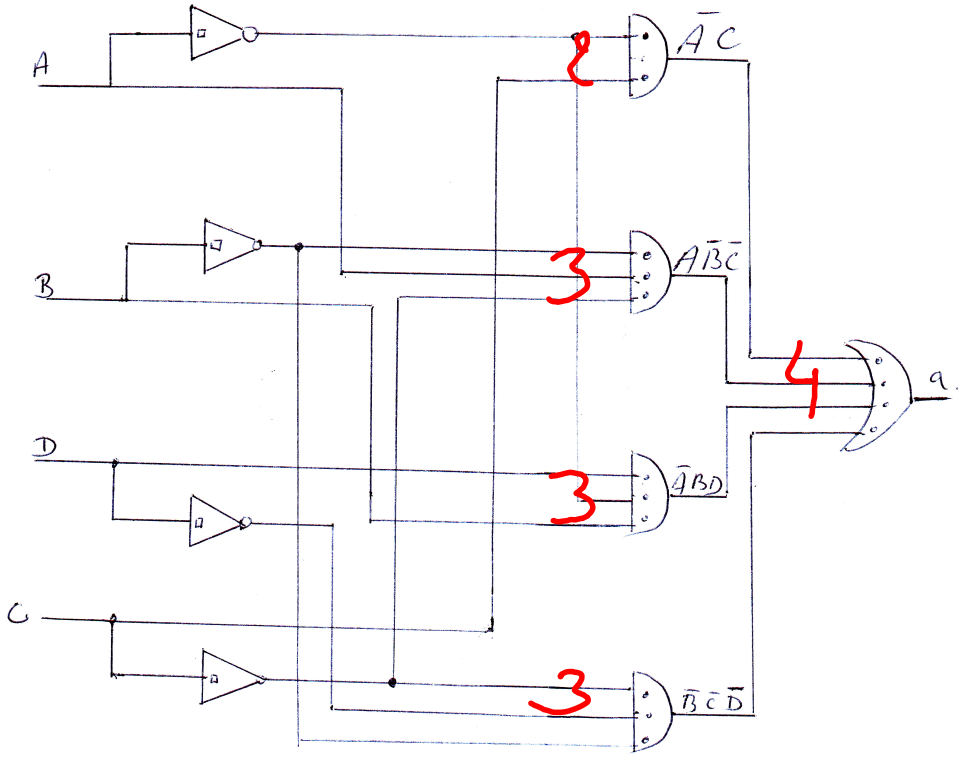
$$\bar{A}B\bar{C}D + \bar{A}BCD = \bar{A}BD.(\bar{C} + C) = \bar{A}BD$$

Nous verrons plus loin comment poursuivre (Karnough) pour obtenir

BREF, ON A $a = \bar{A}C + \bar{A}BD + \bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}$

Réalisation du circuit

AND / OR / INV



Gate input cost : $G = 15 + 4$ inverseurs (*)

Si l'on avait implémenter directement la forme SOM, on aurait

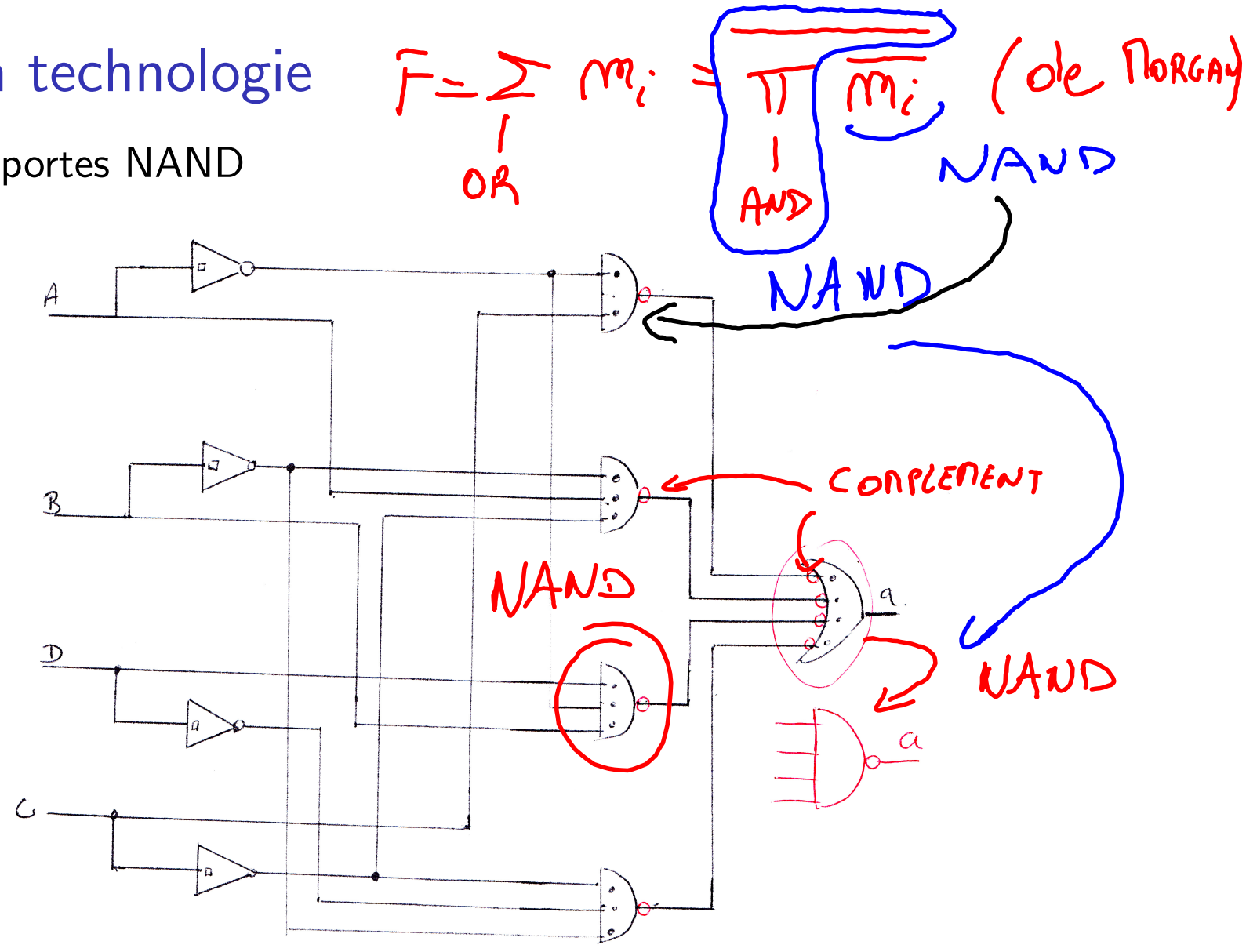
$G = 8 \times 4 + 1 \times 8 = 40 + 4$ inverseurs

Au niveau du délai, dans les deux cas, circuits à 2 niveaux de portes (+ inverseurs).

