Digital Data Processing



M. Combacau combacau@laas.fr

November 6, 2024



Objectif

Simplification des fonctions logiques





Qu'appelle-t-on simplification / optimisation ?

Simplifier ou optimiser fait référence à un critère d'optimisation

- Un trajet peut être optimisé vis à vis du coût global, de la distance à parcourir, du temps de voyage, du confort du voyageur, etc.
- L'optimum est atteint quand le critère est minimisé ou maximisé
- Pour une fonction logique, l'optimum est évalué en fonction d'un critère mesurant la difficulté de réalisation : nombre d'opérateurs + nombre de variales dans l'expression algébrique à réaliser
- Il s'agit de minimiser le critère dans notre cas
- L'optimisation (simplification) était utile quand les circuits étaient synthétisés "à la main". De nos jours, le compilateur VHDL effectue cette optimisation algorithmiquement

Il faut connaître les principes d'optimisation, l'utilisation des outils informatiques prend alors tout son sens





Différentes méthodes de simplification

La méthode de simplification s'appuie sur une représentation.

- Représentation tabulaire : méthode de Karnaugh
- Représentation numérique : méthode de Quine& Mc Clusquey
- Représentation algébrique : méthode des consensus

Dans ce cours, nous nous limitons à la simplification par table de Karnaugh.

La méthode de Quine et Mc Clusquey est parfois qualifiée de méthode "tabulaire", mais elle repose sur une forme d'écriture numérique des points vrais et non spécifiés pour une fonction incomplètement spécifiée, le tableau n'intervenant pas pour la simplification, mais juste pour la présentation des calculs.





Méthode de Karnaugh : complémentarité graphique (1)

La méthode de simplification de Karnaugh repose sur deux principes

- 1 La propriété d'absorption $(x + \overline{x} = 1)$
- 2 L'adjacence de deux combinaisons successives du code Gray

Deux cellules adjacentes (ayant un côté en commun) dans une table de Karnaugh ne diffèrent que par la valeur d'une variable (code Gray)

Ainsi les deux monômes associés à deux cellules adjacentes ne différant que par la variable x; peuvent s'écrire :

$$m.x_i$$
 et $m.\overline{x_i}$

S'il s'agit de points vrais, la forme \(\sum_{\pi}\) formée des mintermes correspondant aux points vrais s'écrit donc

$$f = \ldots + m_i x_i + m_i \overline{x_i} + \ldots = \ldots + m_i (x_i + \overline{x_i}) + \ldots = \ldots + m + \ldots$$





Méthode de Karnaugh : complémentarité graphique (2)

Soit la partie de la table de Karnaugh correspondant à deux point vrais occupant des cellules adjacentes

$$m.x_i + m.\overline{x_i} = m(x_i + \overline{x_i}) = m$$

Pour simplifier algébriquement et graphiquement, on regroupe les quatre points vrais (méthode de Karnaugh \equiv absoption de deux variables)

- regroupement de quatre point vrais adjacents ⇒ absoption de deux variables
- les variables absorbées sont celle qui changent de valeur d'un point à l'autre





Méthode de Karnaugh : complémentarité (1)

Raisonnons sur une fonction à *n* variables $X = (x_{n-1}, \dots, x_0)$

- Faire un regroupement de 2^k combinaisons permet d'éliminer k variables si et seulement si
 - dans les k combinaisons regroupées n-k variables ne changent pas de valeur.
- Le nombre de groupes de n-k composantes (celles qui restent constantes dans le regroupement) parmi n composantes est donné par $\binom{n}{n-k}=\binom{n}{k}$
- Enfin, pour chaque groupe de n-k variables, il existe 2^{n-k} combinaisons (points)

Pour une fonction à *n* variables, il existe donc

$$\binom{n}{k} \times 2^{n-k} = C_k^n \times 2^{n-k} = \frac{n!}{k! \times (n-k)!} \times 2^{n-k}$$

regroupements de 2^k points vrais de la fonction





Méthode de Karnaugh : complémentarité (2)

Raisonnons sur une fonction à *n* variables, $X = (x_{n-1}, \dots, x_0)$, pour des regroupements de 2^k points

		k						
		0	1	2	3	4		
	2	4	4	1	_	_		
1	3	8	12	6	1	_		
	4	16	32	24	8	1		
	5	32	80	40	20	10		

- "A la main", l'utilisation des tables de Karnaugh se limite aux valeurs en gras
- Pour n = 5 ou n = 6 les nombre de regroupements possibles devient prohibitif, seules les méthodes algorithmiques sont viables avec exécution par un ordinateur.
- Le compilateur VHDL fait ce type de regroupement en vue de simplification.





