

UNIVERSITÉ DE NEUCHÂTEL
INSTITUT DE MICROTECHNIQUE

Application du formalisme
booléen algébrique à la synthèse de
systèmes logiques combinatoires

THÈSE

PRÉSENTÉ À LA FACULTÉ DES SCIENCES
POUR OBTENIR LE GRADE DE DOCTEUR ÈS SCIENCES

PAR

N. Peguiron

IMPRIMATUR POUR LA THÈSE

*Application du formalisme booléen algébrique
à la synthèse de systèmes logiques combina-
toires*.....

de Monsieur *Nicolas Peguiron*.....

UNIVERSITÉ DE NEUCHÂTEL

FACULTÉ DES SCIENCES

La Faculté des sciences de l'Université de Neuchâtel,
sur le rapport des membres du jury,

*MM. les professeurs F. Pellandini, A.V. Shah,
A. Robert, Th. Stutz (EPF-Zurich) et*.....

G. Guggenbühl (EPF-Zurich).....

autorise l'impression de la présente thèse.

Neuchâtel, le *30 août 1983*.....

Le doyen:

A. Aeschlimann

A. Aeschlimann

APPLICATION DU FORMALISME BOOLEEN ALGEBRIQUE A
LA SYNTHÈSE DE SYSTEMES LOGIQUES COMBINATOIRES

Résumé

Un ensemble de moyens a été développé dans le cadre de l'algèbre de Boole pour effectuer la description, à l'aide d'une fonction booléenne, d'une table de vérité, d'une relation d'ordre entre deux fonctions booléennes, ou d'un ensemble de fonctions à traiter simultanément. L'application de ces moyens à la synthèse classique de systèmes combinatoires sur deux niveaux permet la justification théorique complète de ce procédé; les méthodes présentées se prêtent également au développement théorique de procédés de synthèse nouveaux.

I. INTRODUCTION

1. Définitions

Un système logique combinatoire est un bloc fonctionnel sans mémoire interne admettant $|N|$ signaux binaires d'entrée et produisant $|J|$ signaux binaires de sortie; une table de vérité définit le fonctionnement que l'on attend de ce système, en spécifiant les valeurs que doivent prendre les signaux de sortie pour un certain nombre $|I|$ de configurations distinctes des signaux d'entrée (fig. 1.1).

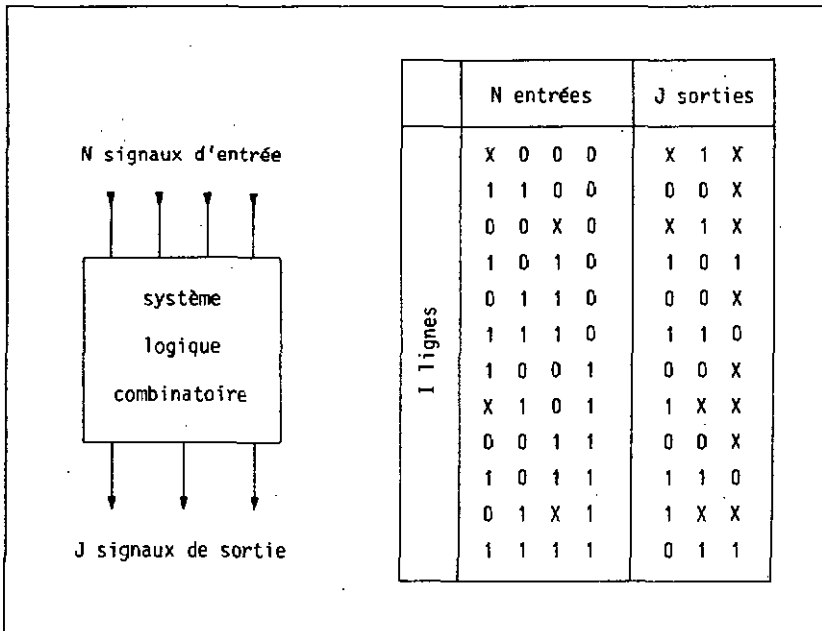


Fig. 1.1 : système logique combinatoire et table de vérité.

Lorsqu'il existe une ou des configurations des signaux d'entrée qui ne sont pas spécifiées dans la table de vérité, ces configurations constituent des états indéfinis implicites du système; lorsque pour l'une ou l'autre des configurations d'entrée mentionnées dans la table de vérité, l'un des signaux de sortie n'a pas de valeur assignée ("X", ou "don't care"), on parle alors d'une indéfinition explicite. Les procédés de synthèse tirent parti de la réalisation indifférente des états indéfinis implicites et explicites pour obtenir une simplification accrue des systèmes à optimiser.

Un "X" (ou "don't care") qui apparaît dans la partie d'entrée de la table de vérité signifie que plusieurs configurations d'entrée sont résumées en une seule ligne; dans ce cas, la table doit remplir une condition dite de cohérence, qui impose un état de sortie identique pour toutes les lignes de la table dont les états d'entrée ont une intersection non nulle. Lorsque la table de vérité ne contient pas de "X" dans sa partie d'entrée, elle est dite canonique; une table canonique est naturellement cohérente.

La table de vérité est complètement définie lorsque toutes les configurations possibles des signaux d'entrée y apparaissent, et que, pour chacune d'elles, tous les signaux de sortie ont une valeur assignée.

2. Notion de synthèse

Les systèmes logiques combinatoires sont réalisés au moyen d'opérateurs booléens élémentaires (AND, OR, NOT), ou de blocs de base réunissant un ou plusieurs opérateurs élémentaires (NAND, NOR, EX-OR, Majority, Threshold, etc.).

La méthode de réalisation systématique la plus immédiate fait intervenir les deux opérateurs de base AND et OR disposés en deux niveaux logiques (si l'on ne dispose pas des variables d'entrée niées, un troisième niveau composé d'inverseurs est nécessaire pour les engendrer); le premier niveau isole les diverses configurations des signaux d'entrée spécifiées dans la table de vérité; le second niveau synthétise chaque signal de sortie en réunissant les configurations d'entrée pour lesquelles ce signal doit être actif.

Cependant, il existe souvent plusieurs réalisations sur deux niveaux d'un système donné, même dans le cas d'une table complètement définie; ces réalisations sont identiques du point de vue de leur fonctionnement statique, mais diffèrent quant au nombre d'opérateurs de base qui les composent. Les réalisateurs de systèmes logiques combinatoires ont alors cherché un moyen d'obtenir de façon systématique l'implantation la moins coûteuse des tables de vérité.

Bien que le critère de coût d'une réalisation varie d'une technologie à l'autre, il est toujours lié au nombre d'opérateurs nécessaires, que ce soit directement (relais, portes) ou indirectement, à cause de la place requise et de la consommation (VLSI).

Le profil des problèmes soumis à la synthèse s'est modifié au cours du temps : à l'époque où les opérateurs de base étaient réalisés avec des composants discrets (relais, tubes, diodes, composants logiques SSI, etc.), les systèmes à optimiser étaient en général de faible taille, et leurs tables étaient fortement définies; la synthèse automatique de systèmes logiques combinatoires était alors source d'économies appréciables.

Plus tard, les mémoires mortes sont apparues sur le marché, avec des capacités toujours plus vastes et des coûts toujours moindres; les systèmes logiques fortement définis, eux aussi de taille grandissante, ont

pu être avantageusement implantés sur de tels composants, sans nécessiter de synthèse.

Parallèlement à cette évolution, le développement de l'intégration de circuits à une échelle toujours plus grande (LSI, puis VLSI) a fait apparaître la notion de PLA ("programmed logic array"), permettant l'implantation en une géométrie régulière de systèmes logiques combinatoires. Ainsi la synthèse automatique de ces circuits a conquis actuellement sa place dans la conception assistée par ordinateur (CAO) de circuits binaires intégrés à grande échelle, dans le cadre de laquelle elle effectue la synthèse en PLA de blocs fonctionnels réunissant de la logique combinatoire; les problèmes à optimiser dans ce contexte ont en général un nombre de variables élevé et sont faiblement définis; c'est alors la surface occupée par la PLA dans le circuit qui constitue un critère de minimalisation, plutôt que le nombre des éléments qui la composent.

La synthèse automatique de systèmes logiques redevient donc d'actualité pour la réalisation de PLA simplifiées, destinées à occuper une surface minimale dans les circuits intégrés dans lesquels elles doivent être implantées.

3. Procédés de synthèse

Les méthodes de synthèse peuvent être classées selon la structure des résultats qu'elles produisent : ainsi, on distingue les méthodes de synthèse sur deux niveaux d'opérateurs (AND-OR, etc.) et leurs essais de généralisation [40] - [44], la synthèse d'arbres de décision binaires, qui donne lieu aux réalisations en multiplexeurs ou en décodeurs [45] - [49], ainsi que les autres approches de synthèse multi-niveaux [50] - [53] et les méthodes utilisant les propriétés d'une technologie donnée (en particulier la technologie C-MOS) [54].

La plupart des méthodes de synthèse sur deux niveaux procèdent en deux phases : 1° calcul de tous les impliquants premiers ou communs d'un système, 2° extraction d'un sous-ensemble irrédondant minimal de ceux-ci; comme ces méthodes sont de loin celles qui ont fait l'objet du plus grand nombre de contributions dans le domaine de la synthèse de systèmes logiques combinatoires, il est légitime de les appeler méthodes classiques de synthèse.

Il existe d'autres méthodes de synthèse sur deux niveaux qui procèdent en une seule phase, en construisant les impliquants premiers ou communs nécessaires autour des éléments de la donnée, jusqu'à la réalisation complète du système [37] - [39]; ces méthodes sont heuristiques par nature, alors que seule la dernière phase des méthodes classiques (résolution du tableau de couverture) est en général effectuée de façon heuristique.

Citons également la méthode graphique de M. Karnaugh [18], qui est encore universellement employée de nos jours pour la réalisation à la main de très petits systèmes; strictement appliquée, c'est une méthode exhaustive (classique), mais elle est souvent utilisée à la manière des méthodes heuristiques.

L'utilisation de l'algèbre de Boole pour la description des systèmes logiques combinatoires a déjà été proposée par C. E. Shannon [6] en 1938; la représentation d'un système à sortie multiple au moyen d'une fonction booléenne unique est due à D. E. Muller [7], et a été reprise par quelques auteurs [8] - [11]; une dérivation formelle de cette fonction unique caractéristique d'un système à sortie multiple est présentée dans le chapitre III ci-dessous.

Toutefois, la plupart des contributions qui traitent de la synthèse simultanée de systèmes à sortie multiple utilisent un équivalent numérique de la représentation de Muller : les tableaux de connexion (en anglais : "connexion arrays") [2], [14] - [17], [34], qui sont au nombre

de trois : le tableau de connexion directe ("ON-array"), le tableau de non-connexion ("OFF-array") et le tableau de connexion indifférente ("DC-array"), l'un d'entre eux pouvant toujours être déduit des deux autres [2].

Les méthodes de synthèse classique de systèmes logiques ont toujours été développées et présentées sous leur aspect algorithmique : elles décrivent un ensemble de procédés qui fournissent le résultat voulu par leur application systématique (itérative ou récursive) sur les données à disposition.

Ainsi l'opération de consensus, définie par A. Blake [5] et généralisée par J. P. Roth [14], appliquée de façon systématique sur les tableaux réunis de connexion directe et indifférente, permet d'obtenir tous les impliquants premiers ou communs d'un système donné [2], [8] - [10], [19] - [22].

Une autre méthode, le découpage (en anglais "sharp"), aboutit au même résultat en retirant de l'impliquant universel tous les éléments du tableau de non-connexion [2], [16], [31], [32], [34], [55].

Une fois l'ensemble des impliquants premiers ou communs obtenus, les méthodes classiques procèdent à la recherche d'un sous-ensemble irrédondant de ceux-ci correspondant à une réalisation de coût minimal; cette opération est effectuée par la formation d'un tableau de couverture [12], [13], [23], suivie de sa résolution; certains auteurs se sont consacrés au développement d'algorithmes efficaces pour la réduction de ces tableaux [27] - [29]. La recherche d'un sous-ensemble irrédondant d'impliquants peut également être effectuée par un calcul de consensus [11], [12], [25], [26].

4. But du présent travail

Cette contribution a pour but de montrer que les méthodes de synthèse, présentées ordinairement sous forme algorithmique, peuvent également être décrites à l'aide d'opérateurs algébriques.

A cette fin, un ensemble d'outils issus de l'algèbre de Boole a été développé dans un cadre général, c'est-à-dire hors du contexte spécifique de la synthèse de systèmes logiques (chap. II et IV ci-dessous); l'application de ces outils au problème de la synthèse classique permet de retrouver les algorithmes existants en tant que cas particuliers (chap. V). Le caractère général des procédés algébriques développés ici permet de les utiliser à d'autres fins; en guise d'illustration, leur application au développement de deux nouvelles méthodes de synthèse est mentionnée au chapitre VI.

Le formalisme algébrique n'apporte pas d'amélioration de nature algorithmique aux procédés classiques de synthèse en ce qui concerne la taille des problèmes que l'on peut traiter et la rapidité de ce calcul, car de nombreux travaux ont déjà donné un grand développement à cet aspect de la question; par contre, il permet d'une part de clarifier, dans un but didactique, le mécanisme qui est à la base de ces méthodes, et d'autre part de fournir des outils théoriques utilisables pour de nouvelles investigations dans le domaine de la synthèse.

Le formalisme algébrique trouve également une application dans une méthode de représentation numérique des expressions booléennes qui conserve la forme littérale de celles-ci (cf. app. C), ce qui autorise leur traitement par ordinateur tout en préservant la généralité inhérente au caractère algébrique de ces expressions.

11. DESCRIPTION ET CALCUL D'UNE RELATION D'ORDRE

Les variables et les fonctions que nous utiliserons dans le cadre du présent travail appartiennent à l'algèbre de Boole et obéissent à ses lois. Les conventions de notation utilisées sont présentées dans l'appendice A; l'appendice B mentionne les définitions de base et les axiomes de l'algèbre de Boole, ainsi que quelques propriétés générales supposées bien connues.

1. Définitions

Soient deux fonctions booléennes $A(\underline{x})$ et $B(\underline{x})$ dont les arguments communs sont les variables booléennes indépendantes $\underline{x} = \{x_i | i \in I\}$; nous nous intéressons à l'existence d'une relation d'ordre $\mathcal{R}\{A(\underline{x}), B(\underline{x})\}$ entre ces deux fonctions.

Il est possible de formuler une fonction booléenne $R(\underline{x})$ dite caractéristique de la relation \mathcal{R} , dont les arguments sont les mêmes variables \underline{x} ; la fonction $R(\underline{x})$ est définie comme suit :

Définition 2.1

$$R(\underline{x}) = 1 \iff \mathcal{R}\{A(\underline{x}), B(\underline{x})\} \text{ est vérifiée}$$

pour toute valeur \underline{x} que l'on peut attribuer aux variables \underline{x} .

Exemples :

a) cas de l'égalité entre $A(\underline{x})$ et $B(\underline{x})$; on obtient la fonction $R(\underline{x})$ en utilisant un théorème de base de l'algèbre de Boole (cf. app. B, § 4, propriétés n° 34 et 29) :

$$\mathcal{R}\{A(\underline{x}) = B(\underline{x})\} \iff R(\underline{x}) = \overline{A(\underline{x})} \cdot \overline{B(\underline{x})} + A(\underline{x}) \cdot B(\underline{x})$$

b) cas de l'implication :

$$\mathcal{R}\{A(\underline{x}) \leq B(\underline{x})\} \iff R(\underline{x}) = \overline{A(\underline{x})} + B(\underline{x})$$

Exprimons la fonction $R(\underline{x})$ sous forme disjonctive :

$$R(\underline{x}) = \sum_{k \in K} P_k(\underline{x})$$

chaque impliquant $P_k(\underline{x}) \leq R(\underline{x})$ décrit un ensemble suffisant de conditions que les variables \underline{x} doivent remplir pour que la relation \mathcal{R} entre les deux fonctions booléennes $A(\underline{x})$ et $B(\underline{x})$ soit vérifiée :

$$P_k(\underline{x}) = 1 \iff \mathcal{R} \text{ est vérifiée.}$$

Exemple : $\underline{x} = \{x_1, x_2, x_3, x_4\}$, $R(\underline{x}) = x_1 \overline{x_4} + x_2 \overline{x_3} x_4$

l'impliquant $x_1 \overline{x_4} \leq R(\underline{x})$ signifie que la relation d'ordre entre $A(\underline{x})$ et $B(\underline{x})$ décrite par $R(\underline{x})$ est satisfaite pour tout vecteur constant $\underline{\xi} = \{\xi_1, \xi_2, \xi_3, \xi_4\}$ répondant aux conditions $\xi_1 = 1$ et $\xi_4 = 0$. L'ensemble de ces deux conditions est suffisant; cependant, il n'est pas nécessaire, dans la mesure où l'impliquant de $R(\underline{x})$ qui le décrit n'est pas le seul (comme c'est le cas dans cet exemple).

Un impliquant premier de $R(\underline{x})$ décrit un ensemble irrédondant de conditions suffisant à la vérification de la relation; ceci signifie qu'aucun sous-ensemble strict de conditions ne conserve la propriété d'être suffisant.

Chaque impliquant de $R(\underline{x})$ décrit une partie du domaine de validité de la relation parmi l'ensemble des valeurs $\underline{\xi}$ que l'on peut attribuer aux variables \underline{x} ; la fonction $R(\underline{x})$ décrit le domaine entier.

Dans le cas où $R(\underline{x})$ est identique à 1, le domaine de validité de la relation s'étend à toutes les valeurs $\underline{\xi}$ que peuvent prendre les variables \underline{x} . Ainsi, il est possible de démontrer l'existence d'une relation d'ordre en formulant sa fonction caractéristique et en vérifiant son identité à 1.

Exemple : soit à démontrer l'identité :

$$ab + \bar{a}c = ab + \bar{a}c + bc$$

on forme la fonction caractéristique de l'égalité :

$$\begin{aligned} R(a,b,c) &= \overline{(ab + \bar{a}c)} \cdot \overline{(ab + \bar{a}c + bc)} + \\ &\quad + (ab + \bar{a}c) \cdot (ab + \bar{a}c + bc) \\ &= (\bar{a} + \bar{b}) \cdot (a + \bar{c}) \cdot (\bar{a} + \bar{b}) \cdot (a + \bar{c}) \cdot (\bar{b} + \bar{c}) + \\ &\quad + (ab + \bar{a}c) \cdot (ab + \bar{a}c + bc) \\ &= \bar{a}\bar{c} + a\bar{b} + \bar{b}\bar{c} + ab + \bar{a}c \\ &= \bar{a} \cdot (\bar{c} + c) + a \cdot (\bar{b} + b) + \dots = 1 \quad \text{cqfd.} \end{aligned}$$

Remarque : pour éviter le calcul d'une expression compliquée, on remplace souvent la démonstration d'une égalité par celle de l'implication appliquée successivement dans les deux sens :

$$\begin{aligned}
 \mathcal{R}\{A(\underline{x}) = B(\underline{x})\} &\iff R(\underline{x}) = \overline{A(\underline{x})} \cdot \overline{B(\underline{x})} + A(\underline{x}) \cdot B(\underline{x}) \\
 &= [\overline{A(\underline{x})} + B(\underline{x})] \cdot [A(\underline{x}) + \overline{B(\underline{x})}] \\
 &= R^{\leq}(\underline{x}) \cdot R^{\geq}(\underline{x})
 \end{aligned}$$

2. Exemples d'utilisation

Citons deux utilisations courantes de la fonction caractéristique d'une relation d'ordre : l'extraction d'une forme disjonctive irrédundante de la forme complète d'une fonction booléenne, et la démonstration de théorèmes.