8 x 4-AND
1 x 8-OR

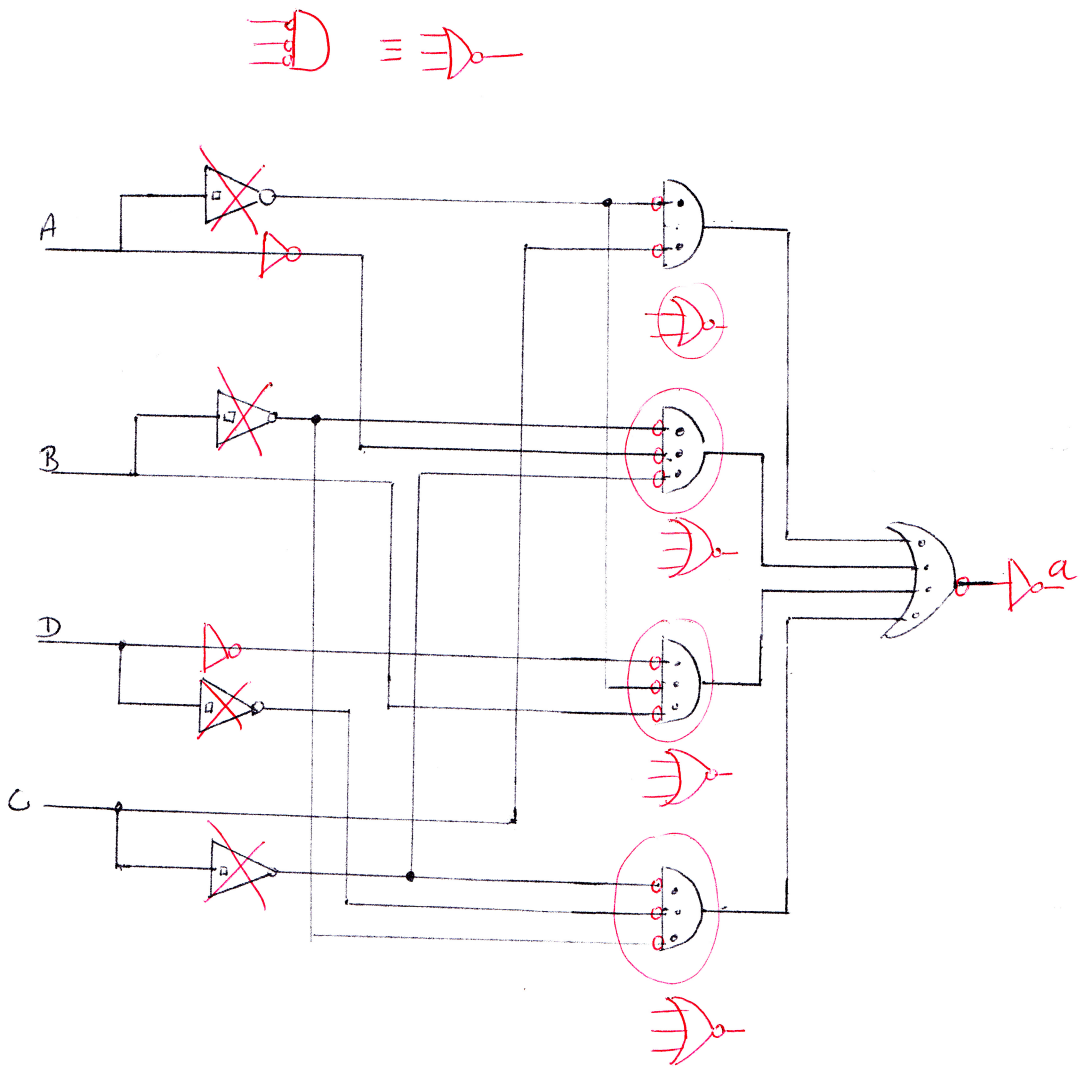
Choix de la technologie

Exemple : portes NAND



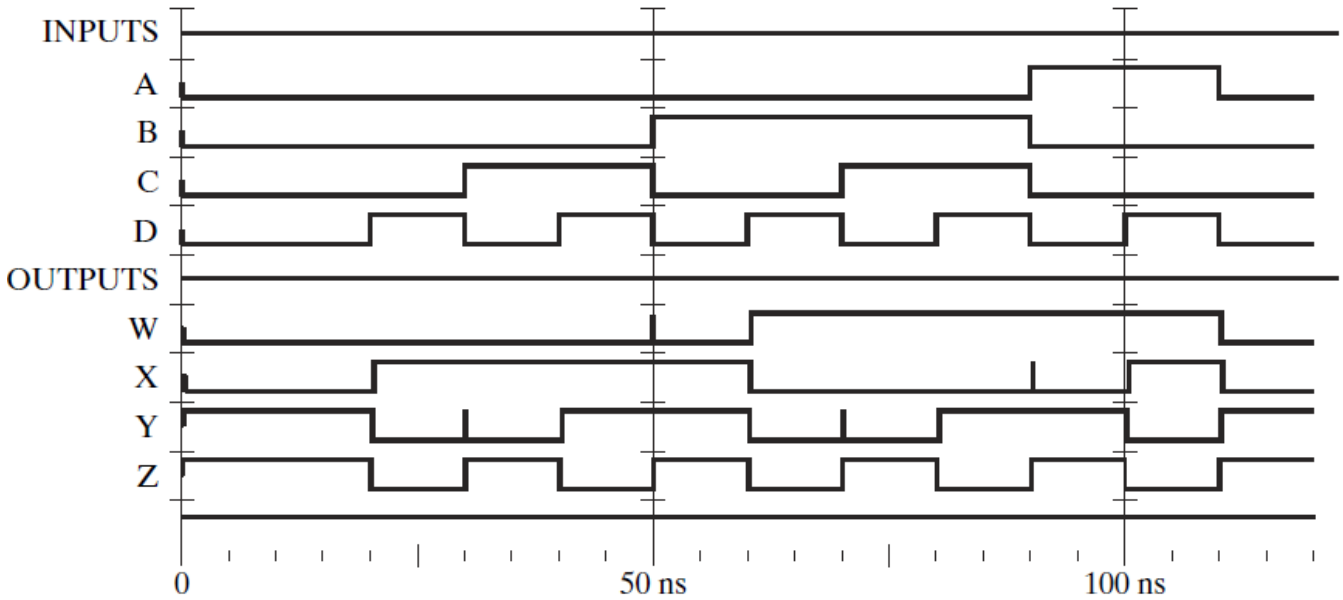
Choix de la technologie

Exemple : portes NOR



Vérification

- Une fois le schéma logique correspondant à la technologie obtenu, il faut vérifier que le circuit satisfait les spécifications de départ.
- Cette opération se réalise le plus souvent via un simulateur de circuits (voir labos VHDL)
- Exemple : simuler une séquence d'entrées générant toutes les combinaisons intéressantes
- Ici, délais non représentés
- Eviter deux changements simultanés dans les entrées !



1 Circuits combinatoires
1.1 Définitions
1.2 Design d'un circuit

2 Simplification de fonctions
2.1 Simplification par table de Karnaugh
Tables de Karnaugh, 2 à 4 variables
Impliquants premiers
Fonction non complètement spécifiée - Don't care
Simplification sous forme d'un produit de sommes
Circuits à 2 niveaux
2.2 Circuits à plus de deux niveaux

Table de Karnaugh

- Une table de Karnaugh est une représentation d'une expression logique dans une table qui met en évidence des termes "adjacents" pouvant être simplifiés
- Elle est formée d'un nombre de cases égal au nombre maximal de mintermes correspondant au nombre de variables apparaissant dans l'expression
- Deux cases sont adjacentes si les deux mintermes correspondants ne diffèrent que d'un littéral : il apparaît sous forme complémentée dans l'un et non complémentée dans l'autre
- Utile pour la simplification de fonctions de 2 à 6 variables
- Au-delà, représentation "géométrique" difficile
- Recours à des méthodes algorithmiques : algorithme de Quine–McCluskey, méthode de Petrick, heuristique de minimisation "Espressso"

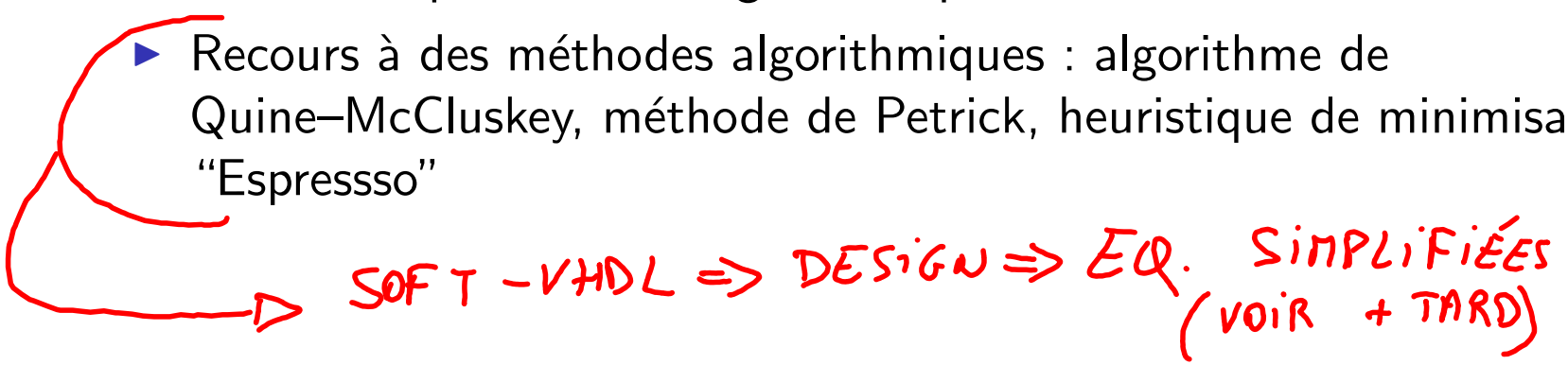
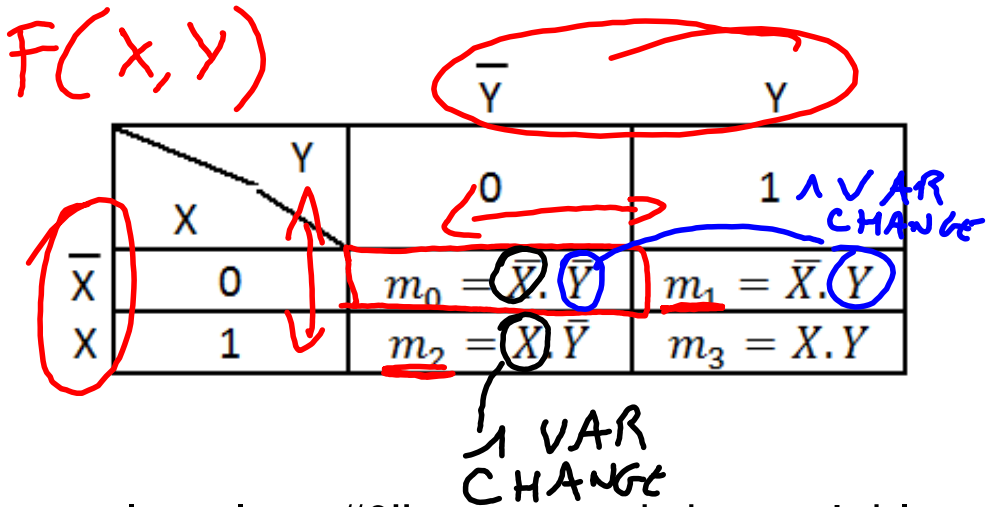