Méthode de Karnaugh : complémentarité graphique (3)

Application : recherche de regroupements de deux points

		<i>x</i> ₁ <i>x</i> ₀					
		00	01	11	10		
	00	0	0	1	0		
<i>X</i> 3 <i>X</i> 2	01	0	1	1	0		
A3A2	11	1	1	0	1		
	10	0	1	1	0		

$$m_1 = x_3.\overline{x_1}.x_0$$

$$m_2 = \overline{x_3}.x_2.x_0$$

$$m_3 = \overline{x_2}.x_1.x_0$$

$$m_4 = x_3.x_2.\overline{x_0}$$

Méthode tabulaire 000000000

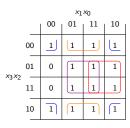
Il existe 32 regroupements différents...





Méthode de Karnaugh : complémentarité graphique (4)

Application : recherche de regroupements de quatre points



		<i>x</i> ₁ <i>x</i> ₀					
		00	ı	01		11	10
	00	1		1		1	1
Ya Ya	01	0		1		0	0
<i>x</i> ₃ <i>x</i> ₂	11	0		1		0	0
	10	0		1		0	0

Méthode tabulaire 00000000

$$m_1 = \overline{x_2}.x_0$$

$$m_2 = x_2.x_0$$

$$m_3 = x_2.x_1$$

$$m_4 = \overline{x_2}.\overline{x_0}$$

$$m_5 = \overline{x_1}.x_0$$

 $m_6 = \overline{x_3}.\overline{x_2}$

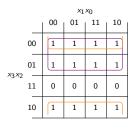
Il existe 24 regroupements différents...





Méthode de Karnaugh : complémentarité graphique (5)

Application : recherche de regroupements de huit points



		<i>x</i> ₁ <i>x</i> ₀					
		00	01	11	10		
	00	0	1	1	0		
<i>x</i> 3 <i>x</i> 2	01	0	1	1	0		
^3^2	11	0	1	1	0		
	10	0	1	1	0		

$$m_1 = \overline{x_2}$$
 $m_2 = \overline{x_3}$

$$m_3 = x_0$$

Il existe 8 regroupements différents...





Implicant d'une Ea, monôme premier et application avec Karnaugh

Notons $m(X_i)$ la valeur du monôme m formé sur X pour la combinaison x_0

$$[m(X_i) \Rightarrow Ea(x_i)] \Leftrightarrow m \text{ est dit } \mathbf{implicant} \text{ de } Ea$$

Un monôme m est dit premier si aucun de ses diviseurs n'est implicant pour Ea Dans un tableau de karnaugh, un diviseur de m correspond à un regroupement contenant tous les points de m Illustration

		x_1x_0					
		00	01	11	10		
	00	1	1	1	1		
<i>X</i> 3 <i>X</i> 2	01	1	1	1	1		
A3A2	11	0	0	0	0		
	10	0	0	0	0		

$$m = \overline{x_3}$$

$$m_1 = \overline{x_3}.\overline{x_1}.\overline{x_0}$$

$$m_2 = \overline{x_3}.x_0$$

$$m_4 = \overline{x_3}.x_2.x_1.\overline{x_0}$$

Méthode tabulaire 000000000

Avec $m + m_i = m \quad \forall i \in [1..4]$





Simplification par la méthode de Karnaugh

Une fonction logique étant donnée sous une forme quelconque :

- 1 La ramener à une représentation dans une table de karnaugh
- 2 Faire tous les regroupements correspondants aux monômes premiers (regroupements non inclus dans un autre)
- 3 L'ensemble des monômes obtenus est appelé la base première complète

Remarque : la forme $\sum \prod$ constitué des monômes de la base première complète est parfois également appelée base première complète (abus de langage)





Simplification par la méthode de Karnaugh : exemples

		<i>x</i> ₁ <i>x</i> ₀					
		00	01	11	10		
<i>x</i> ₂	0	1	1	1	0		
	1	0	0	1	1		
	1_	0	0	1	1		

		<i>x</i> ₁ <i>x</i> ₀					
		00	01	11	10		
<i>x</i> ₃ <i>x</i> ₂	00	1	0	1	1		
	01	0	0	1	0		
	11	0	1	1	0		
	10	1	0	1	1		

Méthode tabulaire 000000000

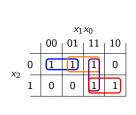
On souhaite déterminer les bases mimimales des deux fonctions représentées dans ces deux tables de karnaugh.

Il faut effectuer tous les regroupements possibles de points vrais non inclus dans un autre regroupement

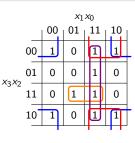




Simplification par la méthode de Karnaugh : exemples







Méthode tabulaire 00000000

Table f_2

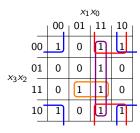
 $BPC(f_1): \{\overline{x_2}.\overline{x_1}, x_1, x_0, \overline{x_2}.x_0, x_2.x_1\}$ $BPC(f_2): \{x_1.x_0, \overline{x_2}.\overline{x_0}, \overline{x_2}.x_1, x_3.x_2.x_0\}$





Simplification par la méthode de Karnaugh

Illustration du besoin (exemple précédent f_2)



$$BPC(f_2): \{x_1.x_0, \overline{x_2}.\overline{x_0}, \overline{x_2}.x_1, x_3.x_2.x_0\}$$

La fonction est correctement représentée par tout ensemble de monômes premiers couvrant tous ses points vrais. Ici, seule une base permet de couvrir tous les points vrais.

$$b_1 = \{x_1.x_0, \overline{x_2}.\overline{x_0}, x_3.x_2.x_0\}$$

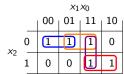
Pas besoin de considérer le coût de cette solution puisqu'elle est unique...