Recherche d'une forme irrédundante

Soit une fonction booléenne $F(\underline{x})$ exprimée sous sa forme complète (forme disjonctive comprenant tous les impliquants premiers, et seulement les impliquants premiers; cf. app. B, § 5) :

$$F(\underline{x}) = \sum_{k \in K} p_k(\underline{x})$$

Problème : trouver un sous-ensemble irrédundant K' d'impliquants, c'est-à-dire un sous-ensemble K' d'impliquants qui suffit à exprimer la fonction, mais tel qu'aucun sous-ensemble strict de K' ne conserve cette propriété de suffisance :

$$F(\underline{x}) = \sum_{k \in K'} p_k(\underline{x}) \quad \text{et} \quad \nexists K'' \subsetneq K' \quad \text{tq.} \quad F(\underline{x}) = \sum_{k \in K''} p_k(\underline{x})$$

M. J. Ghazala [12] a présenté une méthode récursive moitié algébrique et moitié algorithmique pour résoudre ce problème; un procédé entièrement algébrique peut être proposé sur la base du calcul d'une relation d'ordre : à cet effet, on formule un modèle paramétré de l'expression cherchée, en faisant intervenir un nouveau jeu de variables booléennes indépendantes $\underline{y} = \{y_k | k \in K\}$:

$$MF(\underline{x}, \underline{y}) = \sum_{k \in K} P_k(\underline{x}) \cdot y_k$$

Pour chaque valeur constante \underline{v} que l'on attribue aux variables \underline{y} , le modèle $MF(\underline{x}, \underline{v})$ devient une somme d'un sous-ensemble particulier d'impliquants :

$$MF(\underline{x}, \underline{v}) = \sum_{k \in K'} P_k(\underline{x}) \quad \text{avec} \quad K' = \{k | v_k = 1\} \subseteq K$$

On impose à ce modèle d'être égal à la fonction donnée; dans ce cas, par construction, une seule implication est à vérifier :

$$\mathcal{R}(F(\underline{x}) \leq MF(\underline{x}, \underline{y})) \iff R(\underline{x}, \underline{y}) = \overline{F(\underline{x})} + MF(\underline{x}, \underline{y})$$

Exprimée sous forme complète, la fonction $R(\underline{x}, \underline{y})$ contient, parmi d'autres, des impliquants qui ne sont formés que des variables \underline{y} (leur existence est justifiée par la propriété $R(\underline{x}, \underline{1}) = 1$; par construction, les variables y_k qui les composent ne sont pas niées); chacun d'eux décrit une expression disjonctive de $F(\underline{x})$:

Théorème 2.2

$$\prod_{k \in K'} y_k \leq R(\underline{x}, \underline{y}) \iff F(\underline{x}) = \sum_{k \in K'} P_k(\underline{x})$$

Démonstration : a) implication dans le sens \Rightarrow :

$$\text{Hyp. : } \prod_{k \in K'} y_k \leq R(\underline{x}, \underline{y}) = \overline{F(\underline{x})} + \sum_{k \in K} p_k(\underline{x}) \cdot y_k$$

$$\text{Posons : } u_k = 1, \quad k \in K' \quad \text{et} \quad u_k = 0, \quad k \in \bar{K}'$$

$$\Rightarrow \quad \overline{F(\underline{x})} + \sum_{k \in K'} p_k(\underline{x}) = 1$$

$$\Rightarrow \quad F(\underline{x}) \leq \sum_{k \in K'} p_k(\underline{x})$$

$$\Rightarrow \quad F(\underline{x}) = \sum_{k \in K'} p_k(\underline{x}) \quad \text{puisque par construction :}$$

$$F(\underline{x}) \geq \sum_{k \in K'} p_k(\underline{x})$$

b) implication dans le sens \Leftarrow : voir ci-dessous "démonstration de théorèmes".

Si l'impliquant $\prod y_k \leq R(\underline{x}, \underline{y})$ est premier, le sous-ensemble K' d'impliquants qu'il décrit est irrédondant, car dans le cas contraire, il existe un autre impliquant $\prod y_k$ incluant le premier, donc composé d'un sous-ensemble de ses variables et décrivant par conséquent un sous-ensemble $K'' \subset K'$ d'impliquants $p_k(\underline{x})$ suffisant à la réalisation de $F(\underline{x})$.

Démonstration de théorèmes

Soient deux propositions \mathcal{G} et \mathcal{H} dont il faut démontrer l'implication $\mathcal{G} \Rightarrow \mathcal{H}$; s'il est possible de former deux fonctions

booléennes $G(\underline{x})$ et $H(\underline{x})$ vérifiant respectivement :

$$G(\underline{x}) = 1 \iff \text{la proposition } \mathcal{G} \text{ est vérifiée}$$

$$H(\underline{x}) = 1 \iff \text{la proposition } \mathcal{H} \text{ est vérifiée}$$

la démonstration voulue se réduit alors à celle de l'implication booléenne $G(\underline{x}) \leq H(\underline{x})$; dans ce but, on formule la fonction caractéristique $W(\underline{x})$ de cette implication, et on vérifie son identité à 1.

Exemple : démonstration de l'implication b) du théorème 2.2; soient les deux propositions :

$$\mathcal{G} : \sum_{k \in K'} P_k(\underline{x}) \quad \text{est une expression disjonctive de } F(\underline{x})$$

$$\mathcal{H} : \text{le terme } \prod_{k \in K'} y_k \text{ est un impliquant de } R(\underline{x}, \underline{y})$$

On établit les correspondances :

$$\overline{F(\underline{x})} + \sum_{k \in K'} P_k(\underline{x}) = 1 \iff \mathcal{G} \text{ est vérifiée}$$

$$\overline{\prod_{k \in K'} y_k} + R(\underline{x}, \underline{y}) = 1 \iff \mathcal{H} \text{ est vérifiée}$$

La fonction caractéristique $\overline{F(\underline{x})} + \sum P_k(\underline{x})$ est suffisante dans ce cas pour décrire l'égalité $F(\underline{x}) = \sum P_k(\underline{x})$, car par construction $F(\underline{x}) \geq \sum P_k(\underline{x})$. On calcule alors l'expression $W(\underline{x}, \underline{y}) = \overline{G(\underline{x})} + H(\underline{x}, \underline{y})$:

$$\begin{aligned}
 W(\underline{x}, \underline{y}) &= F(\underline{x}) \cdot \prod_{k \in K'} \overline{p_k(\underline{x})} + \sum_{k \in K'} \overline{y_k} + \overline{F(\underline{x})} + \sum_{k \in K} p_k(\underline{x}) \cdot y_k \\
 &= \prod_{k \in K'} \overline{p_k(\underline{x})} + \dots + \sum_{k \in K'} p_k(\underline{x}) = 1 \quad \text{cqfd.}
 \end{aligned}$$

L'identité $W(\underline{x}, \underline{y}) = 1$ démontre l'implication $G(\underline{x}) \leq H(\underline{x}, \underline{y})$ et, par le jeu des équivalences $G(\underline{x}) \iff \mathcal{G}$ et $H(\underline{x}, \underline{y}) \iff \mathcal{H}$, l'implication $\mathcal{G} \Rightarrow \mathcal{H}$.

3. Application à la synthèse d'un système simple

Un système logique à sortie unique dont la table de vérité est incomplètement définie est déterminé par deux fonctions booléennes : la borne inférieure de ce système et sa borne supérieure (fig. 2.1).

La borne inférieure $A(\underline{a})$ du système est la fonction booléenne qui réunit toutes les configurations des variables d'entrée pour lesquelles le signal de sortie est assigné à 1.

La fonction $B(\underline{a})$, formée par la réunion des configurations des variables d'entrée pour lesquelles le signal de sortie est assigné à 0, est l'inverse de la borne supérieure.

Toute fonction booléenne $F(\underline{a})$ satisfaisant la double implication $A(\underline{a}) \leq F(\underline{a}) \leq \overline{B(\underline{a})}$ est solution du système à synthétiser [1], [2], [55]; le but consiste à rechercher une fonction composée d'un minimum d'opérateurs; lors d'une réalisation sur deux niveaux, une telle fonction est composée d'un nombre minimal d'impliquants premiers (sauf dans quelques cas très particuliers [36]).

La méthode de synthèse classique fait intervenir la recherche de tous les impliquants premiers du système, suivie du calcul de tous les

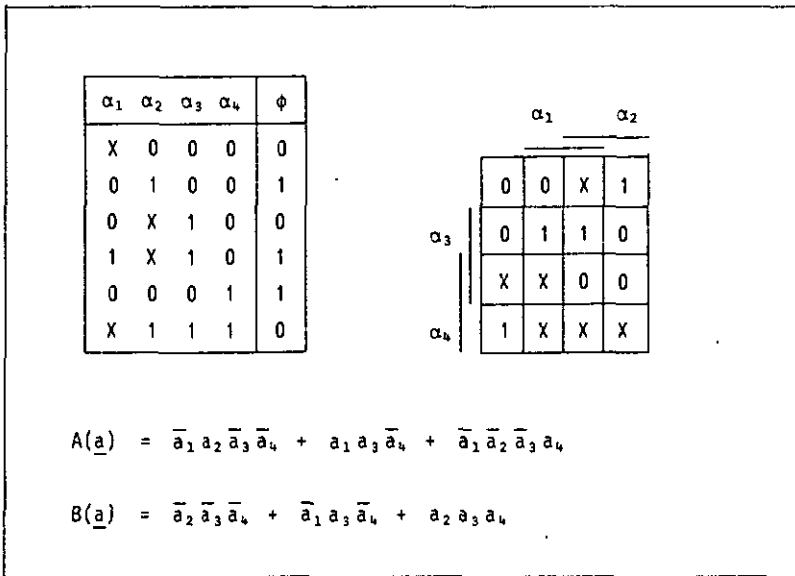


Fig. 2.1 : table de vérité, représentation graphique de Karnaugh et fonctions descriptives d'un système à synthétiser.

sous-ensembles irrédondants de ceux-ci suffisant à la couverture de sa borne inférieure $A(\underline{a})$, pour ne retenir finalement que celui d'entre eux dont le cardinal est le plus petit (ou l'un de ceux-ci lorsqu'ils sont plusieurs à posséder le même cardinal minimal).

Recherche des impliquants premiers du système

Les impliquants premiers du système sont les produits du plus petit nombre de variables \underline{a} , niées ou non niées, dont l'intersection avec la fonction $B(\underline{a})$ est nulle.

Pour les calculer, on forme un modèle $MF^B(\underline{a})$ comprenant tous les impliquants possibles; la somme de tous ces impliquants est absorbée

par le plus grand d'entre eux, 1 : $MF^B(\underline{a}) = 1$. Ce modèle est affiné en interdisant son intersection avec la fonction $B(\underline{a})$ qui rassemble les zéros du système :

$$\mathcal{R}^B_{\{MF^B(\underline{a}), B(\underline{a})\}} : MF^B(\underline{a}) \cdot B(\underline{a}) = 0$$

$$\iff R^B(\underline{a}) = \overline{MF^B(\underline{a}) \cdot B(\underline{a})} = \overline{B(\underline{a})}$$

Nous disposons de $B(\underline{a})$ sous forme disjonctive (cf. chap. III); on exprime alors son inverse sous forme conjonctive en appliquant la règle de Morgan; en effectuant les produits, on obtient alors, après élimination des termes inclus, la forme complète de $R^B(\underline{a})$ (cf. app. B, corollaire B.3).

$$B(\underline{a}) = \bar{a}_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_3 \bar{a}_4 + a_2 a_3 a_4$$

$$R^B(\underline{a}) = \overline{B(\underline{a})} = (a_2 + a_3 + a_4) \cdot (\bar{a}_1 + \bar{a}_3 + a_4) \cdot (\bar{a}_2 + \bar{a}_3 + \bar{a}_4)$$

$$= \bar{a}_2 \bar{a}_3 + \bar{a}_2 a_4 + \bar{a}_3 a_4 + a_1 \bar{a}_2 \bar{a}_4 + a_1 \bar{a}_2 a_3 + a_1 a_3 \bar{a}_4$$

L'expression générale de la fonction caractéristique $R^B(\underline{a})$ que nous avons obtenue en inversant la fonction $B(\underline{a})$ est la suivante :

$$R^B(\underline{a}) = \sum_{k \in K} P_k(\underline{a})$$

Chaque impliquant premier $P_k(\underline{a})$ est un produit d'un nombre minimal de variables \underline{a} , niées ou non, satisfaisant la condition imposée :

$$P_k(\underline{a}) \cdot B(\underline{a}) = 0$$

Les impliquants premiers de $R^B(\underline{a})$ sont donc les impliquants premiers du système, qui sont aussi ceux de sa borne supérieure $\overline{B(\underline{a})}$.

Recherche d'une couverture minimale

La recherche d'une couverture minimale du système se fait par le calcul de toutes les couvertures irrédondantes de la borne inférieure $A(\underline{a})$, dont on ne retient que l'une de celles qui sont composées d'un nombre minimal d'impliquants ou, dans le cas d'une synthèse avec coûts, l'une de celles qui donnent lieu à une réalisation de coût minimal.

Une couverture irrédondante est un sous-ensemble d'impliquants premiers du système dont la somme couvre la borne inférieure $A(\underline{a})$ de celui-ci, et dont chaque terme est nécessaire pour garantir cette propriété de couverture (dans ce contexte, "couvrir" signifie "être impliqué par") :

$$A(\underline{a}) \leq \sum_{k \in K'} P_k(\underline{a}) \quad \text{et} \quad \bigcap_{k \in K''} K'' \subset K' \quad \text{tq.} \quad A(\underline{a}) \leq \sum_{k \in K''} P_k(\underline{a})$$

A nouveau, la façon de procéder consiste à formuler un modèle paramétré de couverture $MF^A(\underline{a}, \underline{y})$ comprenant les impliquants premiers obtenus et un nouveau jeu de variables booléennes indépendantes

$\underline{y} = \{y_k | k \in K\}$:

$$MF^A(\underline{a}, \underline{y}) = \sum_{k \in K} P_k(\underline{y}) \cdot y_k$$

$$MF^A(\underline{a}, \underline{y}) = a_2 \bar{a}_3 y_1 + \bar{a}_2 a_4 y_2 + \bar{a}_3 a_4 y_3 + a_1 a_2 \bar{a}_4 y_4 + a_1 \bar{a}_2 a_3 y_5 + a_1 a_3 \bar{a}_4 y_6$$

On impose à ce modèle de couvrir la borne inférieure $A(\underline{a})$ du système :

$$\mathcal{P}^A\{A(\underline{a}), MF^A(\underline{a}, \underline{y})\} : \quad A(\underline{a}) \leq MF^A(\underline{a}, \underline{y})$$

$$\iff R^A(\underline{a}, \underline{y}) = \overline{A(\underline{a})} + MF^A(\underline{a}, \underline{y})$$

Exprimée sous forme complète, la fonction caractéristique $R^A(\underline{a}, \underline{y})$ fait apparaître, entre autres, des impliquants qui ne sont composés que des variables \underline{y} (leur existence est due à la propriété $R^A(\underline{a}, \underline{1}) = 1$, liée à la condition $A(\underline{a}) \leq \overline{B(\underline{a})}$; par construction, les variables y_k qui composent ces impliquants ne sont pas niées); chacun d'eux décrit une couverture de la borne inférieure :

Théorème 2.3

$$\prod_{k \in K'} y_k \leq R^A(\underline{a}, \underline{y}) \iff A(\underline{a}) \leq \sum_{k \in K'} P_k(\underline{a})$$

La démonstration de ce théorème est analogue à celle du théorème 2.2. A nouveau, si l'impliquant $\prod y_k$ est premier, la couverture qu'il décrit est irrédondante.

La forme complète de $R^A(\underline{a}, \underline{y})$ contient tous les impliquants de la forme cherchée; on peut l'obtenir par double inversion :

$$\overline{R^A(\underline{a}, \underline{y})} = \overline{A(\underline{a}) \cdot M^A(\underline{a}, \underline{y})}$$

$$\overline{R^A(\underline{a}, \underline{y})} = \overline{a_1 a_2 a_3 a_4 y_1} + \overline{a_1 a_2 a_3 a_4 y_2 y_3} + \overline{a_1 a_3 a_4 y_4 y_5 y_6} + \\ + \overline{a_1 a_2 a_3 a_4 y_4 y_6} + \overline{a_1 a_2 a_3 a_4 y_5 y_6}$$

$$R^A(\underline{a}, \underline{y}) = y_1 y_2 y_6 + y_1 y_3 y_6 + y_1 y_2 y_4 y_5 + y_1 y_3 y_4 y_5 + \\ + a_4 y_2 + a_4 y_3 + a_2 a_4 + a_1 y_6 + a_1 \overline{a_3} + \dots$$

Lorsque l'on obtient plusieurs couvertures irrédondantes (et c'est en général le cas), il y a lieu de ne retenir que l'une de celles qui sont composées du plus petit nombre d'impliquants et qui correspondent aux impliquants de $R^A(\underline{a}, \underline{y})$ composés du plus petit nombre de variables \underline{y} .

Dans l'exemple cité, posons $u_1 = u_2 = u_6 = 1$ et $u_3 = u_4 = u_5 = 0$; grâce à l'impliquant $y_1 y_2 y_6 \leq R^A(\underline{a}, \underline{y})$, on a :

$$R^A(\underline{a}, \underline{u}) = 1 \quad \text{et} \quad MF^A(\underline{a}, \underline{u}) = a_2 \bar{a}_3 + \bar{a}_2 a_4 + a_1 \bar{a}_2 a_3$$

qui est l'une des formes minimales cherchées.

Les impliquants mixtes de $R^A(\underline{a}, \underline{y})$, c'est-à-dire ceux qui sont composés de variables \underline{a} et de variables \underline{y} (ou seulement de variables \underline{a}) décrivent des couvertures conditionnelles; ils ne sont pas intéressants dans le cas de cette méthode de synthèse à deux niveaux et sont, par conséquent, à éliminer avant la deuxième inversion en omettant toutes les variables \underline{a} dans l'expression $R^A(\underline{a}, \underline{y})$; cette omission permet d'éviter un travail considérable, car les impliquants mixtes sont en général très nombreux : dans l'exemple cité, qui est pourtant de taille très modeste, la forme complète de $R^A(\underline{a}, \underline{y})$ compte 43 impliquants premiers, dont 4 seulement ne sont composés que de variables \underline{y} .

$$R^A(\underline{a}, \underline{y}) = y_1 \cdot (y_2 + y_3) \cdot (y_4 + y_6) \cdot (y_5 + y_6) + \dots$$

	$\bar{a}_2 \bar{a}_3$	$\bar{a}_2 \bar{a}_4$	$\bar{a}_3 \bar{a}_4$	$\bar{a}_1 \bar{a}_2 \bar{a}_4$	$\bar{a}_1 \bar{a}_2 \bar{a}_3$	$\bar{a}_1 \bar{a}_3 \bar{a}_4$
$\bar{a}_1 \bar{a}_2 \bar{a}_3 \bar{a}_4$	x					
$\bar{a}_1 \bar{a}_2 \bar{a}_3 a_4$		x	x			
$a_1 \bar{a}_2 \bar{a}_3 \bar{a}_4$				x		x
$a_1 \bar{a}_2 a_3 \bar{a}_4$					x	x

Fig. 2.2 : fonction et tableau de Petrick.

La fonction obtenue après l'amputation des variables \underline{a} dans la fonction $R^A(\underline{a}, \underline{y})$ est connue sous le nom de fonction de Petrick [10] - [13], [23], [27] - [29]; on la trouve aussi représentée sous forme de tableau (fig. 2.2).

III. DESCRIPTION ALGEBRIQUE D'UNE TABLE DE VERITE

La table de vérité d'un système logique combinatoire décrit un ensemble de configurations possibles des signaux d'entrée et spécifie pour chacune d'elles une configuration des signaux de sortie (voir fig. 3.1). D. E. Muller [7] a montré qu'un système à sortie multiple peut en fait être décrit au moyen d'une fonction booléenne unique à l'aide de variables auxiliaires; le § 1 de ce chapitre présente une dérivation formelle de cette fonction.