Table de Karnaugh à 2 variables

Les quatre mintermes sont placés comme suit :

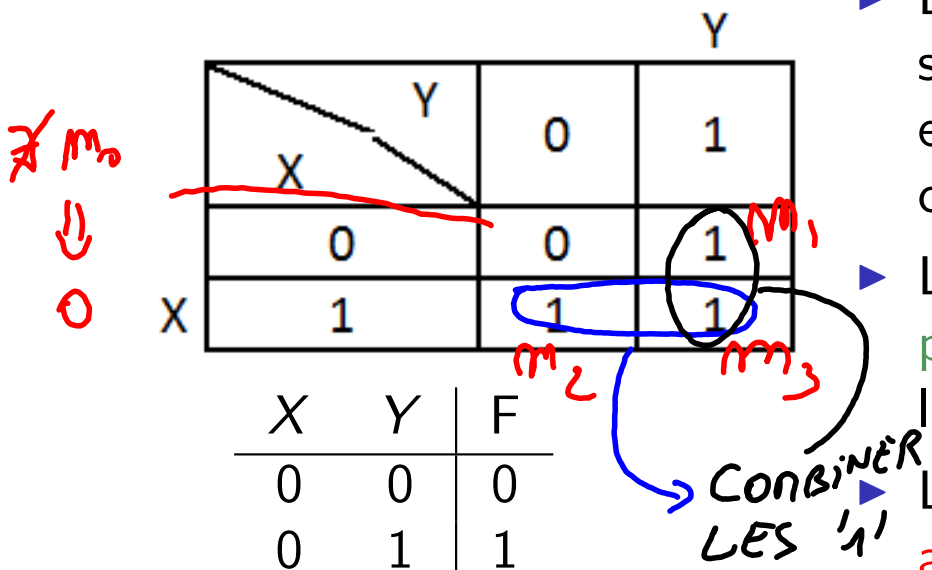


On place également la valeur “0” en regard des variables complémentées et la valeur “1” en regard des variables non complémentées

- Les mintermes m_0 et m_1 sont adjacents, ils différent par la variable Y
- De même,
  - m_0 et m_2 sont adjacents, ils diffèrent par la variable X
  - m_1 et m_3 sont adjacents, ils diffèrent par la variable X
  - m_2 et m_3 sont adjacents, ils diffèrent par la variable Y

Table de Karnaugh et table de vérité

Soit $F = m_1 + m_2 + m_3 = \bar{X}Y + X\bar{Y} + XY = X + Y$

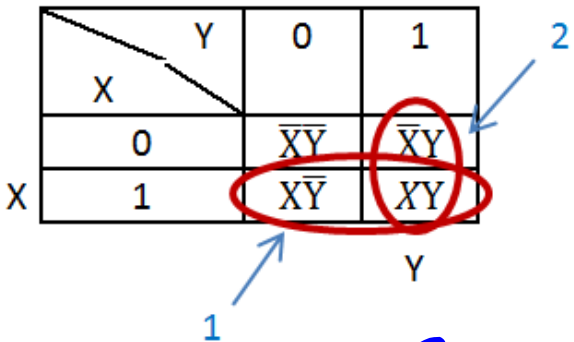


X	Y	F
0	0	0
0	1	1
1	0	1
1	1	1

- Les valeurs de l'expression booléenne sont placées dans la table aux emplacements des mintermes correspondants
- Les "1" correspondent aux mintermes présents dans la forme SOM de l'expression
- Les "0" correspondent aux mintermes absents dans la forme SOM de l'expression

Simplification de la fonction

$F = m_1 + m_2 + m_3$



$\bar{X}Y + XY = Y$

COMBINER = OR

$X\bar{Y} + XY = X$

1. La somme des deux mintermes adjacents m_2 et m_3 donne :

$$X\bar{Y} + XY = X.(\bar{Y} + Y) = X$$

2. La somme des deux mintermes adjacants m_1 et m_3 donne :

$$\bar{X}Y + XY = (\bar{X} + X).Y = Y$$

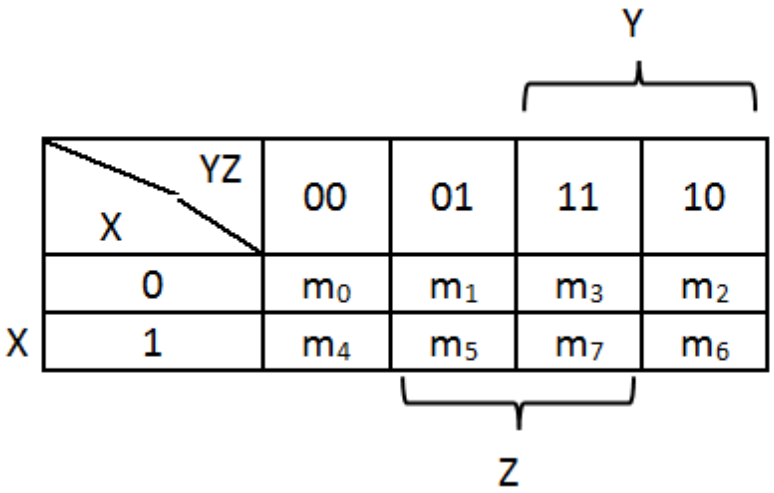
3. Etant donné que l'on peut écrire :

$$F = m_1 + m_2 + m_3 + m_3$$

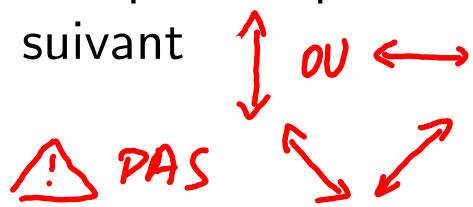
4. La forme simplifiée de F est :

$$F = X + Y$$

Table à 3 variables



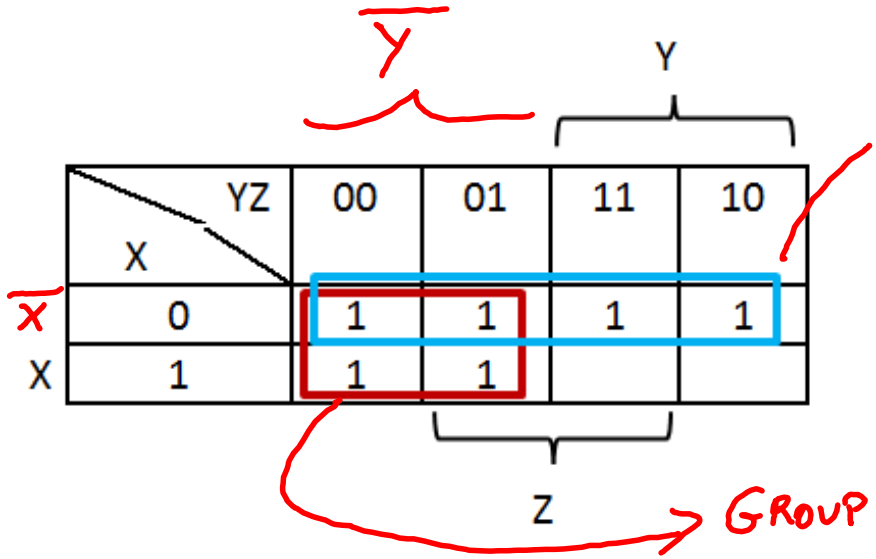
- Remarquez l'emplacement des mintermes !
- Les indices de YZ sont ordonnés selon un code de Gray cyclique : un seul changement lorsque l'on passe d'un élément au suivant



- Les mintermes de la première colonne sont adjacents aux mintermes de la quatrième colonne :
  - m_0 adjacent à m_2
  - m_4 adjacent à m_6
- Des groupement de 4 mintermes adjacents sont possibles
  - $m_1 + m_3 + m_5 + m_7$ donne Z
  - $m_0 + m_2 + m_4 + m_6$ donne $\bar{Z}$
  - $m_0 + m_1 + m_3 + m_2$ donne $\bar{X}$, ...