Simplification par la méthode de Karnaugh

Illustration du besoin (exemple précédent f_1)



$$BPC(f_1) = \{\overline{x_2}.\overline{x_1}, x_1, x_0, \overline{x_2}.x_0, x_2.x_1\}$$

Méthode tabulaire 00000000

La fonction est correctement représentée par tout ensemble de monômes premiers couvrant tous ses points vrais. Ici, deux bases permettent de couvrir tous les points vrais

- $b_1 = \{\overline{x_2}.\overline{x_1}, \overline{x_2}.x_0, x_2.x_1\} \Rightarrow f = \overline{x_2}.\overline{x_1} + \overline{x_2}.x_0 + x_2.x_1 \quad \text{coût} = 14$
- $b_2 = \{\overline{x_2}, \overline{x_1}, x_1, x_0, x_2, x_1\} \Rightarrow f = \overline{x_2}, \overline{x_1} + x_1, x_0 + x_2, x_1 \text{ coût} = 13$

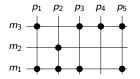
Le calcul des coûts des bases de cette fonction montre que l'optimum est la base b2





Simplification par la méthode de Karnaugh: minimisation

Utilisation d'une table de choix (exemple générique)



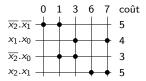
- Le point p₄ est dit essentiel (un seul monôme de couvre)
- Le monôme m_3 qui couvre le point essentiel p_4 est obligatoire : toute base de représentation de la fonction doit le contenir.
- Traitement itératif de la table : suppression des monômes obligatoires et des points qu'ils couvrent \rightarrow nouvelle table à traiter.
- Ici suppression de m_3 et de p_1 , p_3 , p_4 et p_5 . Le choix du monôme pour couvrir le point p_2 est fait en fonction du coût des monômes m_2 et m_1
- Fin de la minimisation quand tous les points sont couverts



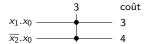


Simplification par la méthode de Karnaugh : minimisation

Utilisation d'une table de choix (exemple de la fonction f_1)



Les points 0 et 6 sont essentiels et rendent les monômes $\overline{x_2}.\overline{x_1}$ et $x_1.x_2$ obligatoires. La table est reconstruite après suppression de ces éléments



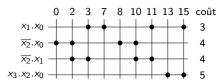
Ce qui ramène bien aux deux solutions déjà évoquées intuitivement. Le choix optimal est celui du monôme de coût minimal. On retrouve la base b2 déterminée intuitivement un peu plus tôt...





Simplification par la méthode de Karnaugh : minimisation

Utilisation d'une table de choix (exemple de la fonction f_2)



Les points 0, 7, et 13 sont essentiels et rendent les monômes $x_1.x_0$ et $\overline{x_2}.x_0$ et $x_3.x_2.x_0$ obligatoires.

Ces trois monômes couvrent tous les points vrais de f_2 et il existe donc une seule base minimale

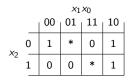
$$b_3 = \{x_1.x_0, \overline{x_2}.x_0, x_3.x_2.x_0\}$$



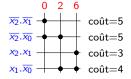


Cas des fonctions incomplètement spécifiées

Couvrir les points vrais de la fonction f_{\min}^* avec les implicants de la BPC de f_{\max}^* **Exemple**



$$\begin{split} f^* &= f_1^*\{0,2,6\}_{10} + f_*^*\{1,7\}_{10} \\ f^*_{min} &= f_1^*\{0,2,6\}_{10} \\ f^*_{max} &= f_1^*\{0,1,2,6,7\}_{10} \\ BPC(f^*_{max}) &= \{\overline{x_2}.\overline{x_1},\overline{x_2}.\overline{x_0},x_2.x_1,x_1.\overline{x_0}\} \end{split}$$



```
\begin{array}{lll} - \  \, \text{solution} \  \, 1 : \left\{ \overline{x_2}.\overline{x_1}, x_1.\overline{x_0} \right\} & \text{coût} = 9 \\ - \  \, \text{solution} \  \, 2 : \left\{ \overline{x_2}.\overline{x_0}, x_2.x_1 \right\} & \text{coût} = 8 \\ - \  \, \text{solution} \  \, 3 : \left\{ \overline{x_2}.\overline{x_0}, x_1.\overline{x_0} \right\} & \text{coût} = 9 \end{array}
```

pas de point essentiel \Rightarrow énumération des solutions et détection de la solution optimale