1. Fonction de vérité

A chacun des $|N|$ signaux d'entrée, nous associons une variable booléenne indépendante a_n , $n \in N$. La partie d'entrée de la table de vérité est composée d'un tableau de valeurs binaires ("bits"), dont chaque élément peut être désigné par le couple d'indices (i,n) , i étant l'indice de l'une des $|I|$ lignes de la table; à chaque bit (i,n) on associe une fonction booléenne $E_{in}(a_n)$ de la variable indépendante correspondante, définie comme suit :

Définition 3.1

$$E_{in}(a_n) = 1 \iff \left\{ \begin{array}{l} \text{le signal d'entrée } a_n \text{ est en} \\ \text{accord avec le bit } a_n(i,n) \text{ de} \\ \text{la partie d'entrée de la table} \end{array} \right.$$

Les fonctions $E_{in}(a_n)$ s'obtiennent directement de la table de vérité :

$$E_{in}(a_n) = \begin{cases} a_n \\ 1 \\ \bar{a}_n \end{cases} \quad \begin{array}{l} \text{si le bit } (i,n) \text{ de la} \\ \text{table de vérité vaut} \end{array} \quad \begin{cases} 1 \\ x \\ 0 \end{cases}$$

Pour chaque ligne de la table de vérité, on définit alors une fonction $E_i(\underline{a})$ comme suit :

Définition 3.2

$$E_i(\underline{a}) = 1 \iff \begin{cases} \text{la configuration } \underline{a} \text{ des signaux} \\ \text{d'entrée réalise la ligne } i \text{ de} \\ \text{la partie d'entrée de la table} \end{cases}$$

La fonction $E_i(\underline{a})$ s'obtient par le produit des $|N|$ fonctions $E_{in}(a_n)$:

$$E_i(\underline{a}) = \prod_{n \in N} E_{in}(a_n)$$

On procède de façon identique pour la partie de sortie de la table de vérité : à chacun des $|J|$ signaux de sortie est associée une variable booléenne indépendante f_j , $j \in J$; on définit alors les fonctions de bit $S_{ij}(f_j)$ et de configuration $S_i(\underline{f})$:

Définition 3.3

$$S_{ij}(\phi_j) = 1 \iff \begin{cases} \text{le signal de sortie } \phi_j \text{ est en} \\ \text{accord avec le bit } (i,j) \text{ de} \\ \text{la partie de sortie de la table} \end{cases}$$

Définition 3.4

$$S_i(\underline{\phi}) = 1 \iff \left\{ \begin{array}{l} \text{la configuration } \underline{\phi} \text{ des signaux} \\ \text{de sortie réalise la ligne } i \text{ de} \\ \text{la partie de sortie de la table} \end{array} \right.$$

$$S_{ij}(f_j) = \begin{cases} f_j & \text{si le bit } (i,j) \text{ de la} \\ 1 & \text{table de vérité vaut} \\ \bar{f}_j & \end{cases} \begin{cases} 1 \\ X \\ 0 \end{cases}$$

$$S_i(\underline{f}) = \prod_{j \in J} S_{ij}(f_j)$$

On définit alors la fonction $V(\underline{a}, \underline{f})$, dite caractéristique de la table de vérité (de façon abrégée : fonction de vérité), comme suit :

Définition 3.5

$$V(\underline{\alpha}, \underline{\phi}) = 1 \iff \left\{ \begin{array}{l} \text{les configurations d'entrée } \underline{\alpha} \text{ et de} \\ \text{sortie } \underline{\phi} \text{ réalisent l'une des lignes} \\ \text{spécifiées dans la table de vérité} \end{array} \right.$$

On forme simplement cette fonction au moyen des états d'entrée $E_i(\underline{a})$ et de sortie $S_i(\underline{f})$:

$$V(\underline{a}, \underline{f}) = \sum_{i \in I} E_i(\underline{a}) \cdot S_i(\underline{f})$$

Nous disposons maintenant d'une fonction booléenne qui possède le même contenu en information que la table de vérité, et que nous pouvons utiliser au gré de nos besoins à l'aide d'opérateurs algébriques.

I	N				J		
	α_1	α_2	α_3	α_4	ϕ_1	ϕ_2	ϕ_3
1	X	0	0	0	X	1	X
2	1	1	0	0	0	0	X
3	0	0	X	0	X	1	X
4	1	0	1	0	1	0	1
5	0	1	1	0	0	0	X
6	1	1	1	0	1	1	0
7	1	0	0	1	0	0	X
8	X	1	0	1	1	X	X
9	0	0	1	1	0	0	X
10	1	0	1	1	1	1	0
11	0	1	X	1	1	X	X
12	1	1	1	1	0	1	1

$$\begin{aligned}
 V(\underline{a}, \underline{f}) &= \sum E_i(\underline{a}) \cdot S_i(\underline{f}) \\
 &= \bar{a}_2 \bar{a}_3 \bar{a}_4 f_2 + \\
 &+ a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + \\
 &+ \bar{a}_1 \bar{a}_2 \bar{a}_4 f_2 + \\
 &+ a_1 \bar{a}_2 a_3 \bar{a}_4 f_1 \bar{f}_2 f_3 + \\
 &+ \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + \\
 &+ a_1 a_2 a_3 \bar{a}_4 f_1 f_2 \bar{f}_3 + \\
 &+ a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 + \\
 &+ a_2 \bar{a}_3 a_4 f_1 + \\
 &+ \bar{a}_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 + \\
 &+ a_1 \bar{a}_2 a_3 a_4 f_1 f_2 \bar{f}_3 + \\
 &+ \bar{a}_1 a_2 a_4 f_1 + \\
 &+ a_1 a_2 a_3 a_4 \bar{f}_1 f_2 f_3
 \end{aligned}$$

Fig. 3.1 : table et fonction de vérité.

En plus de son intérêt théorique, la description algébrique d'une table de vérité au moyen d'une fonction booléenne est à la base d'une méthode de codage numérique pour le traitement par ordinateur des problèmes de synthèse (cf. app. C); cette représentation algébrique convient d'ailleurs particulièrement aux tables de vérité faiblement définies qui possèdent un grand nombre de signaux d'entrée et de sortie, typiques des systèmes qu'il est intéressant, actuellement, de soumettre à la synthèse.

2. Condition de cohérence

Nous avons vu (chap. I, § 1) qu'une table de vérité peut être canonique ou non; une table de vérité est canonique si chacune de ses lignes ne décrit qu'une seule configuration des signaux d'entrée; autrement dit, si la partie d'entrée de la table ne contient pas de valeurs indéfinies ("X", ou "don't care").

Dans le cas où la table n'est pas canonique, il faut veiller à ce qu'elle soit cohérente. La condition de cohérence s'énonce : si une configuration \underline{a} des signaux d'entrée peut réaliser simultanément deux lignes de la table, la partie de sortie de ces deux lignes doit être identique (mêmes "1", mêmes "X" et mêmes "0"); exemple : lignes n° 1 et 3 de la table présentée à la figure 3.1.

Une table canonique est naturellement cohérente, puisque dans ce cas, toutes les configurations d'entrée sont distinctes. L'énoncé algébrique de la condition de cohérence est le suivant :

$$E_{i_1}(\underline{a}) \cdot E_{i_2}(\underline{a}) \neq 0 \quad \Leftrightarrow \quad S_{i_1}(\underline{f}) = S_{i_2}(\underline{f}) \quad \forall i_1, i_2 \in I$$

On formule alors la fonction caractéristique de cette implication en vue d'en obtenir une équation algébrique :

$$\mathcal{G}: \quad E_{i_1}(\underline{a}) \cdot E_{i_2}(\underline{a}) \neq 0 \quad \Leftrightarrow \quad G(\underline{a}) = E_{i_1}(\underline{a}) \cdot E_{i_2}(\underline{a})$$

$$\mathcal{H}: \quad S_{i_1}(\underline{f}) = S_{i_2}(\underline{f})$$

$$\Leftrightarrow \quad H(\underline{f}) = \overline{S_{i_1}(\underline{f})} \cdot \overline{S_{i_2}(\underline{f})} + S_{i_1}(\underline{f}) \cdot S_{i_2}(\underline{f})$$

L'expression voulue est alors fournie par l'équation caractéristique de l'implication $\mathcal{G} \Rightarrow \mathcal{H}$, appliquée à toutes les paires d'indices :

$$\sum G(\underline{a}) \cdot \overline{H(\underline{f})} = 0$$

$$\sum_{i_1, i_2 \in I} E_{i_1}(\underline{a}) \cdot E_{i_2}(\underline{a}) \cdot \left[\overline{S_{i_1}(\underline{f})} \cdot S_{i_2}(\underline{f}) + S_{i_1}(\underline{f}) \cdot \overline{S_{i_2}(\underline{f})} \right] = 0$$

Par un regroupement de termes, on obtient une autre formulation :

$$\left[\sum_{i \in I} E_i(\underline{a}) \cdot S_i(\underline{f}) \right] \cdot \left[\sum_{i \in I} E_i(\underline{a}) \cdot \overline{S_i(\underline{f})} \right] = 0$$

Dans le cas d'une table de vérité canonique, on a $E_{i_1}(\underline{a}) \cdot E_{i_2}(\underline{a}) = 0$ pour toute paire d'indices distincts, ce qui constitue une condition suffisante pour annuler les deux expressions ci-dessus.

3. Transformations de la table de vérité

La fonction $V(\underline{a}, \underline{f})$ contient, sous une forme algébrique condensée, toute l'information spécifiée dans la table de vérité. Cependant, selon l'utilisation que l'on désire faire de cette information, il peut être nécessaire de transformer cette fonction; ainsi, il est possible d'en extraire les bornes inférieure et supérieure d'un sous-système individuel donné; d'autre part, lors de la synthèse classique sur deux niveaux d'un système complexe (cf. chap. V), nous utiliserons deux fonctions partielles $V^A(\underline{a}, \underline{f})$ et $V^B(\underline{a}, \underline{f})$ dérivées de la fonction de vérité qui s'apparentent, par leur contenu, à la généralisation pour un système complexe de la borne inférieure $A(\underline{a})$ et de l'inverse de la borne supérieure $B(\underline{a})$ d'un système simple.

Bornes d'un sous-système individuel

Il est possible d'extraire de la fonction de vérité $V(\underline{a}, \underline{f})$ les fonctions $A_j(\underline{a})$ (borne inférieure) et $B_j(\underline{a})$ (inverse de la borne supérieure) du sous-système individuel formé du seul signal de sortie ϕ_j ; ces deux fonctions s'expriment par la réunion de sous-ensembles

spécifiques (I_j^A , resp. I_j^B) d'états d'entrée $E_i(\underline{a})$:

$$A_j(\underline{a}) = \sum_{i \in I_j^A} E_i(\underline{a}) \quad \text{avec} \quad I_j^A = \{i | S_{ij}(f_j) = f_j\}$$

$$B_j(\underline{a}) = \sum_{i \in I_j^B} E_i(\underline{a}) \quad \text{avec} \quad I_j^B = \{i | S_{ij}(f_j) = \bar{f}_j\}$$

Exemple : cas du système défini sur la figure 3.1 (p. 28) :

$$A_1(\underline{a}) = a_1 \bar{a}_2 a_3 \bar{a}_4 + a_1 a_2 a_3 \bar{a}_4 + a_2 \bar{a}_3 a_4 + a_1 \bar{a}_2 a_3 a_4 + \bar{a}_1 a_2 a_4$$

$$A_2(\underline{a}) = \bar{a}_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 \bar{a}_4 + a_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 a_4 + a_1 a_2 a_3 a_4$$

$$A_3(\underline{a}) = a_1 \bar{a}_2 a_3 \bar{a}_4 + a_1 a_2 a_3 a_4$$

$$B_1(\underline{a}) = a_1 a_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 \bar{a}_3 a_4 + \bar{a}_1 \bar{a}_2 a_3 a_4 + a_1 a_2 a_3 a_4$$

$$B_2(\underline{a}) = a_1 a_2 \bar{a}_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 \bar{a}_4 + \bar{a}_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 \bar{a}_3 a_4 + \bar{a}_1 \bar{a}_2 a_3 a_4$$

$$B_3(\underline{a}) = a_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 a_4$$

Borne inférieure d'un système

La généralisation pour un système de la borne inférieure est une fonction $V^A(\underline{a}, \underline{f})$ que l'on construit sur le modèle de la fonction de vérité, mais en considérant comme des "0" les indéfinitions explicites ("X", ou "don't care") de la partie de sortie de la table de vérité :

$$S_{ij}^A(f_j) = \begin{cases} f_j \\ \bar{f}_j \\ \bar{f}_j \end{cases} \quad \text{si le bit } (i,j) \text{ de la} \quad \begin{cases} 1 \\ X \\ 0 \end{cases} \\ \text{table de vérité vaut}$$

$$S_i^A(\underline{f}) = \prod_{j \in J} S_{ij}^A(f_j) \quad \text{et} \quad V^A(\underline{a}, \underline{f}) = \sum_{i \in I} E_i(\underline{a}) \cdot S_i^A(\underline{f})$$

Dans la pratique (cf. chap. V, § 2), nous utiliserons une version modifiée $V^{A'}$ ($\underline{a}, \underline{f}$) de la fonction V^A ($\underline{a}, \underline{f}$) dans laquelle n'apparaissent que les variables f_j niées :

$$V^{A'}(\underline{a}, \underline{f}) = \sum_{i \in I} E_i(\underline{a}) \cdot \prod_{j \in J} [S_{ij}^A(f_j) + \bar{f}_j]$$

La formation de la fonction $V^{A'}$ ($\underline{a}, \underline{f}$) à partir de la fonction de vérité V ($\underline{a}, \underline{f}$) est immédiate (cf. exemple, fig. 3.2) :

a) on remplace dans chaque terme de la fonction de vérité toute variable f_j manquante par une variable f_j niée, et

b) on omet les variables f_j non niées restantes.

Borne supérieure d'un système

La généralisation pour un système de la borne supérieure est une fonction V^B ($\underline{a}, \underline{f}$) que l'on construit sur le modèle de la fonction de vérité, mais en considérant comme des "1" les indéfinitions explicites de la partie de sortie de la table de vérité :

$$S_{ij}^B(f_j) = \begin{cases} f_j \\ f_j \\ \bar{f}_j \end{cases} \quad \text{si le bit } (i,j) \text{ de la} \quad \begin{cases} 1 \\ X \\ 0 \end{cases} \\ \text{table de vérité vaut}$$

$$S_i^B(\underline{f}) = \prod_{j \in J} S_{ij}^B(f_j) \quad \text{et} \quad V^B(\underline{a}, \underline{f}) = \sum_{i \in I} E_i(\underline{a}) \cdot S_i^B(\underline{f})$$

Dans la pratique (cf. chap. V, § 1), nous utiliserons une version

$$\begin{aligned}
 V(\underline{a}, \underline{f}) &= \bar{a}_2 \bar{a}_3 \bar{a}_4 f_2 + a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + \\
 &+ \bar{a}_1 \bar{a}_2 \bar{a}_4 f_2 + a_1 \bar{a}_2 a_3 \bar{a}_4 f_1 \bar{f}_2 f_3 + \\
 &+ \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + a_1 a_2 a_3 \bar{a}_4 f_1 f_2 \bar{f}_3 + \\
 &+ a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 + a_2 \bar{a}_3 a_4 f_1 + \\
 &+ \bar{a}_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + a_1 \bar{a}_2 a_3 a_4 f_1 f_2 \bar{f}_3 + \\
 &+ \bar{a}_1 a_2 a_4 f_1 + a_1 a_2 a_3 a_4 \bar{f}_1 f_2 f_3 \\
 \\
 V^{A'}(\underline{a}, \underline{f}) &= \bar{a}_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_3 + a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \\
 &+ \bar{a}_1 \bar{a}_2 \bar{a}_4 \bar{f}_1 \bar{f}_3 + a_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_2 + \\
 &+ \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 \bar{a}_4 \bar{f}_3 + \\
 &+ a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_2 \bar{a}_3 a_4 \bar{f}_2 \bar{f}_3 + \\
 &+ \bar{a}_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 a_4 \bar{f}_3 + \\
 &+ \bar{a}_1 a_2 a_4 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 a_4 \bar{f}_1 \\
 \\
 V^{B'}(\underline{a}, \underline{f}) &= a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + a_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_2 + \\
 &+ \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + a_1 a_2 a_3 \bar{a}_4 \bar{f}_3 + \\
 &+ a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 + \bar{a}_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 + \\
 &+ a_1 \bar{a}_2 a_3 a_4 \bar{f}_3 + a_1 a_2 a_3 a_4 \bar{f}_1
 \end{aligned}$$

Fig. 3.2 : la fonction de vérité et ses versions modifiées.

modifiée $V^{B'}(\underline{a}, \underline{f})$ de la fonction $V^B(\underline{a}, \underline{f})$ faisant intervenir une modification de la partie en \underline{f} de chaque terme :

$$V^{B'}(\underline{a}, \underline{f}) = \sum_{i \in I} E_i(\underline{a}) \cdot \prod_{j \in J} [S_{ij}^B(f_j) + \bar{f}_j]$$

La formation de la fonction $V^{B'}(\underline{a}, \underline{f})$ à partir de la fonction de vérité $V(\underline{a}, \underline{f})$ est immédiate (cf. exemple, fig. 3.2) :

a) on omet les variables f_j non niées de chaque terme, et

b) on inverse par un jeu de notation (c'est-à-dire sans effectuer l'opération) la partie en \underline{f} de chacun des termes obtenus.

Remarque : lorsque la table de vérité ne mentionne pas d'indéfinitions explicites, on a :

$$V^A(\underline{a}, \underline{f}) = V^B(\underline{a}, \underline{f}) = V(\underline{a}, \underline{f}) \quad \text{mais :}$$

$$V^{A'}(\underline{a}, \underline{f}) \neq V^{B'}(\underline{a}, \underline{f}) \neq V(\underline{a}, \underline{f})$$

IV. TRAITEMENT SIMULTANE DE FONCTIONS BOOLEENNES

Dans ce chapitre, nous allons développer un outil destiné à traiter simultanément un ensemble de fonctions booléennes possédant des arguments communs.

L'utilisation d'un tel outil rend possible la recherche de solutions communes à un ensemble de relations d'ordre qui doivent être satisfaites simultanément, solutions matérialisées par les impliquants communs de leurs fonctions caractéristiques.

1. Définitions

Soit un ensemble de fonctions booléennes $\{R_j(\underline{x}) \mid j \in J\}$; on utilise un jeu de variables booléennes indépendantes $\underline{z} = \{z_j \mid j \in J\}$ pour composer cet ensemble en une fonction unique. La composition peut être conjonctive ou disjonctive :

Définition 4.1

$$R^{\cap}(\underline{x}, \underline{z}) = \prod_{j \in J} [R_j(\underline{x}) + \bar{z}_j]$$

Définition 4.2

$$R^{\cup}(\underline{x}, \underline{z}) = \sum_{j \in J} R_j(\underline{x}) \cdot z_j$$

L'introduction des variables \underline{z} permet de conserver l'information propre à chaque fonction individuelle $R_j(\underline{x})$ lors du mélange; en effet, toute fonction individuelle $R_j(\underline{x})$ peut être extraite de l'une ou de l'autre fonction composée en donnant une valeur particulière \underline{z} aux variables \underline{z} :

Propriété 4.3

$$R_{j_0}(\underline{x}) = R^{\Pi}(\underline{x}, \underline{z}) = R^{\Sigma}(\underline{x}, \underline{z}) \quad \text{avec} \quad \begin{cases} \zeta_{j_0} = 1 \\ \zeta_j = 0, j \neq j_0 \end{cases}$$

Il y a quatre possibilités de former une fonction composée au moyen d'un ensemble de fonctions individuelles, en tenant compte de la composition en somme ou en produit d'une part, sous forme directe ou sous forme inverse de l'autre; quel est l'intérêt de cette multiplicité ?

En fait, comme nous le verrons au § 2 de ce chapitre, seule la fonction composée en produit sous forme directe mentionne explicitement, dans son expression disjonctive, les impliquants communs aux fonctions individuelles $R_j(\underline{x})$.

Cependant, lorsque dans la pratique cette technique de composition est requise pour aborder un problème donné, il peut arriver, selon la forme sous laquelle se trouvent les données à disposition, que l'une des trois autres fonctions composées soit plus facile à construire que la forme directe de la fonction-produit (le cas se rencontre notamment lors de la recherche de la couverture globale d'un système, cf. chap. V, § 2).