Exemple à 3 variables

Simplifiez la fonction : $F = \sum m(0, 1, 2, 3, 4, 5)$



1. Placer les 1 correspondants aux mintermes dans la table

2. Grouper $m_0 + m_1 + m_4 + m_5$ (rouge) :

$$\bar{X}\bar{Y}\bar{Z} + \bar{X}\bar{Y}Z + X\bar{Y}\bar{Z} + X\bar{Y}Z = \bar{Y}$$

3. Grouper $m_0 + m_1 + m_3 + m_2$ (bleu)

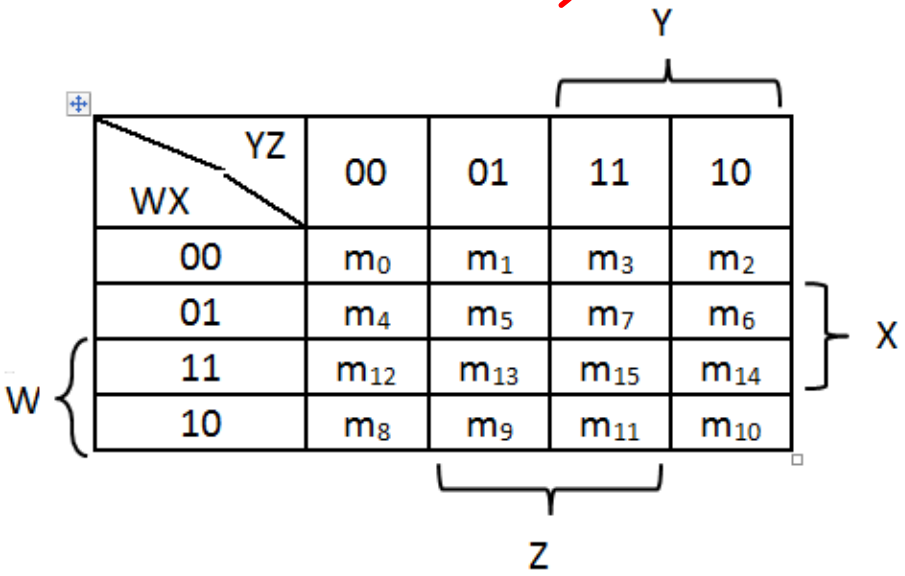
$$\bar{X}\bar{Y}\bar{Z} + \bar{X}\bar{Y}Z + \bar{X}YZ + \bar{X}Y\bar{Z} = \bar{X}$$

4.

$$F = \bar{X} + \bar{Y}$$

Table à 4 variables

F(W, X, Y, Z)

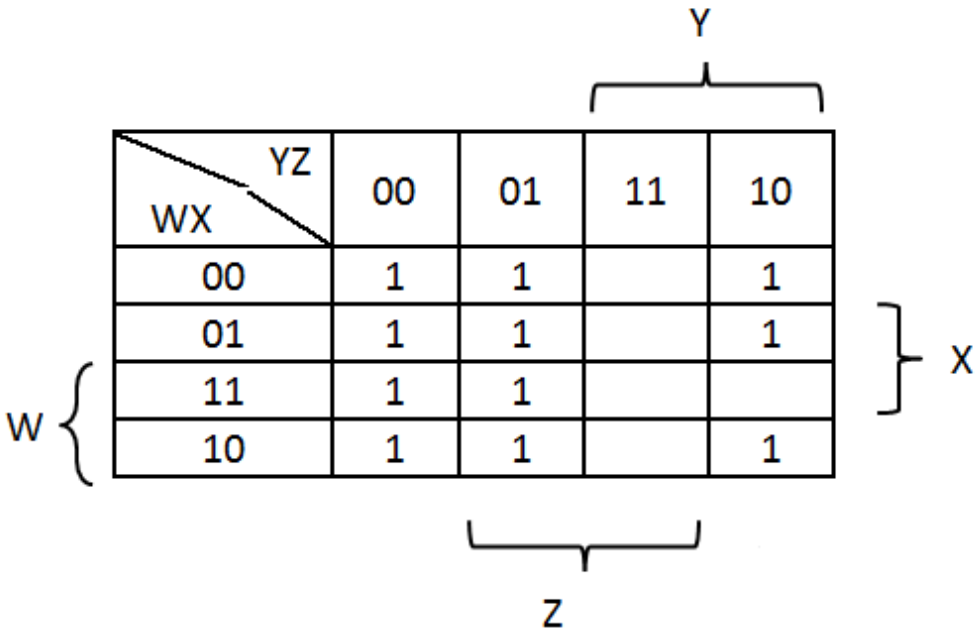


- Ordonnancement des mintermes selon le code de Gray

- Les mintermes de la première colonne sont adjacents aux mintermes de la quatrième colonne
- Les mintermes de la première ligne sont adjacents aux mintermes de la quatrième ligne
- Des groupement de 2, 4 et 8 mintermes adjacents sont possibles

Exemple à 4 variables

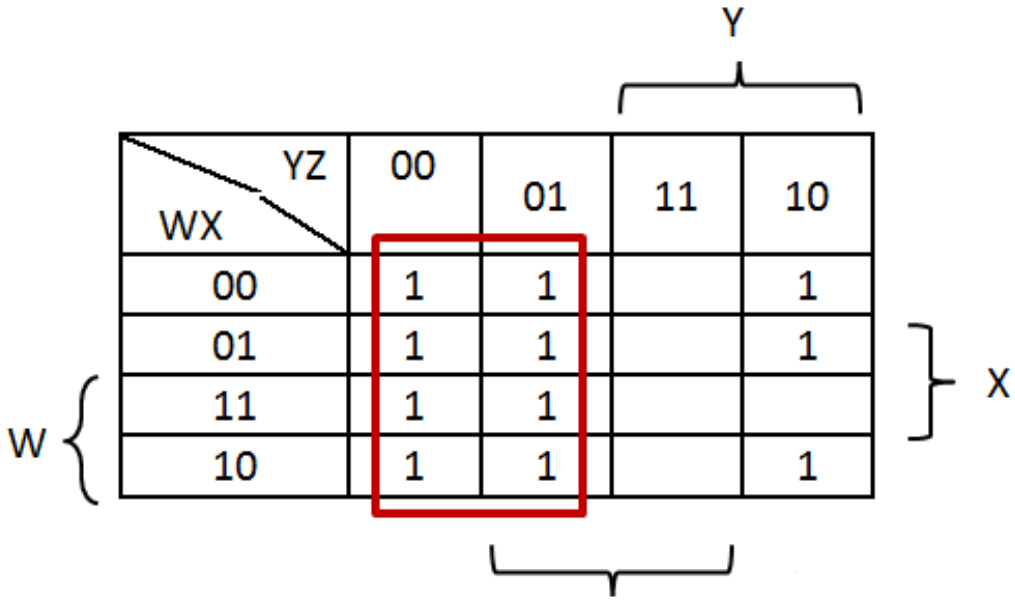
Simplifiez la fonction : $F = \sum m(0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13)$



1. Placer les 1 correspondants aux mintermes dans la table
2. Grouper les 8 mintermes des deux premières colonnes, on obtient : $\bar{Y}$
3. Grouper les 4 mintermes des 4 coins, on obtient : $\bar{X}\bar{Z}$
4. Grouper 4 mintermes permettant d'incorporer le dernier minterme non couvert : $\bar{W}\bar{Z}$

Exemple à 4 variables

Simplifiez la fonction : $F = \sum m(0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13)$