Ainsi, après avoir démontré les propriétés des impliquants de la fonction composée en produit (§ 2 ci-dessous), nous développerons un algorithme qui permet le passage direct (c'est-à-dire sans recours à

une décomposition en fonctions individuelles) entre les fonctions-produit et somme, ainsi qu'entre leurs inverses (§ 3 de ce chapitre; cf. fig. 4.1).

2. Obtention des solutions communes

Impliquants de la fonction-produit

En partant de la définition de la fonction composée en produit, on peut obtenir une forme disjonctive en effectuant les produits; les impliquants obtenus sont alors composés d'un produit $P_k(\underline{x})$ de variables \underline{x} niées ou non, provenant des fonctions individuelles $R_j(\underline{x})$, et d'un produit de variables \underline{z} . Par construction, les impliquants premiers de la fonction-produit ne contiennent que des variables z_j niées; afin de faciliter l'interprétation de ces impliquants, nous désignerons par J_k le sous-ensemble spécifique (c'est-à-dire propre à chaque impliquant) des variables z_j niées qui n'y figurent pas :

$$R^{\Pi}(\underline{x}, \underline{z}) = \underbrace{\prod_{j \in J} [R_j(\underline{x}) + \bar{z}_j]}_{\text{définition}} = \sum_{k \in K} \underbrace{P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j}_{\text{forme disjonctive}}$$

Théorème 4.4

$$P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \leq R^{\Pi}(\underline{x}, \underline{z}) \iff P_k(\underline{x}) \leq \prod_{j \in J_k} R_j(\underline{x})$$

La partie en \underline{x} d'un impliquant de la fonction composée en produit est un impliquant commun à toutes les fonctions individuelles $R_j(\underline{x})$ dont les variables caractéristiques \bar{z}_j ne composent pas sa partie en \underline{z} .

Démonstration : a) implication dans le sens \Rightarrow :

$$\text{Hyp. : } P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \leq \prod_{j \in J} [R_j(\underline{x}) + \bar{z}_j]$$

$$\text{Posons : } z_j = 1, \quad j \in J_k \quad \text{et} \quad z_j = 0, \quad j \in \bar{J}_k$$

$$\Rightarrow P_k(\underline{x}) \leq \prod_{j \in J_k} R_j(\underline{x}) \quad \text{cqfd.}$$

b) implication dans le sens \Leftarrow : formons les deux fonctions :

$$G(\underline{x}) = \overline{P_k(\underline{x})} + \prod_{j \in J_k} R_j(\underline{x})$$

$$H(\underline{x}, \underline{z}) = \overline{P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j} + \prod_{j \in J} [R_j(\underline{x}) + \bar{z}_j]$$

évaluons alors l'expression $W(\underline{x}, \underline{z}) = \overline{G(\underline{x})} + H(\underline{x}, \underline{z})$:

$$\begin{aligned} W(\underline{x}, \underline{z}) &= P_k(\underline{x}) \cdot \overline{\prod_{j \in J_k} R_j(\underline{x})} + \overline{P_k(\underline{x})} + \sum_{j \in \bar{J}_k} z_j + \prod_{j \in J} [R_j(\underline{x}) + \bar{z}_j] \\ &= \overline{\prod_{j \in J_k} R_j(\underline{x})} + \dots + \prod_{j \in J_k} R_j(\underline{x}) = 1 \end{aligned}$$

ce qui entraîne $G(\underline{x}) \leq H(\underline{x}, \underline{z})$ et démontre l'implication.

Propriété 4.5

Si l'impliquant $P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \leq R^{\pi}(\underline{x}, \underline{z})$ est premier,

$$a) \quad \exists P'_k(\underline{x}) \neq P_k(\underline{x}) \quad \text{tq.} \quad P_k(\underline{x}) \leq P'_k(\underline{x}) \leq \prod_{j \in J_k} R_j(\underline{x})$$

$$b) \quad \exists J' \quad \text{tq.} \quad J_k \subset J' \quad \text{et} \quad P_k(\underline{x}) \leq \prod_{j \in J'} R_j(\underline{x})$$

Si $P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j$ est un impliquant premier de $R^{\pi}(\underline{x}, \underline{z})$, il n'existe pas d'impliquant $P'_k(\underline{x})$ incluant strictement $P_k(\underline{x})$ qui soit commun au même sous-ensemble J_k de fonctions $R_j(\underline{x})$ décrites par les variables \bar{z}_j : c'est la primauté en \underline{x} ; d'autre part, il n'existe pas non plus de sous-ensemble J' , incluant strictement J_k , de fonctions $R_j(\underline{x})$ desquelles $P_k(\underline{x})$ soit un impliquant commun : c'est la primauté en \underline{z} .

Autrement dit, chaque impliquant premier de la fonction-produit est un compromis optimal entre la taille de l'impliquant commun qu'il décrit et le nombre de fonctions individuelles auxquelles il est commun.

La forme complète de $R^{\pi}(\underline{x}, \underline{z})$ mentionne ainsi sous forme explicite tous les meilleurs compromis entre les divers impliquants communs et la taille des sous-ensembles de fonctions auxquelles ces impliquants sont communs.

Impliquants de la fonction-somme

La forme générale des impliquants de la fonction-somme comprend un produit $P_k(\underline{x})$ de variables \underline{x} niées ou non, ainsi qu'un produit de

variables \underline{z} . Par construction, les impliquants premiers de la fonction-somme ne contiennent que des variables z_j non niées; désignons par J_k le sous-ensemble de ces variables z_j particulier à chaque impliquant :

$$R^{\Sigma}(\underline{x}, \underline{z}) = \sum_{k \in K} P_k(\underline{x}) \cdot \prod_{j \in J_k} z_j$$

Théorème 4.6

$$P_k(\underline{x}) \cdot \prod_{j \in J_k} z_j \leq R^{\Sigma}(\underline{x}, \underline{z}) \iff P_k(\underline{x}) \leq \sum_{j \in J_k} R_j(\underline{x})$$

(Démonstration analogue à celle du théorème 4.4).

Contrairement au cas du produit, l'impliquant d'une somme n'a pas la propriété d'être impliquant de chacun de ses termes; ainsi, la signification des impliquants de la fonction composée en somme n'est pas aussi explicite que dans le cas de la fonction-produit.

Cependant, la fonction-somme, exprimée sous forme disjonctive, possède par construction les propriétés suivantes :

a) tout impliquant de la fonction-somme possède au moins une variable z_j non niée,

b) tout impliquant premier (et a fortiori non premier) de la fonction-somme possédant plus d'une variable z_j est inclus dans la somme des impliquants qui ne sont composés que d'une seule variable z_j non niée.

Ainsi, par construction, il est possible d'exprimer la fonction-somme au moyen d'une somme de termes composés d'un produit $P_k(\underline{x})$ de

variables \underline{x} niées ou non et d'une somme d'un sous-ensemble spécifique J_k de variables z_j non niées :

$$R^{\Sigma}(\underline{x}, \underline{z}) = \sum_{k \in K} [P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j]$$

Les termes $P_k(\underline{x}) \cdot \sum z_j$ ne sont pas des impliquants au sens strict; il est cependant possible de les interpréter :

Théorème 4.7

$$P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j \leq R^{\Sigma}(\underline{x}, \underline{z}) \iff P_k(\underline{x}) \leq \prod_{j \in J_k} R_j(\underline{x})$$

La partie en \underline{x} d'un tel terme est un impliquant commun à toutes les fonctions individuelles $R_j(\underline{x})$ dont les variables caractéristiques z_j composent sa partie en \underline{z} .

Démonstration : a) implication dans le sens \Rightarrow :

$$\text{Hyp. : } P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j \leq \sum_{j \in J} R_j(\underline{x}) \cdot z_j$$

$$\text{Posons : } z_{j_0} = 1 \quad \text{et} \quad z_j = 0, \quad j \neq j_0.$$

$$\Rightarrow P_k(\underline{x}) \leq R_{j_0}(\underline{x}) \quad \forall j_0 \in J_k$$

$$\Rightarrow P_k(\underline{x}) \leq \prod_{j \in J_k} R_j(\underline{x}) \quad \text{cqfd.}$$

b) démonstration dans le sens \Leftarrow : formons les deux fonctions :

$$G(\underline{x}) = \overline{P_k(\underline{x})} + \prod_{j \in J_k} R_j(\underline{x})$$

$$H(\underline{x}, \underline{z}) = \overline{P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j} + \sum_{j \in J} R_j(\underline{x}) \cdot z_j$$

évaluons alors l'expression $W(\underline{x}, \underline{z}) = \overline{G(\underline{x})} + H(\underline{x}, \underline{z})$:

$$\begin{aligned} W(\underline{x}, \underline{z}) &= P_k(\underline{x}) \cdot \overline{\prod_{j \in J_k} R_j(\underline{x})} + \overline{P_k(\underline{x})} + \prod_{j \in J_k} \bar{z}_j + \sum_{j \in J} R_j(\underline{x}) \cdot z_j \\ &= \overline{\prod_{j \in J_k} R_j(\underline{x})} + \dots + \prod_{j \in J_k} R_j(\underline{x}) = 1 \end{aligned}$$

on en déduit : $G(\underline{x}) \leq H(\underline{x}, \underline{z})$, ce qui démontre la deuxième implication du théorème 4.7.

Corollaire 4.8

$$P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j \leq R^\Sigma(\underline{x}, \underline{z}) \iff P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \leq R^\Pi(\underline{x}, \underline{z})$$

Les termes composés d'une somme de variables z_j qui forment la fonction-somme sont équivalents aux impliquants de la fonction-produit.

C'est sur la base de cette équivalence qu'un algorithme sera développé pour effectuer le passage direct entre les fonctions-produit et somme.

3. Règles de transformation

Montrons qu'il est possible d'obtenir la fonction-somme par transformation terme à terme, selon le corollaire 4.8, d'une expression disjonctive de la fonction-produit :

Théorème 4.9

$$R^{\Pi}(\underline{x}, \underline{z}) = \sum_{k \in K} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \quad \Leftrightarrow \quad R^{\Sigma}(\underline{x}, \underline{z}) = \sum_{k \in K} [P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j]$$

La transformation n'est valable que dans le sens fonction-produit \rightarrow fonction-somme, car dans ce sens la transformation d'un impliquant inclus dans une somme d'autres impliquants fournit un terme lui-aussi inclus (lemme 4.11), alors que cette propriété n'est pas valable dans le sens inverse.

Lemme 4.10

$$\text{Si} \quad P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j \leq \sum_{k \in K'} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \quad \text{où} \quad K' \subseteq K$$

$$\text{alors} \quad P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j \leq \sum_{k \in K''} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j$$

$$\text{avec} \quad K'' = \{k | J_0 \subseteq J_k\} \subseteq K'$$

Si un impliquant $P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j$ est inclus dans la somme d'un ensemble K' d'autres impliquants, alors il est inclus dans la somme du sous-ensemble K'' de ceux-ci qui ne contiennent pas d'autres variables \bar{z}_j que lui.

Démonstration :

$$\text{Hyp. : } P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j \leq \sum_{k \in K'} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j$$

$$\text{Posons : } z_j = 1, \quad j \in J_0 \quad \text{et} \quad z_j = 0, \quad j \in \bar{J}_0$$

$$\Rightarrow P_0(\underline{x}) \leq \sum_{k \in K''} P_k(\underline{x})$$

$$\Rightarrow P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j \leq \sum_{k \in K''} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j$$

$$\Rightarrow P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j' \leq \sum_{k \in K''} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \quad \text{cqfd.}$$

Lemme 4.11

$$\text{Si } P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j \leq \sum_{k \in K'} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j$$

$$\text{alors } P_0(\underline{x}) \cdot \sum_{j \in J_0} z_j \leq \sum_{k \in K'} [P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j]$$

La transformation selon le corollaire 4.8 d'un impliquant inclus de la fonction-produit fournit un terme inclus de la fonction-somme.

Démonstration : par le lemme 4.10, on a la propriété :

$$P_0(\underline{x}) \cdot \prod_{j \in \bar{J}_0} \bar{z}_j \leq \sum_{k \in K''} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \quad \text{avec} \quad K'' = \{k | J_0 \subseteq J_k\}$$

Posons : $z_j = 1, j \in J_0$ et $z_j = 0, j \in \bar{J}_0$.

$$\Rightarrow P_0(\underline{x}) \leq \sum_{k \in K''} P_k(\underline{x})$$

$$\Rightarrow P_0(\underline{x}) \cdot \sum_{j \in J_0} z_j \leq \sum_{k \in K''} [P_k(\underline{x}) \cdot \sum_{j \in J_0} z_j]$$

$$\Rightarrow P_0(\underline{x}) \cdot \sum_{j \in J_0} z_j \leq \sum_{k \in K''} [P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j]$$

$$\Rightarrow P_0(\underline{x}) \cdot \sum_{j \in J_0} z_j \leq \sum_{k \in K'} [P_k(\underline{x}) \cdot \sum_{j \in J_k} z_j] \quad \text{cqfd.}$$

La contraposée du lemme 4.11 s'énonce : tout terme nécessaire (non inclus) de la fonction-somme provient d'un impliquant nécessaire de la fonction-produit; ceci démontre le théorème 4.9.

Diagramme de transformation

Le théorème 4.9 définit un nouvel opérateur qui agit terme à terme sur une expression disjonctive, de façon sélective (sur les variables \underline{z} seulement), par triple inversion :

a) des variables : $\bar{z}_j \rightarrow z_j$

b) de l'opérateur : $\prod \rightarrow \sum$

c) du sous-ensemble : $\bar{J}_k \rightarrow J_k$

nous appellerons translation ce nouvel opérateur.

L'inverse de la fonction composée en produit s'écrit :

$$\overline{R^{\Pi}(\underline{x}, \underline{z})} = \sum_{j \in J} \overline{R_j(\underline{x})} \cdot z_j$$

cette fonction possède la forme et, par conséquent, les propriétés d'une fonction composée en somme; réciproquement, l'inverse de la fonction-somme s'écrit :

$$\overline{R^{\Sigma}(\underline{x}, \underline{z})} = \prod_{j \in J} [\overline{R_j(\underline{x})} + \overline{z}_j]$$

cette fonction possède la forme et les propriétés d'une fonction composée en produit.

Ainsi, l'opérateur de translation permet d'obtenir la fonction-somme à partir de la fonction-produit exprimée sous forme disjonctive, ainsi que l'inverse de la fonction-produit à partir de l'inverse de la fonction-somme, également exprimée sous forme disjonctive (cf. fig. 4.1).

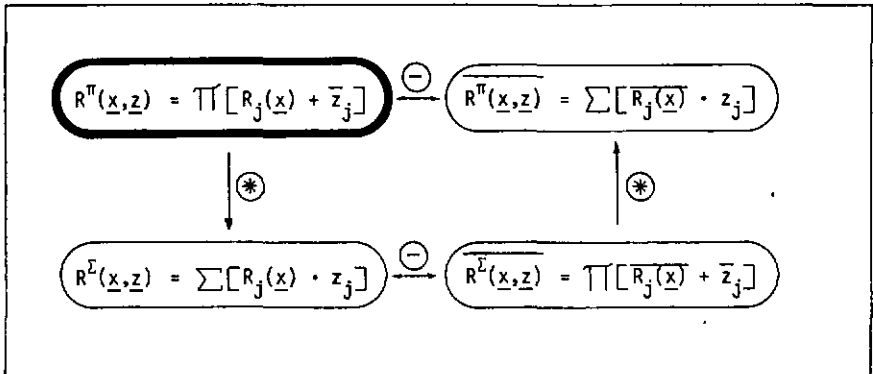


Fig. 4.1 : fonctions composées et règles de transformation;

(*) : opérateur de translation,

(-) : opérateur d'inversion.

Parmi ces quatre formulations, celle qu'il faut chercher à obtenir est la fonction-produit sous forme disjonctive directe, dont il est possible d'interpréter immédiatement les impliquants premiers.

Cependant, lorsque l'on envisage d'utiliser cette méthode pour traiter un problème donné, il peut arriver que la fonction-produit soit plus difficile à formuler que l'une des autres fonctions, à partir des données à disposition; dans ce cas, l'utilisation des règles de passage permet d'aboutir au résultat voulu sans avoir recours à une décomposition en fonctions individuelles.

4. Aspect algorithmique

Lorsque l'application de l'opérateur de translation \odot est suivie de l'inversion \ominus , ce qui est le cas lorsque l'on recherche la forme complète de la fonction-produit (cf. fig. 4.1), il faut éviter d'effectuer l'expansion en somme du produit des variables \underline{z} de chaque terme, afin de réduire le nombre des opérations qui seront à effectuer lors de l'inversion :

$$\overline{R^{\Sigma}(\underline{x}, \underline{z})} = \sum_{k \in K} P_k(\underline{x}) \cdot \prod_{j \in \bar{J}_k} \bar{z}_j \quad (\text{forme disjonctive})$$

$$\odot \Rightarrow \overline{R^{\Pi}(\underline{x}, \underline{z})} = \sum_{k \in K} P_k(\underline{x}) \cdot \overline{\prod_{j \in J_k} \bar{z}_j}$$

$$\ominus \Rightarrow R^{\Pi}(\underline{x}, \underline{z}) = \prod_{k \in K} \left[\overline{P_k(\underline{x})} + \prod_{j \in J_k} \bar{z}_j \right]$$

En effectuant alors les produits de l'expression ci-dessus, on obtient la forme complète de la fonction composée cherchée.

Exemple : $J = \{1, 2, 3, 4, 5\}$

$$\overline{R^{\Sigma}(\underline{x}, \underline{z})} = x_1 \bar{x}_3 x_4 \bar{z}_2 \bar{z}_5 + \bar{x}_1 x_2 \bar{z}_1 \bar{z}_2 \bar{z}_4$$

$$R^{\Pi}(\underline{x}, \underline{z}) = x_1 \bar{x}_3 x_4 \overline{\bar{z}_1 \bar{z}_3 \bar{z}_4} + \bar{x}_1 x_2 \overline{\bar{z}_3 \bar{z}_5}$$

$$\begin{aligned} R^{\Pi}(\underline{x}, \underline{z}) &= (\bar{x}_1 + x_3 + \bar{x}_4 + \bar{z}_1 \bar{z}_3 \bar{z}_4) \cdot (x_1 + \bar{x}_2 + \bar{z}_3 \bar{z}_5) \\ &= \bar{x}_1 \bar{x}_2 + \bar{x}_1 \bar{z}_3 \bar{z}_5 + x_1 x_3 + \bar{x}_2 x_3 + x_3 \bar{z}_3 \bar{z}_5 + x_1 \bar{x}_4 + \\ &\quad + \bar{x}_2 \bar{x}_4 + \bar{x}_4 \bar{z}_3 \bar{z}_5 + x_1 \bar{z}_1 \bar{z}_3 \bar{z}_4 + \bar{x}_2 \bar{z}_1 \bar{z}_3 \bar{z}_4 + \\ &\quad + \bar{z}_1 \bar{z}_2 \bar{z}_3 \bar{z}_4 \bar{z}_5 \end{aligned}$$

Le nombre de produits à effectuer pour obtenir la forme disjonctive cherchée se trouve considérablement augmenté lorsque l'expansion des sommes de variables \underline{z} a eu lieu :

$$\overline{R^{\Sigma}(\underline{x}, \underline{z})} = x_1 \bar{x}_3 x_4 z_1 + x_1 \bar{x}_3 x_4 z_3 + x_1 \bar{x}_3 x_4 z_4 + \bar{x}_1 x_2 z_3 + \\ + \bar{x}_1 x_2 z_5$$

$$\begin{aligned} R^{\Pi}(\underline{x}, \underline{z}) &= (\bar{x}_1 + x_3 + \bar{x}_4 + \bar{z}_1) \cdot (\bar{x}_1 + x_3 + \bar{x}_4 + \bar{z}_3) \cdot \\ &\quad \cdot (\bar{x}_1 + x_3 + \bar{x}_4 + \bar{z}_4) \cdot (x_1 + \bar{x}_2 + \bar{z}_3) \cdot (x_1 + \bar{x}_2 + \bar{z}_5) \end{aligned}$$

Dans la pratique, les opérateurs de translation \odot et d'inversion \ominus se contractent en une seule opération :

$$\overline{R^{\Sigma}(\underline{x}, \underline{z})} = \sum_{k \in K} P_k(\underline{x}) \cdot \prod_{j \in J_k} \bar{z}_j \quad (\text{forme disjonctive})$$

$$R^{\Pi}(\underline{x}, \underline{z}) = \prod_{k \in K} [\overline{P_k(\underline{x})} + \prod_{j \in J_k} \bar{z}_j] \quad (\text{produit à effectuer}).$$

V. APPLICATION A LA SYNTHÈSE CLASSIQUE SUR DEUX NIVEAUX

D'UN SYSTEME LOGIQUE COMBINATOIRE A SORTIE MULTIPLE

Soit un système logique combinatoire donné par sa fonction de vérité (fig. 5.1).