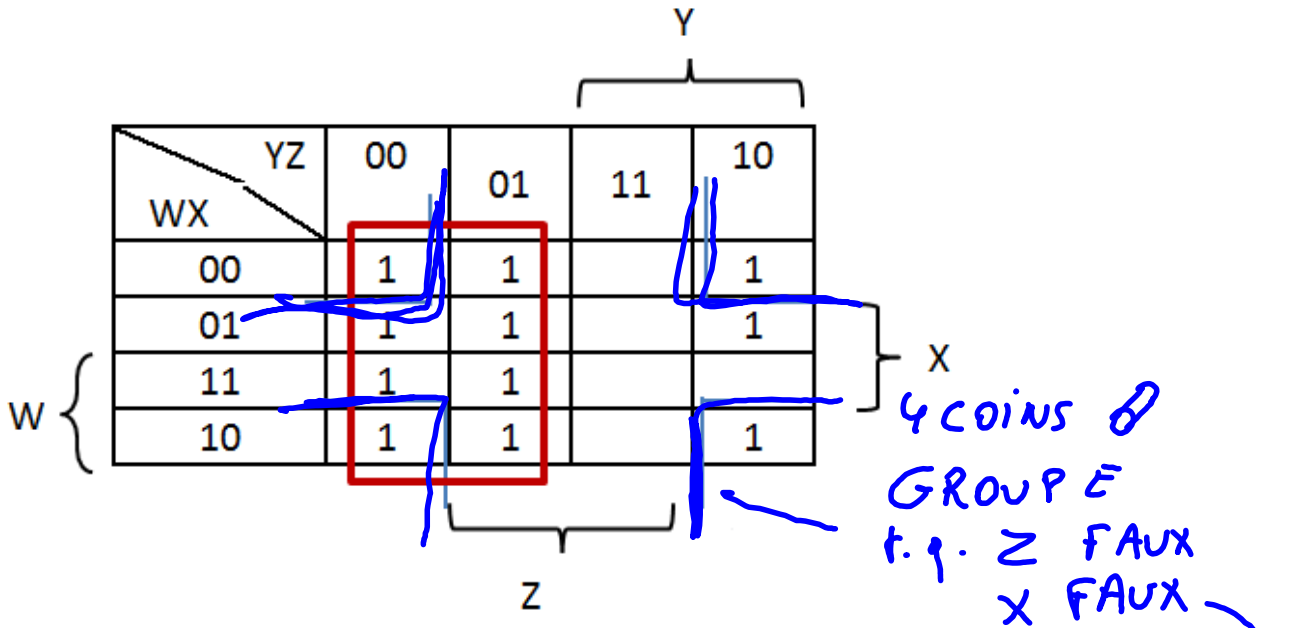


TOUJOURS CHERCHER GROUPE LE + GRAND (=>+SIMPLE)

1. Placer les 1 correspondants aux mintermes dans la table
2. Grouper les 8 mintermes des deux premières colonnes, on obtient : $\bar{Y}$
3. Grouper les 4 mintermes des 4 coins, on obtient : $\bar{X}\bar{Z}$
4. Grouper 4 mintermes permettant d'incorporer le dernier minterme non couvert : $\bar{W}\bar{Z}$

Exemple à 4 variables

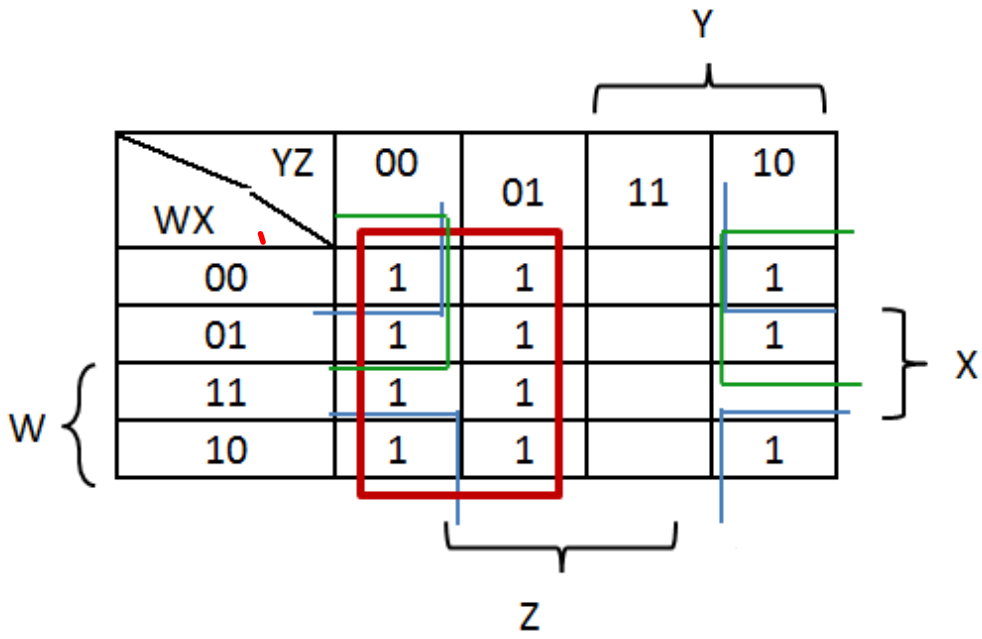
Simplifiez la fonction : $F = \sum m(0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13)$



1. Placer les 1 correspondants aux mintermes dans la table
2. Grouper les 8 mintermes des deux premières colonnes, on obtient : $\bar{Y}$
3. Grouper les 4 mintermes des 4 coins, on obtient : $\bar{X}\bar{Z}$
4. Grouper 4 mintermes permettant d'incorporer le dernier minterme non couvert : $\bar{W}\bar{Z}$

Exemple à 4 variables

Simplifiez la fonction : $F = \sum m(0, 1, 2, 4, 5, 6, 8, 9, 10, 12, 13)$



1. Placer les 1 correspondants aux mintermes dans la table
2. Grouper les 8 mintermes des deux premières colonnes, on obtient : $\bar{Y}$
3. Grouper les 4 mintermes des 4 coins, on obtient : $\bar{X}\bar{Z}$
4. Grouper 4 mintermes permettant d'incorporer le dernier minterme non couvert : $\bar{W}\bar{Z}$

$$F = \bar{Y} + \bar{X}\bar{Z} + \bar{W}\bar{Z}$$

Remarques

- Pour une fonction de n variables, un groupement de 2^k , $(k < n)$ mintermes permet de remplacer ces mintermes par un seul produit de $n - k$ variables
- Pour simplifier au maximum la fonction, intérêt de rechercher les groupements les plus grands possibles = impliquants premiers
- Chaque minterme de la fonction doit être inclus si possible dans au moins un groupement (1, 2, 4, 8, ...)
- Stopper la recherche de groupements lorsque tous les mintermes sont "couverts"

Impliquants

Pour une fonction booléenne donnée et sa table de Karnaugh :

Impliquant premier

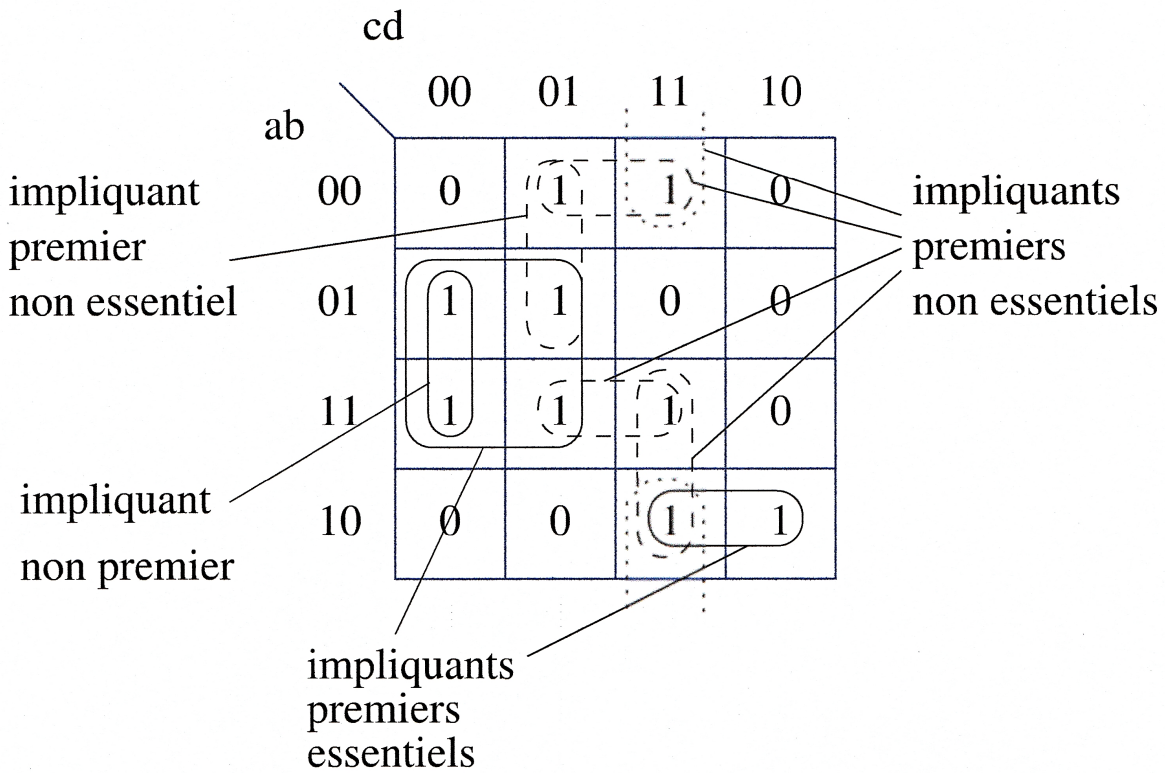
Un impliquant premier est un produit obtenu en combinant le plus grand nombre 2^k possible de mintermes adjacents de sorte que ce produit ne peut plus être simplifié.