		N entrées	J sorties
I lignes	=	$\bar{a}_2 \bar{a}_3 \bar{a}_4$	f_2
	+	$a_1 \bar{a}_2 \bar{a}_3 \bar{a}_4$	$\bar{f}_1 \bar{f}_2$
	+	$\bar{a}_1 \bar{a}_2 \bar{a}_4$	f_2
	+	$a_1 \bar{a}_2 a_3 \bar{a}_4$	$f_1 \bar{f}_2 f_3$
	+	$\bar{a}_1 a_2 a_3 \bar{a}_4$	$\bar{f}_1 \bar{f}_2$
	+	$a_1 a_2 a_3 \bar{a}_4$	$f_1 f_2 \bar{f}_3$
	+	$a_1 \bar{a}_2 \bar{a}_3 a_4$	$\bar{f}_1 \bar{f}_2$
	+	$a_2 \bar{a}_3 a_4$	f_1
	+	$\bar{a}_1 \bar{a}_2 a_3 a_4$	$\bar{f}_1 \bar{f}_2$
	+	$a_1 \bar{a}_2 a_3 a_4$	$f_1 f_2 \bar{f}_3$
	+	$\bar{a}_1 a_2 a_4$	f_1
	+	$a_1 a_2 a_3 a_4$	$\bar{f}_1 f_2 f_3$

Fig. 5.1 : fonction de vérité d'un système à synthétiser, disposée en tableau.

Lors de la synthèse sur deux niveaux de systèmes logiques à sortie multiple, on procède de la même manière que pour un système à sortie

unique (cf. chap. II, § 3); cependant, au lieu de calculer individuellement les fonctions caractéristiques $R_j^B(\underline{a})$ et $R_j^A(\underline{a}, \underline{y})$ de chacun des $|J|$ sous-systèmes particuliers constitué par chacune des sorties prise isolément, on fait intervenir les procédés de composition décrits au chapitre IV de façon à obtenir des solutions communes.

1. Recherche des impliquants premiers du système

Un impliquant du système est un produit élémentaire $P_k(\underline{a})$ possédant la propriété d'avoir une intersection nulle avec les zéros d'un sous-ensemble J_k de sous-systèmes individuels; par impliquant premier du système, on désigne un terme qui offre un compromis optimal entre sa taille (c'est-à-dire son pouvoir de couverture, qui augmente lorsque le nombre des variables qui le composent diminue) et le nombre de sous-systèmes pour la réalisation desquels il est susceptible de convenir.

Les fonctions $B_j(\underline{a})$ rassemblent, pour chaque sous-système individuel, le sous-ensemble spécifique I_j^B des états d'entrée $E_i(\underline{a})$ pour lesquels le signal de sortie correspondant ϕ_j est assigné à "0" dans la table de vérité (cf. chap. III, § 3) :

$$B_j(\underline{a}) = \sum_{i \in I_j^B} E_i(\underline{a}) \quad \text{avec} \quad I_j^B = \{i | S_{ij}(f_j) = \bar{F}_j\}$$

$$B_1(\underline{a}) = a_1 a_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 \bar{a}_3 a_4 + \bar{a}_1 \bar{a}_2 a_3 a_4 + a_1 a_2 a_3 a_4$$

$$B_2(\underline{a}) = a_1 a_2 \bar{a}_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 \bar{a}_4 + \bar{a}_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 \bar{a}_3 a_4 + \bar{a}_1 \bar{a}_2 a_3 a_4$$

$$B_3(\underline{a}) = a_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 a_4$$

Nous avons vu (chap. II, § 3) que l'obtention des impliquants premiers de chaque sous-système pris individuellement peut se faire par la formulation d'un modèle $MF_j^B(\underline{a})$ réunissant tous les impliquants possibles (absorbés par le plus grand d'entre eux, 1), auquel on impose une intersection nulle avec la fonction correspondante $B_j(\underline{a})$:

$$\mathcal{R}_j^B(MF_j^B(\underline{a}), B_j(\underline{a})) : MF_j^B(\underline{a}) \cdot B_j(\underline{a}) = 0$$

$$\iff R_j^B(\underline{a}) = \overline{MF_j^B(\underline{a}) \cdot B_j(\underline{a})} = \overline{B_j(\underline{a})}$$

Dans le but d'effectuer un calcul simultané, on réunit ces fonctions $R_j^B(\underline{a})$ en une fonction composée unique; dans le cas présent, il est possible de former directement la fonction-produit :

$$R^{B\pi}(\underline{a}, \underline{f}) = \prod_{j \in J} [R_j^B(\underline{a}) + \bar{f}_j] = \prod_{j \in J} [\prod_{i \in I_j^B} \overline{E_i(\underline{a})} + \bar{f}_j]$$

Il serait alors possible de calculer cette expression en effectuant le double produit; cependant, la formation de l'expression ci-dessus, ainsi présentée, est malaisée, car elle fait intervenir la décomposition de la fonction de vérité en sous-ensembles spécifiques I_j^B d'états d'entrée $E_i(\underline{a})$.

Formation directe de la fonction composée

C'est à ce stade qu'une utilisation judicieuse des fonctions $S_{ij}^B(f_j)$, définies au chapitre III (§ 3), permet d'étendre les sous-ensembles spécifiques I_j^B à tout l'ensemble I ; et c'est d'ailleurs dans ce but que l'on a utilisé les mêmes variables \underline{f} pour décrire les $|J|$ signaux de sortie de la table de vérité et pour composer les $|J|$ fonctions individuelles $R_j^B(\underline{a})$.

$$s_{ij}^B(f_j) = \begin{cases} f_j, & i \in \overline{I_j^B} \\ \bar{f}_j, & i \in I_j^B \end{cases}$$

$$R^{B\pi}(\underline{a}, \underline{f}) = \prod_{j \in J} \left[\prod_{i \in I} (\overline{E_i(\underline{a})} + s_{ij}^B(f_j)) + \bar{f}_j \right]$$

La validité de la transformation effectuée est évidente si l'on considère l'inverse de l'expression entre crochets. Quelques manipulations de la fonction ainsi formulée nous permettent alors de la rapprocher considérablement de la fonction de vérité (qui constitue la donnée du problème) :

$$\begin{aligned} R^{B\pi}(\underline{a}, \underline{f}) &= \prod_{j \in J} \prod_{i \in I} [\overline{E_i(\underline{a})} + s_{ij}^B(f_j) + \bar{f}_j] \\ &= \prod_{i \in I} \prod_{j \in J} [\overline{E_i(\underline{a})} + s_{ij}^B(f_j) + \bar{f}_j] \\ &= \prod_{i \in I} [\overline{E_i(\underline{a})} + \prod_{j \in J} (s_{ij}^B(f_j) + \bar{f}_j)] \\ &= \overline{V^{B'}(\underline{a}, \underline{f})} \end{aligned}$$

Rappelons que la fonction $V^{B'}(\underline{a}, \underline{f})$ s'obtient simplement de la fonction de vérité $V(\underline{a}, \underline{f})$ en omettant dans cette dernière toutes les variables f_j non niées, puis en inversant, par un jeu de notation (c'est-à-dire sans effectuer l'opération), la partie en \underline{f} de chaque terme obtenu (cf. chap. III, § 3).

Par rapport à la formulation initiale de $R^{B\pi}(\underline{a}, \underline{f})$, on note la disparition du double produit, ainsi que celle des sous-ensembles spécifiques I_j^B ; il en résulte notamment une réduction du nombre des opérations à effectuer pour exprimer $R^{B\pi}(\underline{a}, \underline{f})$ sous forme disjonctive.

Exemple :

$$\begin{aligned} \gamma^{B'}(\underline{a}, \underline{f}) = & \bar{a}_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 + a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + \bar{a}_1 \bar{a}_2 \bar{a}_4 \bar{f}_1 + \\ & + a_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_2 + \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 + a_1 a_2 a_3 \bar{a}_4 \bar{f}_3 + \\ & + a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 + a_2 \bar{a}_3 a_4 \bar{f}_1 + \bar{a}_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 + \\ & + a_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_3 + \bar{a}_1 a_2 a_4 \bar{f}_1 + a_1 a_2 a_3 a_4 \bar{f}_1 \end{aligned}$$

$$\begin{aligned} R^{B\pi}(\underline{a}, \underline{f}) = & (\bar{a}_1 + \bar{a}_2 + a_3 + a_4 + \bar{f}_1 \bar{f}_2) \cdot (\bar{a}_1 + a_2 + \bar{a}_3 + a_4 + \bar{f}_2) \cdot \\ & \cdot (a_1 + \bar{a}_2 + \bar{a}_3 + a_4 + \bar{f}_1 \bar{f}_2) \cdot (\bar{a}_1 + \bar{a}_2 + \bar{a}_3 + a_4 + \bar{f}_3) \cdot \\ & \cdot (\bar{a}_1 + a_2 + a_3 + \bar{a}_4 + \bar{f}_1 \bar{f}_2) \cdot (a_1 + a_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_1 \bar{f}_2) \cdot \\ & \cdot (\bar{a}_1 + a_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_3) \cdot (\bar{a}_1 + \bar{a}_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_1) \end{aligned}$$

$$\begin{aligned} R^{B\pi}(\underline{a}, \underline{f}) = & \bar{f}_1 \bar{f}_2 \bar{f}_3 + \bar{a}_1 \bar{a}_3 + a_1 a_2 a_3 \bar{f}_1 \bar{f}_3 + \bar{a}_1 \bar{f}_1 \bar{f}_2 + \bar{a}_3 \bar{f}_1 \bar{f}_2 + \\ & + \bar{a}_2 \bar{a}_4 \bar{f}_2 + a_2 a_4 \bar{f}_1 + \bar{a}_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 \bar{a}_4 + \\ & + a_1 a_3 a_4 \bar{f}_1 \bar{f}_3 + a_1 \bar{a}_2 a_3 \bar{f}_2 \bar{f}_3 + a_1 a_3 \bar{a}_4 \bar{f}_2 \bar{f}_3 + \\ & + a_1 a_2 a_3 \bar{a}_4 \bar{f}_3 + \bar{a}_1 a_2 a_4 + a_2 \bar{a}_3 a_4 + a_1 \bar{a}_2 a_3 a_4 \bar{f}_3 \end{aligned}$$

La partie en \underline{f} des impliquants obtenus (p. ex. $\bar{f}_1 \bar{f}_3$ dans $a_1 a_2 a_3 \bar{f}_1 \bar{f}_3$) décrit l'ensemble des sous-systèmes pour la réalisation desquels l'impliquant $P_k(\underline{a})$ représenté par la partie en \underline{a} (ici $a_1 a_2 a_3$) ne peut pas être requis; dans ce cas, $a_1 a_2 a_3$ est un impliquant du sous-système n° 2. La figure 5.2 présente une interprétation de ce type pour tous les impliquants de $R^{B\pi}(\underline{a}, \underline{f})$ obtenus.

Interprétation du résultat

Les impliquants premiers qui constituent la forme complète de la fonction composée $R^{B\pi}(\underline{a}, \underline{f})$ possèdent une partie $P_k(\underline{a})$ formée d'un produit de variables \underline{a} niées ou non, et un produit de variables \underline{f} qui sont, par construction, toutes niées; désignons par J_k l'ensemble des variables \bar{f}_j qui n'apparaissent pas dans l'impliquant d'indice k :

$$R^{B\pi}(\underline{a}, \underline{f}) = \sum_{k \in K} P_k(\underline{a}) \cdot \prod_{j \in \bar{J}_k} \bar{f}_j$$

L'interprétation de ces impliquants est fournie par le théorème suivant :

Théorème 5.1

$$P_k(\underline{a}) \cdot \prod_{j \in \bar{J}_k} \bar{f}_j \leq R^{B\pi}(\underline{a}, f) \iff P_k(\underline{a}) \cdot B_j(\underline{a}) = 0 \quad \forall j \in J_k$$

Démonstration : immédiate, par la construction de la fonction $R^{B\pi}(\underline{a}, f)$, l'application du théorème 4.4, ainsi que l'équivalence :

$$P_k(\underline{a}) \cdot B_j(\underline{a}) = 0 \quad \forall j \in J_k \iff P_k(\underline{a}) \leq \prod_{j \in J_k} \overline{B_j(\underline{a})}$$

sous-systèmes	impliquants du système	partie en \underline{f}
1 2 3	$\bar{a}_1 \bar{a}_3$ $\bar{a}_2 \bar{a}_3 \bar{a}_4$ $\bar{a}_1 \bar{a}_2 \bar{a}_4$ $\bar{a}_1 a_2 a_4$ $a_2 \bar{a}_3 a_4$	1
2 3	$a_2 a_4$	\bar{f}_1
1 3	$\bar{a}_2 \bar{a}_4$	\bar{f}_2
1 2	$a_1 a_2 a_3 \bar{a}_4$ $a_1 \bar{a}_2 a_3 a_4$	\bar{f}_3
3	\bar{a}_1 \bar{a}_3	$\bar{f}_1 \bar{f}_2$
2	$a_1 a_2 a_3$ $a_1 a_3 a_4$	$\bar{f}_1 \bar{f}_3$
1	$a_1 \bar{a}_2 a_3$ $a_1 a_3 \bar{a}_4$	$\bar{f}_2 \bar{f}_3$
aucuns	1	$\bar{f}_1 \bar{f}_2 \bar{f}_3$

Fig. 5.2 : signification des impliquants de $R^{B\pi}(\underline{a}, f)$.

De plus, la propriété 4.5 garantit, pour chaque impliquant, un compromis optimal entre la taille de sa partie $P_k(\underline{a})$ et le nombre $|J_k|$ de sous-systèmes à la construction desquels il est autorisé à participer; en effet :

$$P_k(\underline{a}) \cdot \prod_{j \in \bar{J}_k} \bar{f}_j \leq R^{B\pi}(\underline{a}, \underline{f}) \quad \text{premier, implique :}$$

$$\begin{aligned} \text{a) } \quad \nexists P'_k(\underline{a}) \neq P_k(\underline{a}) \quad \text{tq.} \quad P_k(\underline{a}) \leq P'_k(\underline{a}) \\ \text{et} \quad P'_k(\underline{a}) \cdot B_j(\underline{a}) = 0 \quad \forall j \in J_k \\ \text{b) } \quad \nexists J' \supset J \quad \text{tq.} \quad P_k(\underline{a}) \cdot B_j(\underline{a}) = 0 \quad \forall j \in J' \end{aligned}$$

Ainsi, la fonction $R^{B\pi}(\underline{a}, \underline{f})$ contient, sous forme explicite, tous les meilleurs éléments de construction parmi lesquels seront choisis ceux qui réaliseront le système à moindres frais.

Remarque : la façon de procéder décrite ci-dessus remplace en améliorant la méthode connue qui consiste à chercher les impliquants premiers de toutes les intersections des bornes supérieures du système : dans le cas présent, seules sont retenues les intersections qui contiennent des impliquants premiers n'appartenant pas à l'intersection d'un ensemble plus grand de fonctions; par exemple, les sept sorties du système qui réalise la conversion du code BCD ("binary coded decimal") en un affichage à 7 segments possèdent 120 intersections, alors que ce système ne compte que 25 impliquants premiers :

$$\begin{aligned} V(\underline{a}, \underline{f}) = & \bar{a}_1 \bar{a}_2 \bar{a}_3 \bar{a}_4 f_1 f_2 f_3 f_4 f_5 f_6 \bar{f}_7 + a_1 \bar{a}_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 f_2 f_3 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + \\ & + \bar{a}_1 a_2 \bar{a}_3 \bar{a}_4 f_1 f_2 \bar{f}_3 f_4 f_5 \bar{f}_6 f_7 + a_1 a_2 \bar{a}_3 \bar{a}_4 f_1 f_2 f_3 f_4 \bar{f}_5 \bar{f}_6 f_7 + \\ & + \bar{a}_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_1 f_2 f_3 \bar{f}_4 \bar{f}_5 f_6 f_7 + a_1 \bar{a}_2 a_3 \bar{a}_4 f_1 \bar{f}_2 f_3 f_4 \bar{f}_5 f_6 f_7 + \\ & + \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 f_3 f_4 f_5 f_6 f_7 + a_1 a_2 a_3 \bar{a}_4 f_1 f_2 f_3 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + \\ & + \bar{a}_1 \bar{a}_2 \bar{a}_3 a_4 f_1 f_2 f_3 f_4 f_5 f_6 f_7 + a_1 \bar{a}_2 \bar{a}_3 a_4 f_1 f_2 f_3 \bar{f}_4 \bar{f}_5 f_6 f_7 \end{aligned}$$

$$\begin{aligned}
R^{B\pi}(\underline{a}, \underline{f}) = & \bar{f}_1 \bar{f}_2 \bar{f}_3 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + a_4 \bar{f}_4 \bar{f}_5 + a_3 \bar{f}_1 \bar{f}_2 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + \\
& + a_3 a_4 + \bar{a}_3 \bar{f}_1 \bar{f}_3 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + a_2 a_4 + a_2 \bar{a}_3 \bar{f}_3 \bar{f}_5 \bar{f}_6 + \\
& + \bar{a}_2 \bar{f}_1 \bar{f}_2 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + \bar{a}_2 a_3 \bar{f}_1 \bar{f}_2 \bar{f}_4 \bar{f}_5 + \bar{a}_2 \bar{a}_3 \bar{f}_1 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + \\
& + a_1 \bar{f}_1 \bar{f}_2 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + a_1 a_3 \bar{f}_2 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + \\
& + a_1 \bar{a}_3 \bar{f}_1 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + a_1 a_2 \bar{f}_4 \bar{f}_5 \bar{f}_6 \bar{f}_7 + a_1 a_2 \bar{a}_3 \bar{f}_5 \bar{f}_6 + \\
& + a_1 \bar{a}_2 a_3 \bar{f}_2 \bar{f}_5 + \bar{a}_1 a_4 + \bar{a}_1 a_3 \bar{f}_1 \bar{f}_2 \bar{f}_4 \bar{f}_5 + \bar{a}_1 \bar{a}_3 \bar{f}_3 \bar{f}_6 \bar{f}_7 + \\
& + \bar{a}_1 a_2 \bar{f}_1 \bar{f}_2 \bar{f}_3 \bar{f}_6 + \bar{a}_1 a_2 a_3 \bar{f}_1 \bar{f}_2 + \bar{a}_1 a_2 \bar{a}_3 \bar{f}_3 \bar{f}_6 + \\
& + \bar{a}_1 \bar{a}_2 \bar{f}_1 \bar{f}_4 \bar{f}_5 \bar{f}_7 + \bar{a}_1 \bar{a}_2 a_3 \bar{f}_1 \bar{f}_4 \bar{f}_5 + \bar{a}_1 \bar{a}_2 \bar{a}_3 \bar{f}_7
\end{aligned}$$

2. Recherche des couvertures irrédondantes globales

Le problème consiste à rechercher un sous-ensemble $K' \subseteq K$ d'impliquants $P_k(\underline{a})$ qui suffit à couvrir la borne inférieure $A_j(\underline{a})$ de chacun des sous-systèmes individuels, en tenant compte des restrictions indiquées par la partie en \underline{f} qui accompagne chaque impliquant $P_k(\underline{a})$ dans l'expression disjonctive de $R^{B\pi}(\underline{a}, \underline{f})$ précédemment obtenue.