Impliquant premier essentiel

Un impliquant premier est un impliquant premier essentiel si il contient un minterme qui n'est inclus dans aucun autre impliquant premier.

La suppression de cet impliquant ne permet plus de couvrir tous les mintermes de la fonction : impliquant essentiel.

Impliquants premiers



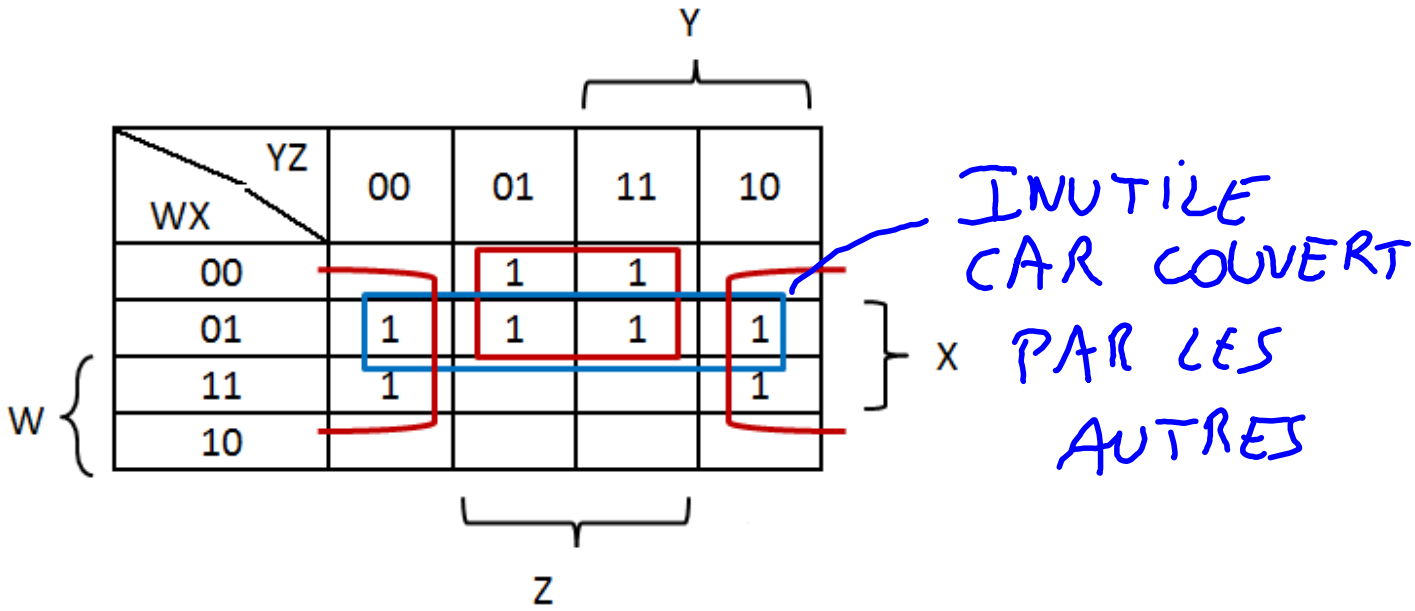
Tentative de simplification systématique

1. Identifier tous les impliquants premiers
2. Sélectionner :
   - 2.1 tous les impliquants premiers essentiels
   - 2.2 un ensemble minimal d'impliquants premiers qui permettent de couvrir tous les mintermes restants de la fonction

- Cette procédure peut donner plusieurs solutions selon les choix opérés au point 2.2
- Possibilité d'obtenir des solutions équivalentes en terme de coût
- La solution optimale n'est pas nécessairement trouvée

Exemple 1

$$F = \sum m(1, 3, 4, 5, 6, 7, 12, 14)$$

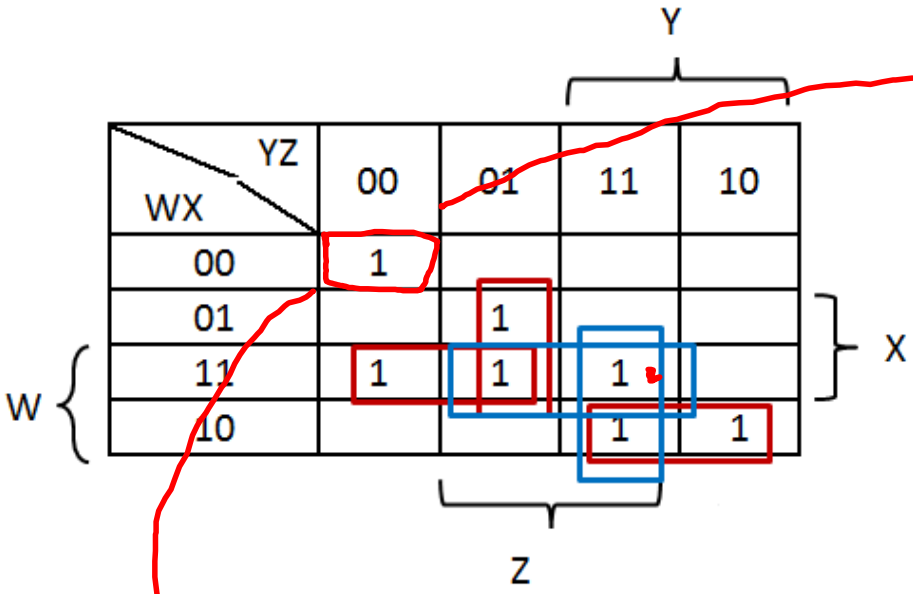


- Les impliquants $(1, 3, 5, 7)$ et $(4, 6, 12, 14)$ sont des impliquants essentiels
- L'impliquant $(4, 5, 7, 6)$ n'est pas essentiel
- Les deux premiers suffisent à couvrir tous les mintermes de la fonction

$$F = \bar{W}Z + X\bar{Z}$$

Exemple 2

$$F = \sum m(0, 5, 10, 11, 12, 13, 15)$$



- Les impliquants (0), $(5, 13)$, $(12, 13)$ et $(10, 11)$ sont des impliquants essentiels

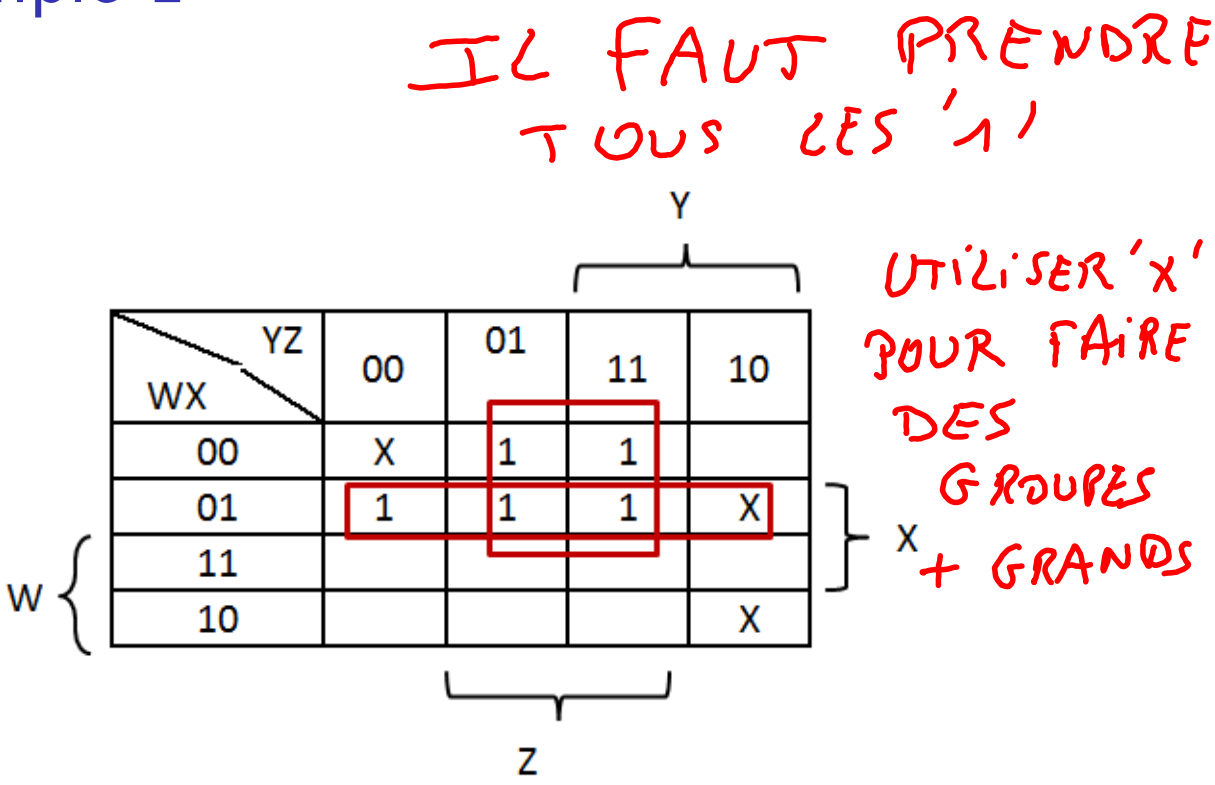
- Les impliquants $(13, 15)$ et $(11, 15)$ ne sont pas pas essentiels
- Il faut sélectionner un de ces deux impliquants pour inclure le minterme m_{15}
- Il existe deux solutions équivalentes

$$F = \bar{W}\bar{X}\bar{Y}\bar{Z} + X\bar{Y}Z + WX\bar{Y} + \begin{cases} WYZ \\ \text{ou} \\ WXZ \end{cases}$$

Fonction non complètement spécifiée - Don't care

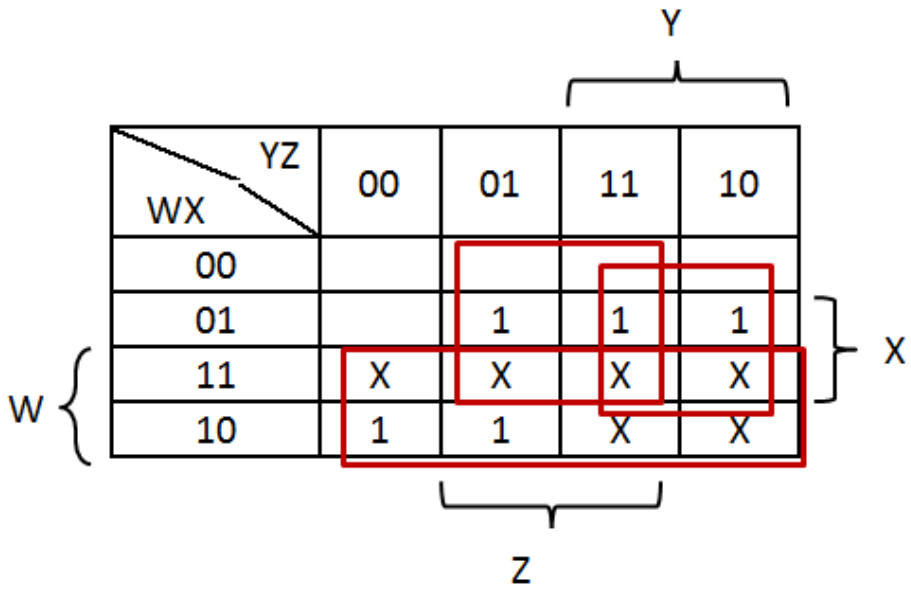
- Dans certaines situations, la fonction booléenne de sortie n'est pas définie ou n'a pas de sens pour certaines combinaisons des entrées
- Situation relativement fréquente en pratique
- Exemple : des signaux d'entrée fournis par des capteurs qui ne sont jamais actifs simultanément
- La valeur affectée à la sortie peut être indifféremment 0 ou 1 = indifférents ou don't care
- Représentés par un "X" dans la table de Karnaugh
- Lors de la simplification de la fonction, le "Don't care" peut être traité comme "0" ou "1" selon la situation la plus avantageuse

Don'care - Exemple 1



- Les "Don't care" sont traités comme des 0
- Nombre d'impliquants plus élévé si traités comme des 1

Don't care - Exemple 2



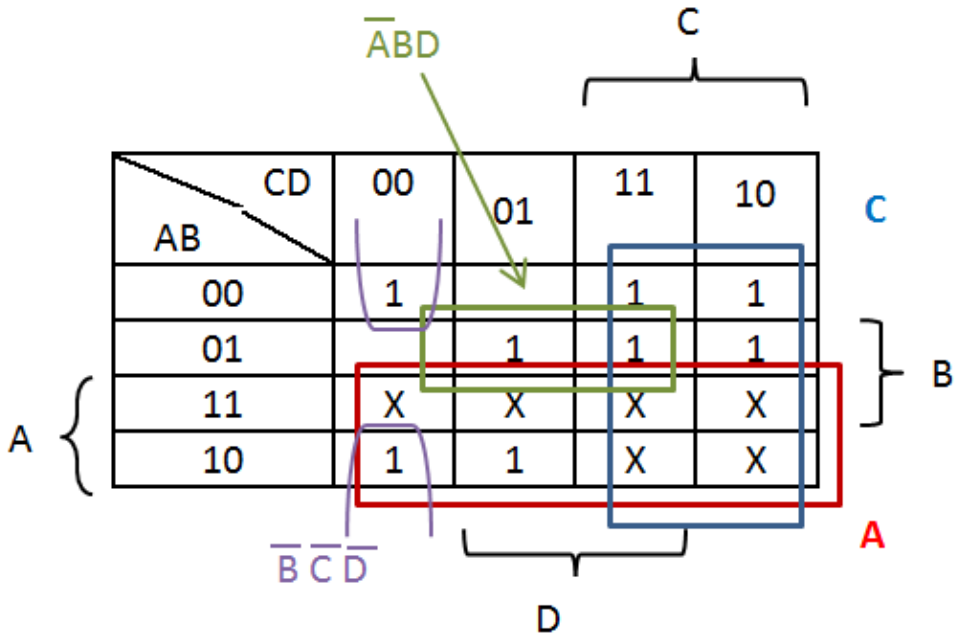
- Il est plus intéressant ici de traiter les "Don't care" comme des 1
- Impliquants de plus grande taille $\Rightarrow$ simplification plus importante de la fonction

Intérêt des "Don't care"

- Fonction a de l'afficheur 7-segments

$$a = \sum m(0, 2, 3, 5, 6, 7, 8, 9)$$

- Il existe 6 "Dont't care" pour les combinaisons des entrées correspondants à 10, 11, 12, 13 ,14 et 15



- On obtient :

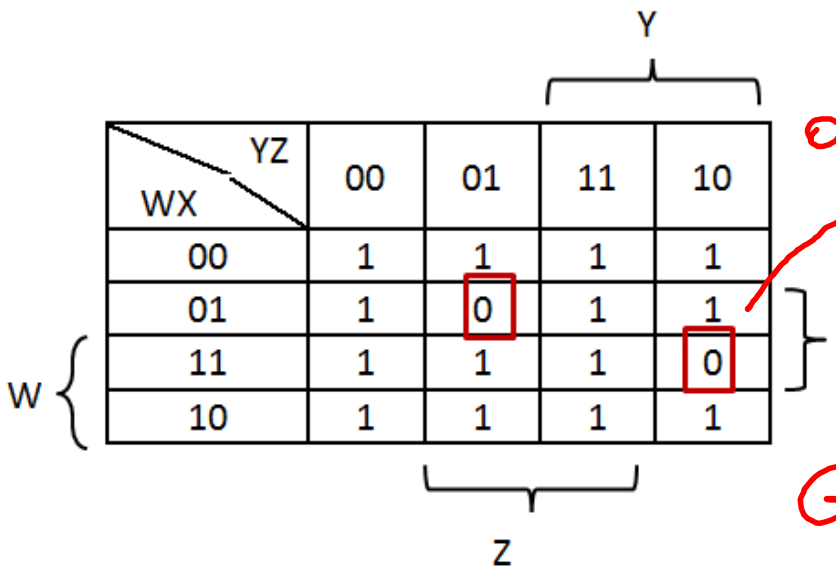
$$a = C + A + \bar{A}BD + \bar{B}\bar{C}\bar{D}$$

- Gate input cost : $G = 10 + 4$ au lieu de 15+4 ! (si ~~0~~ X)

(2x3+4) → 2 + 3x3 + 4

Simplification sous forme d'un produit de sommes

- Il peut être plus intéressant (ou plus aisé) de simplifier la fonction sous la forme d'un produit de Maxtermes (POM)
- Exemple : minorité de 0 dans la table de Karnaugh
- On applique la même procédure sur la fonction complémentée
- L'application de la loi de De Morgan fournit la solution