Individuellement, on réaliserait cette recherche par la formulation d'un modèle paramétré de couverture comprenant, parmi les impliquants premiers du système, ceux qui sont compatibles avec un sous-système donné; on utilise un jeu de variables indépendantes $\underline{y} = \{y_k | k \in K\}$ pour fabriquer ce modèle.

$$\text{A partir de : } R^{B\pi}(\underline{a}, \underline{f}) = \sum_{k \in K} P_k(\underline{a}) \cdot \prod_{j \in \bar{J}_k} \bar{f}_j$$

$$\text{on forme : } MF_j^A(\underline{a}, \underline{y}) = \sum_{k \in K_j} P_k(\underline{a}) \cdot y_k$$

$$\text{avec : } K_j = \{k | j \in J_k\} \subseteq K$$

On impose alors à ce modèle de couvrir la borne inférieure correspondante $A_j(\underline{a})$, qui s'exprime par le sous-ensemble spécifique I_j^A des

états d'entrée $E_i(\underline{a})$ pour lesquels le signal de sortie ϕ_j doit prendre la valeur 1 :

$$A_j(\underline{a}) = \sum_{i \in I_j^A} E_i(\underline{a}) \quad \text{avec} \quad I_j^A = \{i \mid S_{ij}(f_j) = f_j\}$$

$$A_1(\underline{a}) = a_1 \bar{a}_2 a_3 \bar{a}_4 + a_1 a_2 a_3 \bar{a}_4 + a_2 \bar{a}_3 a_4 + a_1 \bar{a}_2 a_3 a_4 + \bar{a}_1 a_2 a_4$$

$$A_2(\underline{a}) = \bar{a}_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 \bar{a}_4 + a_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 a_4 + a_1 a_2 a_3 a_4$$

$$A_3(\underline{a}) = a_1 \bar{a}_2 a_3 \bar{a}_4 + a_1 a_2 a_3 a_4$$

$$\mathcal{R}_j^A(A_j(\underline{a}), MF_j^A(\underline{a}, \underline{y})) : A_j(\underline{a}) \leq MF_j^A(\underline{a}, \underline{y})$$

$$\iff \overline{R_j^A(\underline{a}, \underline{y})} = \overline{A_j(\underline{a})} + MF_j^A(\underline{a}, \underline{y})$$

Dans le but d'effectuer un calcul simultané de ces $|J|$ fonctions caractéristiques, il faut les réunir en une fonction composée; parmi les quatre possibilités à disposition, il faut choisir celle qu'il est possible de formuler sans décomposition à partir des données dont nous disposons, à savoir la fonction de vérité $V(\underline{a}, \underline{f})$ et la fonction $R^{B\pi}(\underline{a}, \underline{f})$; pour cette raison, nous formulerons l'inverse de la fonction composée en somme $\overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})}$:

$$\begin{aligned} \overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})} &= \prod_{j \in J} [\overline{R_j^A(\underline{a}, \underline{y})} + \bar{f}_j] \\ &= \prod_{j \in J} [A_j(\underline{a}) \cdot \overline{MF_j^A(\underline{a}, \underline{y})} + \bar{f}_j] \\ &= \prod_{j \in J} [(\sum_{i \in I_j^A} E_i(\underline{a})) \cdot \prod_{k \in K_j} \{\overline{P_k(\underline{a})} + \bar{y}_k\} + \bar{f}_j] \end{aligned}$$

On note la présence des sous-ensembles spécifiques I_j^A et K_j , qui rendraient la formulation telle quelle de cette expression extrêmement ardue. Par une succession de transformations, nous allons en rechercher une forme qui peut se déduire simplement des données à disposition; c'est d'ailleurs dans ce but que nous avons à nouveau utilisé les mêmes variables \underline{f} pour composer les $|J|$ fonctions caractéristiques $R_j^A(\underline{a}, \underline{y})$.

Formation directe de la fonction composée

Comme dans le cas de la recherche des impliquants premiers, il est judicieux de faire intervenir à ce stade les fonctions $S_{ij}^A(f_j)$, définies au chapitre III (§ 3), ainsi que des fonctions $S_{kj}(f_j)$, définies de façon analogue :

$$S_{ij}^A(f_j) = \begin{cases} f_j, & i \in I_j^A \\ \bar{f}_j, & i \in \bar{I}_j^A \end{cases} \quad S_{kj}(f_j) = \begin{cases} f_j, & j \in J_k \\ \bar{f}_j, & j \in \bar{J}_k \end{cases}$$

L'introduction des fonctions $S_{ij}^A(f_j)$ et $S_{kj}(f_j)$ permet d'étendre aux ensembles complets I et K les sous-ensembles spécifiques I_j^A et K_j (voir développement, fig. 5.3 ci-contre).

La fonction $V^A(\underline{a}, \underline{f})$ se dérive de la fonction de vérité $V(\underline{a}, \underline{f})$ en remplaçant dans chaque terme de celle-ci les variables f_j manquantes par des variables f_j niées, puis en omettant les variables \underline{f} non niées restantes; la fonction $MF^A(\underline{a}, \underline{y}, \underline{f})$ s'obtient par translation de la fonction $R^{B\pi}(\underline{a}, \underline{f})$ suivie de l'indexation de chacun des termes obtenus par une variable indépendante y_k :

$$MF^A(\underline{a}, \underline{y}, \underline{f}) = \sum_{k \in K} p_k(\underline{a}) \cdot y_k \cdot \prod_{j \in J_k} \bar{f}_j$$

$$\begin{aligned}
\overline{R^{A_1}(\underline{a}, \underline{y}, \underline{f})} &= \prod_{j \in J} \left[\left[\sum_{i \in I} E_i(\underline{a}) \cdot S_{ij}^A(f_j) \right] \cdot \left[\prod_{k \in K} (\overline{P_k(\underline{a})} + \overline{y_k} + \overline{S_{kj}(f_j)}) \right] + \overline{f_j} \right] \\
&= \prod_{j \in J} \left[\left[\sum_{i \in I} E_i(\underline{a}) \cdot (S_{ij}^A(f_j) + \overline{f_j}) \right] \cdot \left[\prod_{k \in K} (\overline{P_k(\underline{a})} + \overline{y_k} + \overline{S_{kj}(f_j)} + \overline{f_j}) \right] \right] \\
&= \left[\prod_{j \in J} \sum_{i \in I} E_i(\underline{a}) \cdot (S_{ij}^A(f_j) + \overline{f_j}) \right] \cdot \prod_{j \in J} \prod_{k \in K} \left[\overline{P_k(\underline{a})} + \overline{y_k} + \overline{S_{kj}(f_j)} + \overline{f_j} \right] \\
&= \left[\sum_{i \in I} \prod_{j \in J} E_i(\underline{a}) \cdot (S_{ij}^A(f_j) + \overline{f_j}) \right] \cdot \prod_{k \in K} \prod_{j \in J} \left[\overline{P_k(\underline{a})} + \overline{y_k} + \overline{S_{kj}(f_j)} + \overline{f_j} \right]
\end{aligned}$$

La somme et le produit peuvent être intervertis grâce à la condition de cohérence d'écriture de la table de vérité (démonstration dans l'appendice D):

$$\begin{aligned}
\overline{R^{A_2}(\underline{a}, \underline{y}, \underline{f})} &= \left[\sum_{i \in I} E_i(\underline{a}) \cdot \prod_{j \in J} (S_{ij}^A(f_j) + \overline{f_j}) \right] \prod_{k \in K} \left[\overline{P_k(\underline{a})} + \overline{y_k} + \prod_{j \in J} (\overline{S_{kj}(f_j)} + \overline{f_j}) \right] \\
&= V^A(\underline{a}, \underline{f}) \cdot MF^A(\underline{a}, \underline{y}, \underline{f})
\end{aligned}$$

Fig. 5.3 : transformations de la fonction composée en somme.

Exemple :

$$\begin{aligned}
 V^A(\underline{a}, \underline{f}) = & \bar{a}_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_3 + a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \\
 & + \bar{a}_1 \bar{a}_2 \bar{a}_4 \bar{f}_1 \bar{f}_3 + a_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_2 + \\
 & + \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 \bar{a}_4 \bar{f}_3 + \\
 & + a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_2 \bar{a}_3 a_4 \bar{f}_2 \bar{f}_3 + \\
 & + \bar{a}_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 a_4 \bar{f}_3 + \\
 & + \bar{a}_1 a_2 a_4 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 a_4 \bar{f}_1
 \end{aligned}$$

$$\begin{aligned}
 MF^A(\underline{a}, \underline{y}, \underline{f}) = & \bar{a}_1 \bar{a}_3 y_1 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 y_2 \bar{f}_2 + \bar{a}_1 y_3 \bar{f}_3 + \\
 & + \bar{a}_3 y_4 \bar{f}_3 + \bar{a}_2 \bar{a}_4 y_5 \bar{f}_1 \bar{f}_3 + a_2 a_4 y_6 \bar{f}_2 \bar{f}_3 + \\
 & + \bar{a}_2 \bar{a}_3 a_4 y_7 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \bar{a}_1 \bar{a}_2 \bar{a}_4 y_8 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_3 a_4 y_9 \bar{f}_2 + \\
 & + a_1 \bar{a}_2 a_3 y_{10} \bar{f}_1 + a_1 a_3 \bar{a}_4 y_{11} \bar{f}_1 + a_1 a_2 a_3 \bar{a}_4 y_{12} \bar{f}_1 \bar{f}_2 + \\
 & + \bar{a}_1 a_2 a_4 y_{13} \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_2 \bar{a}_3 a_4 y_{14} \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 a_4 y_{15} \bar{f}_1 \bar{f}_2
 \end{aligned}$$

La formation de $\overline{R^{\Lambda\Sigma}(\underline{a}, \underline{y}, \underline{f})}$ avec $V^A(\underline{a}, \underline{f})$ et $MF^A(\underline{a}, \underline{y}, \underline{f})$ peut ainsi se faire sans nécessiter de décomposition des données à disposition; de plus, on note la disparition des opérations doubles $\Pi\Sigma$ et $\Pi\Pi$, ce qui réduit le nombre des opérations à effectuer pour obtenir $\overline{R^{\Lambda\Sigma}(\underline{a}, \underline{y}, \underline{f})}$ sous forme disjonctive.

$$\begin{aligned}
 \overline{R^{\Lambda\Sigma}(\underline{a}, \underline{y}, \underline{f})} = & \bar{a}_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_2 \bar{a}_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \\
 & + \bar{a}_1 \bar{a}_2 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \\
 & + \bar{a}_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 \bar{a}_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \\
 & + a_2 \bar{a}_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \bar{a}_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + \\
 & + \bar{a}_1 a_2 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 a_4 \bar{f}_1 \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 \bar{a}_4 y_5 \bar{f}_1 \bar{f}_2 + \\
 & + a_1 a_2 a_3 a_4 y_6 \bar{f}_1 \bar{f}_2 + a_1 \bar{a}_2 \bar{a}_3 \bar{a}_4 y_7 \bar{f}_1 \bar{f}_3 + \bar{a}_1 \bar{a}_2 a_3 \bar{a}_4 y_8 \bar{f}_1 \bar{f}_3 + \\
 & + \bar{a}_2 \bar{a}_3 \bar{a}_4 y_1 y_7 y_8 \bar{f}_1 \bar{f}_3 + \bar{a}_1 \bar{a}_2 \bar{a}_4 y_1 y_7 y_8 \bar{f}_1 \bar{f}_3 + \\
 & + a_1 a_2 a_3 a_4 y_2 y_6 y_9 \bar{f}_1 + a_1 \bar{a}_2 a_3 \bar{a}_4 y_5 y_{10} y_{11} \bar{f}_2 + \\
 & + a_1 a_2 a_3 \bar{a}_4 y_{11} y_{12} \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 \bar{a}_4 y_2 y_{12} \bar{f}_1 \bar{f}_3 + \\
 & + \bar{a}_1 a_2 a_3 a_4 y_{13} \bar{f}_2 \bar{f}_3 + a_1 a_2 a_3 \bar{a}_4 y_2 y_{11} y_{12} \bar{f}_3 + \\
 & + a_1 a_2 \bar{a}_3 \bar{a}_4 y_{14} \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 a_4 y_9 y_{15} \bar{f}_1 \bar{f}_3 + \\
 & + a_2 \bar{a}_3 a_4 y_1 y_{13} y_{14} \bar{f}_2 \bar{f}_3 + \bar{a}_1 a_2 a_4 y_1 y_{13} y_{14} \bar{f}_2 \bar{f}_3 + \\
 & + a_1 \bar{a}_2 a_3 a_4 y_{10} y_{15} \bar{f}_2 \bar{f}_3 + a_1 \bar{a}_2 a_3 a_4 y_9 y_{10} y_{15} \bar{f}_3
 \end{aligned}$$

Obtention du résultat

Ecrivons le résultat du calcul de $\overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})}$ sous forme disjonctive (cf. exemple, p. ci-contre) :

$$\overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})} = \sum_{\ell \in L} H_{\ell}(\underline{a}) \cdot \prod_{k \in K_{\ell}} \bar{y}_k \cdot \prod_{j \in \bar{J}_{\ell}} \bar{f}_j$$

Il s'agit maintenant de transformer cette expression de façon à obtenir la forme complète de la fonction composée en produit, en vue de faire apparaître explicitement les conditions de couverture. A cette fin, il y a lieu d'appliquer successivement les opérateurs de translation \odot et d'inversion \ominus à la fonction-somme inverse $\overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})}$:

$$\overline{R^{A\pi}(\underline{a}, \underline{y}, \underline{f})} = \sum_{\ell \in L} H_{\ell}(\underline{a}) \cdot \prod_{k \in K_{\ell}} \bar{y}_k \cdot \prod_{j \in \bar{J}_{\ell}} \bar{f}_j$$

$$R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) = \prod_{\ell \in L} [H_{\ell}(\underline{a}) + \sum_{k \in K_{\ell}} y_k + \prod_{j \in \bar{J}_{\ell}} \bar{f}_j]$$

Il reste alors à effectuer les produits de cette dernière expression pour obtenir la forme complète cherchée.

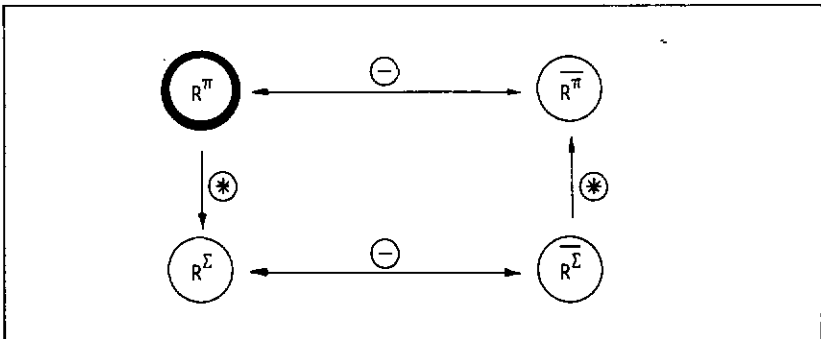


Fig. 5.4 : rappel du diagramme de transformation.

Interprétation du résultat

Considérons un impliquant de $R^{A\pi}(\underline{a}, \underline{y}, \underline{f})$ qui ne soit formé que des variables \underline{y} et \underline{f} ; par construction, si cet impliquant est premier, les variables \underline{y} qui y figurent sont toutes directes (non niées), alors que ses variables \underline{f} sont toutes niées.

Théorème 5.2

$$\prod_{k \in K'} y_k \cdot \prod_{j \in J'} \bar{f}_j \leq R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) \iff A_j(\underline{a}) \leq \sum_{k \in K' \cap K_j} P_k(\underline{a})$$

$$\forall j \in J'$$

Démonstration : par les théorèmes 2.3 et 4.4.

Si l'impliquant en question est premier, la couverture composée des impliquants $\{P_k(\underline{a}) \mid k \in K'\}$ est irrédondante, et elle ne peut suffire à couvrir aucun ensemble J'' de sous-systèmes individuels incluant strictement le sous-ensemble J' .

Dans le cas où l'impliquant de $R^{A\pi}(\underline{a}, \underline{y}, \underline{f})$ n'est composé que de variables \underline{y} , la couverture K' décrite est valable pour le système entier ($J' = J$); si un tel impliquant est premier, la couverture globale qui lui est associée est irrédondante.

L'existence d'impliquants de $R^{A\pi}(\underline{a}, \underline{y}, \underline{f})$ qui ne sont composés que de variables \underline{y} est assez simple à démontrer; elle se base sur l'hypothèse $A_j(\underline{a}) \leq B_j(\underline{a}) \quad \forall j \in J$, qui est évidemment vérifiée par la définition même de ces fonctions. Afin d'obtenir ces impliquants sans travail inutile, on élimine les variables \underline{a} et \underline{f} dans l'expression $R^{A\pi}(\underline{a}, \underline{y}, \underline{f})$ avant d'effectuer son inversion.

$$R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) = \prod_{\ell \in L} [\overline{H_{\ell}(\underline{a})}] + \sum_{k \in K_{\ell}} y_k + \prod_{j \in J_{\ell}} \overline{f_j}$$

$$R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) = \prod_{\ell \in L} [\sum_{k \in K_{\ell}} y_k] + \dots$$

$$R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) = y_5 \cdot y_6 \cdot y_7 \cdot y_8 \cdot (y_1 + y_7 + y_8) \cdot (y_1 + y_7 + y_8) \cdot \\ \cdot (y_2 + y_6 + y_9) \cdot (y_5 + y_{10} + y_{11}) \cdot (y_2 + y_{12}) \cdot \\ \cdot (y_{11} + y_{12}) \cdot (y_2 + y_{11} + y_{12}) \cdot y_{13} \cdot y_{14} \cdot (y_1 + y_{13} + y_{14}) \cdot \\ \cdot (y_1 + y_{13} + y_{14}) \cdot (y_9 + y_{15}) \cdot (y_{10} + y_{15}) \cdot (y_9 + y_{10} + y_{15}) + \dots$$

La fonction partielle ainsi obtenue est connue sous le nom de fonction de Petrick; on la trouve aussi représentée sous forme de tableau [10] - [13], [23], [27] - [29] (cf. fig. 5.5).

fonction	termes impli- quants	$\overline{a_1} \overline{a_2} \overline{a_3}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7} \overline{a_8}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7} \overline{a_8} \overline{a_9}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7} \overline{a_8} \overline{a_9} \overline{a_{10}}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7} \overline{a_8} \overline{a_9} \overline{a_{10}} \overline{a_{11}}$	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7} \overline{a_8} \overline{a_9} \overline{a_{10}} \overline{a_{11}} \overline{a_{12}}$
		1	$\overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5}$ $\overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6}$	x							
2	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5} \overline{a_6} \overline{a_7} \overline{a_8}$	x									
3	$\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4}$ $\overline{a_1} \overline{a_2} \overline{a_3} \overline{a_4} \overline{a_5}$										

Fig. 5.5 : tableau de couverture de Petrick.

Dans notre exemple, la fonction $R^{A\pi}(\underline{a}, \underline{y}, \underline{f})$ sous forme complète se compose de 339 impliquants premiers, dont 31 ne contiennent que des variables \underline{y} et \underline{f} , parmi lesquels 4 ne sont formés que de variables \underline{y} :

$$\begin{aligned} R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) = & y_5 y_6 y_7 y_8 y_{12} y_{13} y_{14} y_{15} + y_5 y_6 y_7 y_8 y_9 y_{10} y_{12} y_{13} y_{14} + \\ & + y_2 y_5 y_6 y_7 y_8 y_9 y_{10} y_{11} y_{13} y_{14} + y_2 y_5 y_6 y_7 y_8 y_{11} y_{13} y_{14} y_{15} + \\ & + y_5 y_6 \bar{f}_1 \bar{f}_2 + y_{10} y_{11} y_{13} y_{14} \bar{f}_2 \bar{f}_3 + y_2 y_7 y_8 y_9 \bar{f}_1 \bar{f}_3 + \\ & + y_5 y_6 y_{10} y_{11} y_{13} y_{14} \bar{f}_2 + \dots + \bar{a}_3 y_{14} \bar{f}_2 + \dots \end{aligned}$$

Ainsi, huit impliquants premiers sont nécessaires au minimum pour réaliser ce système sur deux niveaux (voir réalisations en portes et en PLA, fig. 6.4 et 6.5) :

$$F_1(\underline{a}) = \bar{a}_2 \bar{a}_4 + a_1 a_2 a_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 a_4 + a_2 \bar{a}_3 a_4 + a_1 \bar{a}_2 a_3 a_4$$

$$F_2(\underline{a}) = a_2 a_4 + \bar{a}_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 \bar{a}_4 + a_1 a_2 a_3 \bar{a}_4 + a_1 \bar{a}_2 a_3 a_4$$

$$F_3(\underline{a}) = \bar{a}_2 \bar{a}_4 + a_2 a_4$$

Les impliquants en \underline{y} et \underline{f} décrivent des couvertures partielles : $y_5 y_6 y_{10} y_{11} y_{13} y_{14} \bar{f}_2$ indique que l'ensemble des 6 impliquants correspondants est suffisant pour couvrir les sous-systèmes n° 1 et 3; on trouve parmi eux des couvertures individuelles ($y_{10} y_{11} y_{13} y_{14}$, $y_2 y_7 y_8 y_9$, $y_5 y_6$ resp. pour les sous-systèmes n° 1, 2 et 3).

Les impliquants en \underline{a} , \underline{y} et \underline{f} se trouvent en général en très grand nombre dans $R^{A\pi}(\underline{a}, \underline{y}, \underline{f})$; ils décrivent des couvertures conditionnelles. Ainsi, $\bar{a}_3 y_{14} \bar{f}_2$ signifie que l'impliquant $P_{14}(\underline{a})$ est suffisant pour réaliser les systèmes n° 1 et 3 pour toutes les configurations d'entrée vérifiant $\alpha_3 = 0$.

L'indication de l'attribution aux différents sous-systèmes des impliquants décrits par une couverture globale est fournie par la partie en \underline{f} qui accompagne ceux-ci dans la fonction $R^{B\pi}(\underline{a}, \underline{f})$; toutefois, cette

indication ne concerne que les incompatibilités : elle mentionne les sous-systèmes pour lesquels un impliquant donné $P_k(\underline{a})$ ne peut pas être pris. Cependant, chaque impliquant ne doit pas forcément entrer dans la composition de tous les sous-systèmes avec lesquels il est compatible. Résoudre exactement une couverture, c'est-à-dire rechercher pour chaque sous-système individuel, à partir d'une couverture globale donnée, le sous-ensemble des impliquants de cette couverture qui sont compatibles avec lui et qui sont nécessaires à sa réalisation, implique le calcul supplémentaire suivant, à effectuer (individuellement, cette fois) pour chacun des sous-systèmes :

$$\text{A partir de : } R^{B\pi}(\underline{a}, \underline{f}) = \sum_{k \in K} P_k(\underline{a}) \cdot \prod_{j \in J_k} \bar{f}_j$$

$$\text{et de : } \prod_{k \in K'} y_k \leq R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) \quad (\text{couverture sélectionnée})$$

$$\text{on forme : } MF_j^{A'}(\underline{a}, \underline{y}) = \sum_{k \in K'_j} P_k(\underline{a}) \cdot y_k$$

$$\text{avec : } K'_j = K' \cap K_j = \{k | k \in K' \text{ et } j \in J_k\}$$

$$\text{et on résout : } R_j^{A'}(\underline{a}, \underline{y}) = \overline{A_j(\underline{a})} + MF_j^{A'}(\underline{a}, \underline{y})$$

Toutefois, selon le genre de synthèse que l'on doit effectuer, cette dernière opération n'est pas toujours nécessaire; ainsi, dans le cas de la synthèse de PLA ("programmed logic array"), seul compte le nombre d'impliquants de la couverture globale, alors que le nombre de sous-systèmes pour la réalisation desquels ils peuvent ou doivent intervenir n'a aucune importance. Au contraire, dans le cas de la synthèse d'un système à réaliser avec des éléments discrets (portes SSI, etc.), cette étape supplémentaire permet de réduire au minimum le nombre des opérateurs du niveau de sortie. La durée de ce calcul additionnel est de toute façon très nettement inférieure à celui qui le précède directement

(inversion de la fonction $\overline{R^{A\pi}(\underline{a}, \underline{y}, \underline{f})}$, soit résolution du tableau de Petrick).

Le nombre de couvertures irrédondantes que l'on obtient est en général considérable; pour un problème de taille moyenne ou grande, le calcul de toutes ces couvertures est impossible à réaliser en un temps raisonnable. Il est alors important de disposer d'un algorithme possédant certains critères de sélection permettant de restreindre le calcul à quelques impliquants premiers de la fonction de Petrick dont on sait, grâce aux critères implantés, qu'ils correspondent à des couvertures quasi-minimales.

Un algorithme heuristique de ce type a été notamment développé par R. M. Bowman et al. [29].

3. Remarques

A) Pour synthétiser un système sur deux niveaux, trois applications de l'algorithme d'inversion ont été nécessaires; ces trois opérations correspondent à trois phases connues de la synthèse classique :

- 1) obtention des impliquants premiers :

$$R^{B\pi}(\underline{a}, \underline{f}) = \overline{V^{B^1}(\underline{a}, \underline{f})}$$

- 2) formation du tableau de couverture :

$$\overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})} = V^{A^1}(\underline{a}, \underline{f}) \cdot \overline{MF^A(\underline{a}, \underline{y}, \underline{f})}$$

- 3) résolution du tableau de couverture :

$$R^{A\pi}(\underline{a}, \underline{y}, \underline{f}) = \prod[\sum y_k] + \dots$$

Il faut souligner que l'inversion d'une expression est la seule opération qui nécessite un calcul et qui, par conséquent, prend du temps (formation des produits de variables, tests d'inclusion pour éliminer les impliquants non premiers); les autres opérations (formation de $V^A(\underline{a}, \underline{f})$ et $V^B(\underline{a}, \underline{f})$, translations) ne sont en fait que des manipulations de variables, effectuées immédiatement.

B) Les opérations d'inversion que nécessite la synthèse classique sont chacune précédée par une translation de la donnée (ou par une opération similaire, dans le cas de la première d'entre elles) :

$$1) \quad V(\underline{a}, \underline{f}) = \sum_I E_i(\underline{a}) \cdot S_i(\underline{f})$$

$$\xrightarrow{\sim \textcircled{*}} \quad V^B(\underline{a}, \underline{f}) = \sum_I E_i(\underline{a}) \cdot \overline{S_i(\underline{f})}$$

$$2) \quad R^{B\pi}(\underline{a}, \underline{f}) = \sum_K P_k(\underline{a}) \cdot \prod_{\overline{J}_k} \overline{f}_j$$

$$\xrightarrow{\begin{matrix} \textcircled{*} \\ + \text{indexation} \end{matrix}} \quad MF^A(\underline{a}, \underline{y}, \underline{f}) = \sum_K P_k(\underline{a}) \cdot y_k \cdot \overline{\prod_{\overline{J}_k} \overline{f}_j}$$

$$3) \quad \overline{R^{A\Sigma}(\underline{a}, \underline{y}, \underline{f})} = \sum_L H_\ell(\underline{a}) \cdot \prod_{K_\ell} \overline{y}_k \cdot \prod_{\overline{J}_\ell} \overline{f}_j$$

$$\xrightarrow{\textcircled{*}} \quad \overline{R^{A\pi}(\underline{a}, \underline{y}, \underline{f})} = \sum_L H_\ell(\underline{a}) \cdot \prod_{K_\ell} \overline{y}_k \cdot \overline{\prod_{\overline{J}_\ell} \overline{f}_j}$$

Cette propriété a permis à J.-J. Monbaron [55] de réunir en un seul algorithme la translation et l'inversion, et de réaliser ainsi les trois phases de la synthèse classique par la mise en oeuvre d'un algorithme unique, favorisant ainsi l'implantation de cette méthode sur des petites

machines. Une autre approche consiste à écrire trois versions du même algorithme, en veillant à adapter chaque version à la phase qu'elle doit traiter, en vue d'éliminer de chaque phase toutes les opérations superflues; la troisième phase, notamment, doit pouvoir être exécutée de façon approchée (heuristique), pour qu'il soit possible d'obtenir une couverture quasi-minimale en un temps raisonnable, dans le cas de systèmes à synthétiser de taille moyenne à grande; cette façon de procéder, destinée à de plus grands ordinateurs, fournit des algorithmes rapides et puissants.

Un ensemble de programmes d'ordinateur pour effectuer la synthèse de grands systèmes non canoniques faiblement définis a été développé sur la base des méthodes présentées dans ce chapitre [56].

VI. AUTRES APPLICATIONS DU FORMALISME ALGÈBRE

Nous avons cité en exemple l'application du formalisme algébrique à la synthèse classique de systèmes logiques combinatoires sur deux niveaux.

Ce formalisme peut également s'appliquer à des méthodes de synthèse plus générales. Dans ce chapitre, nous aborderons la généralisation de la notion d'impliquant premier en composant premier, puis en composant mixte; nous mentionnerons ensuite deux méthodes nouvelles de synthèse qui utilisent ces composants : la synthèse approchée et la synthèse à quatre niveaux des systèmes.

1. Composants premiers

La synthèse classique d'un système fait intervenir la recherche de ses impliquants premiers; ceux-ci sont en fait les éléments de base qui servent à sa construction. Un impliquant premier d'un sous-système donné est un élément de ce sous-système qui a la propriété d'être le plus grand possible, sans toutefois avoir une intersection avec l'un des zéros de ce sous-système.

En abandonnant cette dernière restriction, on aboutit à la notion de composant premier : ces composants sont les plus grands éléments d'un sous-système qui ont une intersection avec un minimum de ses zéros; lorsque ce minimum est nul, le composant premier redevient un simple impliquant premier du sous-système en question. Un composant est premier si tout composant qui l'inclut couvre plus de zéros que lui. L'obtention des composants premiers d'un système est assez semblable à celle de ses impliquants premiers; en effet, un impliquant premier est un terme $P_k(\underline{a})$

justifiant d'une intersection nulle avec une fonction $B_j(\underline{a})$ qui décrit les zéros d'un sous-système :

$$P_k(\underline{a}) \cdot B_j(\underline{a}) = 0$$

On calcule alors la fonction caractéristique simultanée :

$$R^{B\pi}(\underline{a}, \underline{f}) = \prod_{j \in J} [\overline{B_j(\underline{a})} + \overline{f_j}]$$

Un composant premier $P_k(\underline{a})$ est un terme qui possède une intersection nulle avec un certain nombre de zéros $Z_{ij}(\underline{a})$ du système :

$$P_k(\underline{a}) \cdot Z_{ij}(\underline{a}) = 0 \quad \text{avec} \quad Z_{ij}(\underline{a}) = \begin{cases} E_i(\underline{a}), & S_{ij}(f_j) = \overline{f_j} \\ 0 & \text{autrement.} \end{cases}$$

$$R^{Z\pi}(\underline{a}, \underline{f}) = \prod_{i \in I, j \in J} [\overline{Z_{ij}(\underline{a})} + \overline{f_{ij}}]$$

Calcul à partir de la table de vérité

Pour composer les $|I| \times |J|$ fonctions caractéristiques, on a recours à un jeu de variables booléennes à deux indices $\underline{f} = \{f_{ij} \mid i \in I, j \in J\}$. Il est possible d'exprimer la nouvelle fonction $R^{Z\pi}(\underline{a}, \underline{f})$ directement à partir de la fonction de vérité $V(\underline{a}, \underline{f})$:

$$R^{Z\pi}(\underline{a}, \underline{f}) = \prod_{i \in I} [\overline{E_i(\underline{a})} + \prod_{j \in J} S_{ij}^Z(f_{ij})]$$

$$\text{avec} \quad S_{ij}^Z(f_{ij}) = \begin{cases} 1 \\ 1 \\ \overline{f_{ij}} \end{cases} \quad \text{si} \quad S_{ij}(f_j) = \begin{cases} f_j \\ 1 \\ \overline{f_j} \end{cases}$$

Exemple : (table de vérité, voir fig. 5.1; comparer avec l'exemple cité en p. 53).

$$\begin{aligned}
 R^Z_{\Pi}(\underline{a}, \underline{f}) &= (\bar{a}_1 + \bar{a}_2 + a_3 + a_4 + \bar{f}_{2.1} \bar{f}_{2.2}) \cdot \\
 &\cdot (\bar{a}_1 + a_2 + \bar{a}_3 + a_4 + \bar{f}_{4.2}) \cdot \\
 &\cdot (a_1 + \bar{a}_2 + \bar{a}_3 + a_4 + \bar{f}_{5.1} \bar{f}_{5.2}) \cdot \\
 &\cdot (\bar{a}_1 + \bar{a}_2 + \bar{a}_3 + a_4 + \bar{f}_{6.3}) \cdot \\
 &\cdot (\bar{a}_1 + a_2 + a_3 + \bar{a}_4 + \bar{f}_{7.1} \bar{f}_{7.2}) \cdot \\
 &\cdot (a_1 + a_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_{9.1} \bar{f}_{9.2}) \cdot \\
 &\cdot (\bar{a}_1 + a_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_{10.3}) \cdot \\
 &\cdot (\bar{a}_1 + \bar{a}_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_{12.1})
 \end{aligned}$$

$$\begin{aligned}
 R^Z_{\Pi}(\underline{a}, \underline{f}) &= \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{4.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} \bar{f}_{12.1} + \\
 &+ a_4 \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} \bar{f}_{12.1} + \bar{a}_3 a_4 \bar{f}_{7.1} \bar{f}_{7.2} + \\
 &+ \bar{a}_4 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{4.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} + \bar{a}_3 \bar{a}_4 \bar{f}_{2.1} \bar{f}_{2.2} + \\
 &+ a_3 \bar{f}_{4.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} \bar{f}_{12.1} + a_2 \bar{a}_3 \bar{f}_{2.1} \bar{f}_{2.2} + \\
 &+ a_1 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{4.2} \bar{f}_{6.3} \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{10.3} \bar{f}_{12.1} + \bar{a}_1 a_2 \bar{f}_{5.1} \bar{f}_{5.2} + \\
 &+ a_3 a_4 \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} \bar{f}_{12.1} + a_3 \bar{a}_4 \bar{f}_{4.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} + \\
 &+ a_2 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} \bar{f}_{12.1} + a_2 a_3 \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} \bar{f}_{12.1} + \\
 &+ a_2 \bar{a}_4 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} + a_2 a_3 \bar{a}_4 \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{6.3} + \\
 &+ \bar{a}_2 \bar{f}_{4.2} \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} + \bar{a}_2 a_3 a_4 \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} + \\
 &+ \bar{a}_2 a_4 \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} + a_1 a_2 \bar{a}_4 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{6.3} + \\
 &+ \bar{a}_2 a_3 \bar{f}_{4.2} \bar{f}_{9.1} \bar{f}_{9.2} \bar{f}_{10.3} + a_1 a_2 a_3 \bar{a}_4 \bar{f}_{6.3} + \bar{a}_2 \bar{a}_3 \bar{f}_{7.1} \bar{f}_{7.2} + \\
 &+ a_1 a_4 \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{10.3} \bar{f}_{12.1} + a_1 \bar{a}_2 a_3 a_4 \bar{f}_{10.3} + \bar{a}_1 \bar{a}_2 \bar{f}_{9.1} \bar{f}_{9.2} + \\
 &+ a_1 \bar{a}_4 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{4.2} \bar{f}_{6.3} + a_1 a_3 a_4 \bar{f}_{10.3} \bar{f}_{12.1} + \bar{a}_1 a_4 \bar{f}_{9.1} \bar{f}_{9.2} + \\
 &+ a_1 a_3 \bar{f}_{4.2} \bar{f}_{6.3} \bar{f}_{10.3} \bar{f}_{12.1} + a_1 a_2 a_3 \bar{f}_{6.3} \bar{f}_{12.1} + \bar{a}_1 \bar{a}_4 \bar{f}_{5.1} \bar{f}_{5.2} + \\
 &+ a_1 a_2 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{6.3} \bar{f}_{12.1} + a_1 \bar{a}_2 a_3 \bar{f}_{4.2} \bar{f}_{10.3} + a_2 a_4 \bar{f}_{12.1} + \\
 &+ a_1 \bar{a}_2 \bar{f}_{4.2} \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{10.3} + a_1 a_3 \bar{a}_4 \bar{f}_{4.2} \bar{f}_{6.3} + \bar{a}_2 \bar{a}_4 \bar{f}_{4.2} + \\
 &+ a_1 \bar{a}_2 a_4 \bar{f}_{7.1} \bar{f}_{7.2} \bar{f}_{10.3} + \bar{a}_3 \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{7.1} \bar{f}_{7.2} + \bar{a}_2 \bar{a}_3 a_4 + \\
 &+ \bar{a}_1 \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{9.1} \bar{f}_{9.2} + \bar{a}_1 \bar{a}_3 + \bar{a}_2 \bar{a}_3 \bar{a}_4 + \bar{a}_1 \bar{a}_2 \bar{a}_4 + \bar{a}_1 a_2 a_4
 \end{aligned}$$

Les impliquants de $R^Z_{\Pi}(\underline{a}, \underline{f})$ obtenus sont composés d'une partie $P_k(\underline{a})$ exprimée au moyen des variables d'entrée \underline{a} (c'est le composant premier), et d'un produit de variables f_{ij} niées; chacune de ces variables f_{ij}

décrit un zéro $Z_{ij}(\underline{a})$ (celui du sous-système j spécifié à la ligne i de la table de vérité) avec lequel le composant $P_k(\underline{a})$ a une intersection (interprétation par le théorème 4.4).

Exemple : $a_2 a_4 \bar{f}_{12.1} \leq R^{Z\pi}(\underline{a}, f)$ indique que le composant $a_2 a_4$ possède une intersection avec un zéro : celui du sous-système n° 1 décrit à la ligne n° 12 de la table de vérité; ce composant est donc un vrai impliquant pour les sous-systèmes n° 2 et 3 (il correspond à l'impliquant $a_2 a_4 \bar{f}_1$ de la fonction $R^{B\pi}(\underline{a}, f)$).

$a_1 a_3 \bar{f}_{4.2} \bar{f}_{6.3} \bar{f}_{10.3} \bar{f}_{12.1} \leq R^{Z\pi}(\underline{a}, f)$ indique que le composant $a_1 a_3$ possède une intersection avec 4 zéros du système; de plus, il n'est un vrai impliquant d'aucun sous-système, puisqu'il a une intersection avec au moins un zéro de chacun d'eux (on n'en trouve d'ailleurs pas trace dans la fonction $R^{B\pi}(\underline{a}, f)$).

La primauté de l'impliquant de $R^{Z\pi}(\underline{a}, f)$ garantit le caractère optimal du compromis entre la taille du composant et le nombre de zéros englobés (cf. propriété 4.5, p. 39).

2. Synthèse approchée de systèmes

La synthèse approchée de systèmes tolère un certain nombre d'erreurs de réalisation et en tire parti pour obtenir une réduction plus substantielle des circuits à réaliser; un tel procédé n'est bien sûr applicable que dans le cas où une réalisation exacte du système n'est pas absolument nécessaire.