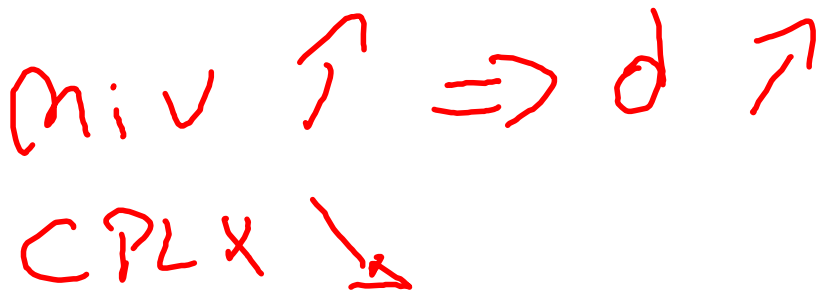


$$\bar{F} = \bar{W}X\bar{Y}Z + WXY\bar{Z} = \sum m$$

$$F = (W+\bar{X}+Y+\bar{Z}).(\bar{W}+\bar{X}+\bar{Y}+Z)$$

Réalisation sous forme de circuits à 2 niveaux

- Les formes simplifiées POS et SOP découlant de la technique des tables de Karnaugh donnent toutes deux lieu à des réalisations sous forme de circuits à deux niveaux de portes logiques
- Il peut être avantageux de considérer d'autres transformations de la fonction
- Cela conduit à des circuits à plus de deux niveaux
- La complexité peut être réduite mais le délai introduit par le circuit est augmenté



Circuits à plus de deux niveaux

LISEZ À VOTRE AISE

1. Factorisation

- On factorise l'expression :

$$\begin{aligned} G &= A\bar{C}E + A\bar{C}F + A\bar{D}E + A\bar{D}F + BCD\bar{E}\bar{F} \\ &= A(\bar{C}E + \bar{C}F + \bar{D}E + \bar{D}F) + BCD\bar{E}\bar{F} \\ &= A(\bar{C} + \bar{D})(E + F) + BCD\bar{E}\bar{F} \end{aligned}$$

- Réduction du gate input cost : de 22+4 à 14+4
- Augmentation du nombre de niveaux de portes : 3 au lieu de 2

2. Décomposition

- Définition de variables intermédiaires

$$\begin{aligned} G &= A\bar{X}_1X_2 + BX_1\bar{X}_2 \\ X_1 &= CD \\ X_2 &= E + F \end{aligned}$$

UTILE POUR PROJET !?

- Réduction du gate input cost : de 14+4 à 12+2

3. Extraction

- Si plusieurs fonctions sont à réaliser, exploiter les termes communs
- Seconde fonction à réaliser

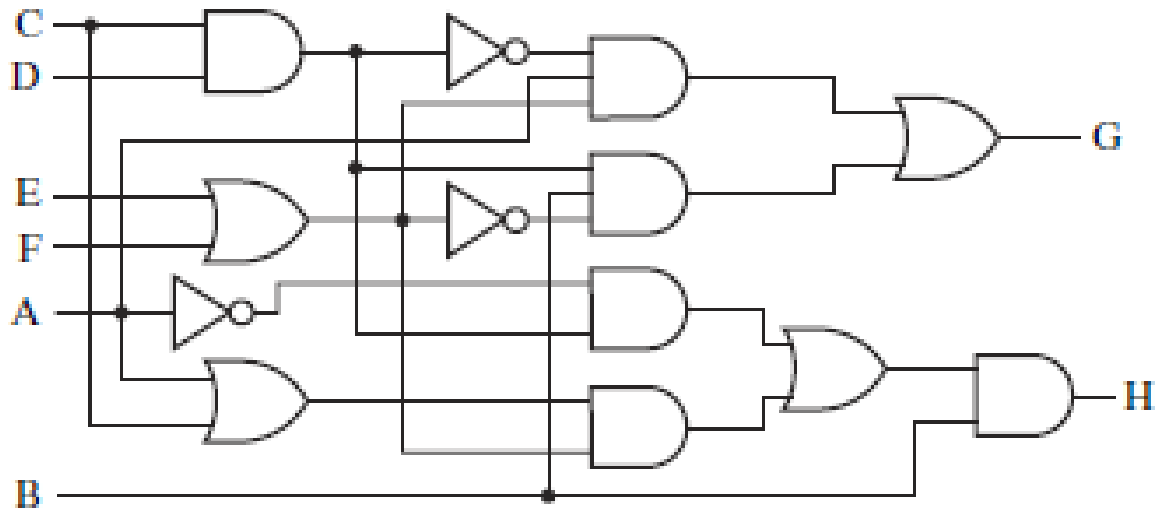
$$H = \bar{A}BCD + ABE + ABF + BCE + BCF$$
$$= B(\bar{A}CD + (A + C)(E + F))$$

- Si l'on considère G et H simultanément :

$$G = A\bar{X}_1X_2 + BX_1\bar{X}_2$$
$$H = B(\bar{A}X_1 + X_3X_2)$$
$$X_1 = CD$$
$$X_2 = E + F$$
$$X_3 = A + C$$

- gate input cost :
  - G seul, forme non factorisée : 22+4
  - H seul, forme non factorisée : 21+1
  - G et H, forme non factorisée : 43+5
  - G et H factorisés avec termes communs : 22+3
- Circuit à 4 niveaux de portes

Extraction - circuit



Pour réduire le nombre de niveaux, soit :

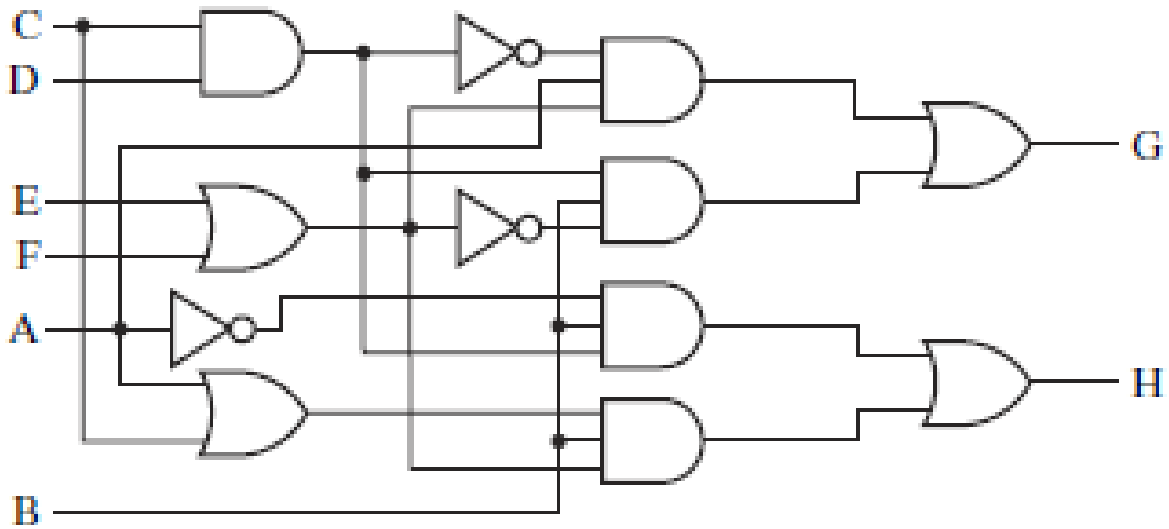
- éliminer une factorisation
- éliminer une variable intermédiaire

Elimination

Le plus intéressant ici est d'éliminer la factorisation de B qui est en fait inutile :

$$G = A\bar{X}_1X_2 + BX_1\bar{X}_2$$
$$H = B\bar{A}X_1 + BX_3X_2$$
$$X_1 = CD$$
$$X_2 = E + F$$
$$X_3 = A + C$$

- gate input cost : 22+ 3, inchangé
- 3 niveaux



Références

- *Logic and Computer Design Fundamentals, 4/E*, M. Morris Mano Charles Kime , Course material
  http ://writphotec.com/mano4/
- *Cours d'électronique numérique*, Aurélie Gensbittel, Bertrand Granado, Université Pierre et Marie Curie
  http ://bertrand.granado.free.fr/Licence/ue201/coursbeameranime.pdf

Terms of Use

- All (or portions) of this material © 2008 by Pearson Education, Inc.
- Permission is given to incorporate this material or adaptations thereof into classroom presentations and handouts to instructors in courses adopting the latest edition of Logic and Computer Design Fundamentals as the course textbook.
- These materials or adaptations thereof are not to be sold or otherwise offered for consideration.
- This Terms of Use slide or page is to be included within the original materials or any adaptations thereof.