Fonction de couverture généralisée

La synthèse approchée s'effectue en indexant les $|K|$ composants premiers obtenus avec un jeu de variables $\underline{y} = \{y_k | k \in K\}$, puis en

formant une équation de Petrick modifiée avec ces variables d'indice. Cette fonction $R^{Y\pi}(\underline{y}, \underline{f})$ est un produit d'expressions disjonctives qui décrivent chacune la couverture d'un "1" du système.

$$R^{Y\pi}(\underline{y}, \underline{f}) = \prod R_{i,j}^Y(\underline{y}, \underline{f})$$

Considérons le "1" du sous-système j décrit à la ligne i de la table de vérité; la fonction $R_{i,j}^Y(\underline{y}, \underline{f})$ qui le décrit possède trois sortes d'impliquants :

a) les variables y_k associées aux composants $P_k(\underline{a})$ qui n'ont pas d'intersection avec les zéros $Z_{i,j}(\underline{a})$ du sous-système j auquel appartient le "1" à couvrir (ces composants sont donc les vrais impliquants de ce sous-système);

b) une variable indépendante $f_{i,j}$ non niée qui, par sa présence dans la fonction $R_{i,j}^Y(\underline{y}, \underline{f})$, permet l'omission du "1" concerné lors de la recherche d'une couverture globale, et

c) des termes $y_k \cdot \bar{f}_{i,j}$ composés d'une variable y_k décrivant un composant premier $P_k(\underline{a})$ ayant une intersection non nulle avec un ou plusieurs zéros $Z_{i,j}(\underline{a})$ du sous-système individuel j , accompagnée d'une variable $f_{i,j}$ niée par "0" de ce sous-système avec lequel ce composant possède une intersection non nulle.

Remarque : dans le cas de la synthèse classique, chaque produit de la fonction de Petrick ordinaire est composé des variables décrites sous a); la fonction de Petrick modifiée ci-dessus est donc une généralisation de la fonction utilisée dans la méthode classique.

Exemple : la fonction $R_{4,1}^Y(\underline{y}, \underline{f})$ associée au "1" du sous-système n° 1 décrit à la ligne n° 4 de la table de vérité se compose des termes suivants (les indices des variables y_k correspondent à l'ordre dans lequel apparaissent les impliquants de $R^{Z\pi}(\underline{a}, \underline{f})$ dans l'exemple p. 71) :

a) $y_{33} + y_{36} + y_{37}$, trois variables qui correspondent aux trois vrais impliquants du système n° 1 et qui couvrent le "1" n° 4.1;

b) $f_{4.1}$, variable caractéristique de "1" n° 4.1 à couvrir;

c) $y_1 \bar{f}_{2.1} \bar{f}_{5.1} \bar{f}_{7.1} \bar{f}_{9.1} \bar{f}_{12.1} + y_4 \bar{f}_{2.1} \bar{f}_{5.1} +$
 $+ y_6 \bar{f}_{5.1} \bar{f}_{9.1} \bar{f}_{12.1} + y_8 \bar{f}_{2.1} \bar{f}_{7.1} \bar{f}_{12.1} + y_{11} \bar{f}_{5.1} +$
 $+ y_{16} \bar{f}_{7.1} \bar{f}_{9.1} + y_{20} \bar{f}_{9.1} + y_{26} \bar{f}_{2.1} + y_{29} \bar{f}_{12.1} +$
 $+ y_{35} \bar{f}_{7.1}$, termes décrivant chacun un composant dont

l'acceptation pour la réalisation du "1" en question entraînerait des erreurs, signalées par la présence des variables $\bar{f}_{i,j}$ respectives; ainsi $y_{11} \bar{f}_{5.1}$ décrit le composant $a_3 \bar{a}_4$ qui couvre le "0" du sous-système n° 1 décrit à la ligne n° 5 de la table de vérité.

Le système dont nous essayons de faire la synthèse approchée compte 14 fonctions partielles $R_{ij}^Y(\underline{y}, \underline{f})$ qu'il faut formuler selon le modèle ci-dessus (les "1" du système qui ont une intersection avec un composant sans être couverts par lui doivent être décomposés).

Interprétation des impliquants obtenus

En effectuant les produits $\prod R_{ij}^Y(\underline{y}, \underline{f})$, on obtient la forme complète de la fonction $R^Y(\underline{y}, \underline{f})$; ses impliquants premiers sont composés d'un produit de variables y_k qui, par construction, ne sont pas niées, d'un produit de variables f_{ij} non niées, et d'un produit de variables \bar{f}_{ij} niées :

$$\prod y_k \cdot \prod f_{ij} \cdot \prod \bar{f}_{ij} \leq R^Y(\underline{y}, \underline{f})$$

Ces trois parties de chaque impliquant décrivent respectivement :

a) $\prod y_k$ représente une pseudo-couverture, c'est-à-dire un ensemble de composants $P_k(\underline{a})$ qui suffisent à synthétiser le système

avec un certain nombre d'erreurs de réalisation;

b) $\prod f_{ij}$ décrit l'ensemble des "1" du système que la pseudo-couverture mentionnée sous a) oublie de couvrir (erreurs commises par défaut);

c) $\prod \bar{f}_{ij}$ décrit l'ensemble des "0" du système que la pseudo-couverture mentionnée sous a) couvre indûment (erreurs commises par excès).

Remarque : les impliquants de $R^{Y\pi}(\underline{y}, \underline{f})$ qui ne contiennent pas de variables f_{ij} non niées décrivent des pseudo-couvertures qui ne commettent que des erreurs par excès; de même, les impliquants de $R^{Y\pi}(\underline{y}, \underline{f})$ qui ne contiennent pas de variables f_{ij} niées décrivent des pseudo-couvertures qui ne commettent que des erreurs par défaut; finalement, les impliquants de $R^{Y\pi}(\underline{y}, \underline{f})$ qui ne contiennent que des variables \underline{y} décrivent des couvertures du système au vrai sens du terme.

Dans l'exemple cité, on obtient entre autres l'impliquant :

$$y_{29} y_{34} y_{37} \bar{f}_{12 \cdot 1} \bar{f}_{4 \cdot 2} \leq R^{Y\pi}(\underline{y}, \underline{f})$$

qui décrit un ensemble de trois composants permettant la réalisation du système entier avec deux erreurs seulement (alors que 8 impliquants sont nécessaires au minimum pour réaliser ce système sur deux niveaux, sans erreurs) :

$$F_1(\underline{a}) = a_1 a_3 + a_2 a_4 \quad (\text{couvre } a_1 a_2 a_3 a_4)$$

$$F_2(\underline{a}) = \bar{a}_2 \bar{a}_4 + a_1 a_3 \quad (\text{couvre } a_1 \bar{a}_2 a_3 \bar{a}_4)$$

$$F_3(\underline{a}) = \bar{a}_2 \bar{a}_4 + a_2 a_4 \quad (\text{exact})$$

(exemple de réalisation en PLA : voir fig. 6.5, p. 88).

Réduction du nombre de solutions obtenues

Le nombre des impliquants de $R^{Y\pi}(\underline{y}, \underline{f})$ est considérable, déjà pour des systèmes de faible taille; aussi, il est impensable de les calculer tous. Par ailleurs, seule une faible proportion de ces impliquants décrit des solutions vraiment intéressantes pour la réalisation approchée d'un système; l'algorithme chargé de calculer les impliquants de cette fonction doit donc être capable d'éliminer en cours de processus tout impliquant fournissant une solution inintéressante.

Deux méthodes peuvent être utilisées à cette fin : l'une agit au niveau de la formation des produits $R_{ij}^Y(\underline{y}, \underline{f})$ qui composent la fonction $R^{Y\pi}(\underline{y}, \underline{f})$ dans son expression initiale, l'autre agit au cours du calcul de cette expression sous forme disjonctive.

A) Lors de la formation de chaque fonction $R_{ij}^Y(\underline{y}, \underline{f})$, on élimine d'emblée les variables y_k qui décrivent des composants dont l'acceptation entraînerait un nombre de fautes excessif; deux limites peuvent être fixées concurremment :

- le nombre de fautes n (représenté par le nombre de variables f_{ij} niées qui accompagnent la variable y_k dans la fonction $R_{ij}^Y(\underline{y}, \underline{f})$) doit être inférieur à un nombre fixé N_0 (critère absolu);

- le nombre de fautes n doit être inférieur à un nombre N_k qui dépend de la taille du composant $P_k(\underline{a})$ (critère relatif).

B) Lors du calcul de $R^{Y\pi}(\underline{y}, \underline{f})$, on élimine de chaque résultat intermédiaire tous les impliquants qui contiennent un nombre de variables f_{ij} supérieur à un nombre fixé à l'avance, correspondant au nombre d'erreurs qui est tolérable pour la réalisation de ce système; ce critère peut d'ailleurs être appliqué séparément pour les deux polarités des variables f_{ij} ; on élimine également de chaque résultat partiel tout impliquant dont le nombre de variables y_k est supérieur ou égal à celui

de la meilleure couverture exacte (que l'on a préalablement calculée avec la méthode classique).

3. Synthèse multi-niveaux

Il existe plusieurs approches de la synthèse de systèmes sur plus de deux niveaux de portes [1], [2], [40], [41], [44], [50] - [53]; la plupart des travaux sur ce sujet préconisent les essais systématiques de factorisation des impliquants premiers. Toutefois, le formalisme algébrique nous ouvre une voie originale que nous allons esquisser dans ce paragraphe.

Cette méthode fait intervenir la recherche simultanée des composants premiers d'un système et de ceux du système inverse ("1" et "0" permutés); on synthétise alors le système direct avec les composants directs, en tolérant des erreurs par excès seulement, puis on utilise les composants du système inverse pour fabriquer des fonctions de masque qui corrigent les erreurs commises.

Recherche des composants mixtes

La recherche simultanée des composants premiers directs et inversés ne diffère que peu de celle des composants premiers, exposée au § 1 de ce chapitre.

Un composant du système direct est un terme $P_k(\underline{a})$ qui possède une intersection non nulle avec un certain nombre de zéros $Z_{ij}(\underline{a})$ du système; un composant du système inverse est un terme qui possède une intersection non nulle avec un certain nombre de "1" du système. Comme les ensembles de "1" et de "0" sont disjoints, il est possible de calculer en une seule passe tous les éléments $P_k(\underline{a})$ qui justifient d'une intersection nulle avec un certain nombre d'éléments $Z'_{ij}(\underline{a})$ du

système, éléments qui appartiennent aux ensembles réunis des "1" et des "0" du système :

$$P_k(\underline{a}) \cdot Z'_{ij}(\underline{a}) = 0 \quad \text{où} \quad Z'_{ij}(\underline{a}) = \begin{cases} E_i(\underline{a}), & S_{ij}(f_j) = f_j \text{ ou } \bar{f}_j \\ 0 & \text{autrement.} \end{cases}$$

On forme alors la fonction caractéristique simultanée :

$$R^{Z'}(\underline{a}, \underline{f}) = \prod_{i \in I, j \in J} [\overline{Z'_{ij}(\underline{a})} + S_{ij}^{Z'}(f_{ij})]$$

Pour composer les $|I| \times |J|$ fonctions caractéristiques, on utilise un jeu de variables booléennes à deux indices $\underline{f} = \{f_{ij} | i \in I, j \in J\}$; de plus, on utilise une polarité différente de ces variables si l'élément $Z'_{ij}(\underline{a})$ décrit est un "1" ou un "0" (cette distinction n'est pas indispensable; elle a seulement pour but de rendre le résultat plus simple à interpréter) :

$$S_{ij}^{Z'}(f_{ij}) = \begin{cases} f_{ij} \\ 1 \\ \bar{f}_{ij} \end{cases} \quad \text{si} \quad S_{ij}(f_j) = \begin{cases} f_j \\ 1 \\ \bar{f}_j \end{cases}$$

La fonction $R^{Z'}(\underline{a}, \underline{f})$ peut se former directement à partir de la table de vérité :

$$R^{Z'}(\underline{a}, \underline{f}) = \prod_{i \in I} [\overline{E_i(\underline{a})} + \prod_{j \in J} S_{ij}^{Z'}(f_{ij})]$$

Les impliquants premiers de la fonction $R^{Z'}(\underline{a}, \underline{f})$ ont la forme suivante (cf. exemple, p. ci-contre) :

$$P_k(\underline{a}) \cdot \prod f_{ij} \cdot \prod \bar{f}_{ij} \leq R^{Z'}(\underline{a}, \underline{f})$$

Exemple : (table de vérité : voir fig. 5.1; comparer avec les exemples cités aux pp. 53 et 71).

$$\begin{aligned}
 R^{Z^1\pi}(\underline{a}, \underline{f}) = & (\bar{a}_2 + \bar{a}_3 + \bar{a}_4 + f_{1.2}) \\
 & (a_1 + a_2 + \bar{a}_3 + \bar{a}_4 + \bar{f}_{2.1} \bar{f}_{2.2}) \\
 & (\bar{a}_1 + \bar{a}_2 + \bar{a}_4 + f_{3.2}) \\
 & (a_1 + \bar{a}_2 + a_3 + \bar{a}_4 + f_{4.1} \bar{f}_{4.2} f_{4.3}) \\
 & (\bar{a}_1 + a_2 + a_3 + \bar{a}_4 + \bar{f}_{5.1} \bar{f}_{5.2}) \\
 & (a_1 + a_2 + a_3 + \bar{a}_4 + f_{6.1} f_{6.2} \bar{f}_{6.3}) \\
 & (a_1 + \bar{a}_2 + \bar{a}_3 + a_4 + \bar{f}_{7.1} \bar{f}_{7.2}) \\
 & (a_2 + \bar{a}_3 + a_4 + f_{8.1}) \\
 & (\bar{a}_1 + \bar{a}_2 + a_3 + a_4 + \bar{f}_{9.1} \bar{f}_{9.2}) \\
 & (a_1 + \bar{a}_2 + a_3 + a_4 + f_{10.1} f_{10.2} \bar{f}_{10.3}) \\
 & (\bar{a}_1 + a_2 + a_4 + f_{11.1}) \\
 & (a_1 + a_2 + a_3 + a_4 + \bar{f}_{12.1} f_{12.2} f_{12.3})
 \end{aligned}$$

$$\begin{aligned}
 R^{Z^1\pi}(\underline{a}, \underline{f}) = & a_2 a_4 f_{8.1} f_{11.1} \bar{f}_{12.1} f_{12.2} f_{12.3} & + \\
 & + a_2 a_3 \bar{f}_{5.1} \bar{f}_{5.2} f_{6.1} f_{6.2} \bar{f}_{6.3} f_{11.1} \bar{f}_{12.1} f_{12.2} f_{12.3} & + \\
 + & \bar{a}_2 \bar{a}_4 f_{1.2} f_{3.2} f_{4.1} \bar{f}_{4.2} f_{4.3} & + \\
 + & \bar{a}_2 a_3 f_{3.2} f_{4.1} \bar{f}_{4.2} f_{4.3} \bar{f}_{9.1} \bar{f}_{9.2} f_{10.1} f_{10.2} \bar{f}_{10.3} & + \\
 + & a_1 a_4 \bar{f}_{7.1} \bar{f}_{7.2} f_{8.1} f_{10.1} f_{10.2} \bar{f}_{10.3} \bar{f}_{12.1} f_{12.2} f_{12.3} & + \\
 + & a_1 \bar{a}_4 f_{1.2} \bar{f}_{2.1} \bar{f}_{2.2} f_{4.1} \bar{f}_{4.2} f_{4.3} f_{6.1} f_{6.2} \bar{f}_{6.3} & + \\
 + & a_1 a_3 f_{4.1} \bar{f}_{4.2} f_{4.3} f_{6.1} f_{6.2} \bar{f}_{6.3} f_{10.1} f_{10.2} \bar{f}_{10.3} \bar{f}_{12.1} f_{12.2} f_{12.3} & + \\
 + & a_1 \bar{a}_3 f_{1.2} \bar{f}_{2.1} \bar{f}_{2.2} \bar{f}_{7.1} \bar{f}_{7.2} f_{8.1} & + a_1 a_2 a_3 a_4 \bar{f}_{12.1} f_{12.2} f_{12.3} & + \\
 + & a_1 a_2 a_3 \bar{a}_4 f_{6.1} f_{6.2} \bar{f}_{6.3} & + a_1 \bar{a}_2 a_3 a_4 f_{10.1} f_{10.2} \bar{f}_{10.3} & + \\
 + & a_1 \bar{a}_2 a_3 \bar{a}_4 f_{4.1} \bar{f}_{4.2} f_{4.3} & + \bar{a}_1 a_3 f_{3.2} \bar{f}_{5.1} \bar{f}_{5.2} \bar{f}_{9.1} \bar{f}_{9.2} f_{11.1} & + \\
 + & \bar{a}_1 \bar{a}_4 f_{1.2} f_{3.2} \bar{f}_{5.1} \bar{f}_{5.2} & + \bar{a}_1 a_4 f_{8.1} \bar{f}_{9.1} \bar{f}_{9.2} f_{11.1} & + \\
 + & \bar{a}_2 \bar{a}_3 f_{1.2} f_{3.2} \bar{f}_{7.1} \bar{f}_{7.2} & + a_2 \bar{a}_3 \bar{f}_{2.1} \bar{f}_{2.2} f_{8.1} f_{11.1} & + \\
 + & \dots
 \end{aligned}$$

(la fonction $R^{Z^1\pi}(\underline{a}, \underline{f})$ compte 75 impliquants premiers).

Chaque impliquant premier de $R^{Z^1\pi}(\underline{a}, \underline{f})$ décrit un composant mixte $P_k(\underline{a})$ qui possède une intersection avec les "1" du système signalés

par les variables f_{ij} non niées qui le composent, ainsi qu'avec les "0" indiqués par ses variables f_{ij} niées.

Si un composant mixte $P_k(\underline{a})$ n'est accompagné d'aucune variable f_{ij} , niée telle que $j' = j_0$, ce composant est un impliquant au sens vrai du sous-système individuel j_0 ; si un composant mixte $P_k(\underline{a})$ n'est accompagné d'aucune variable f_{ij} , non niée telle que $j' = j_0$, ce composant est un impliquant de l'inverse du système individuel j_0 .

Synthèse à quatre niveaux avec Les composants mixtes

Les composants mixtes peuvent être utilisés pour synthétiser un système sur quatre niveaux; la synthèse s'effectue par la formation d'une fonction de Petrick modifiée, comme dans le cas de la synthèse approchée.

Dans ce cas, les composants requis pour la construction du système sont indexés par un jeu de variables booléennes indépendantes à deux indices $\underline{y} = \{y_{k\ell} | k \in K, \ell \in L\}$, où K représente l'ensemble des composants mixtes $P_k(\underline{a})$ et ℓ l'indice de la version de chaque composant; en effet, lorsqu'un composant est requis pour la synthèse d'un sous-système donné et qu'il n'est pas un vrai impliquant de ce sous-système, il doit être restreint par une fonction de masque; plusieurs versions distinctes de ce composant sont alors nécessaires lorsqu'il doit intervenir plusieurs fois avec des fonctions de masque incompatibles.

La fonction de Petrick modifiée $R^{Y'}(\underline{y})$ se compose d'un produit de fonctions partielles $R_{\ell}^{Y'}(\underline{y})$ qui décrivent chacune la couverture d'un "1" du système.

La fonction partielle $R_{\ell}^{Y'}(\underline{y})$ associée au "1" (i_0, j_0) est composée d'une somme de termes $y_{k\ell} \cdot Y_{k\ell}(\underline{y})$ dans lesquels $y_{k\ell}$ est la variable associée à la version ℓ d'un composant $P_k(\underline{a})$ qui couvre le "1" (i_0, j_0)

en question, et $Y_{k\ell}(\underline{y})$ est une fonction des mêmes variables \underline{y} , définie ci-dessous (il s'agit en fait de la fonction de couverture d'une fonction de masque pour le composant $P_k(\underline{a})$).

A chaque version d'un composant mixte, décrite par une variable $y_{k\ell}$ dans la fonction de couverture $R^{Y^*\pi}(\underline{y})$, on associe un système logique combinatoire à sortie unique $U_{k\ell}(\underline{a}, y_{k\ell})$ défini comme suit :

- les variables d'entrée \underline{a} de ce système sont les mêmes que celles du système à synthétiser,

- le signal de sortie $u_{k\ell}$ vaut 0 pour l'état d'entrée $E_{i_0}(\underline{a})$ correspondant au "1" (i_0, j_0) du système que la version ℓ du composant $P_k(\underline{a})$ est appelée à couvrir,

- le signal de sortie $u_{k\ell}$ vaut 1 pour tous les états d'entrée $E_i(\underline{a})$ correspondant aux zéros du sous-système individuel j_0 avec lesquels le composant $P_k(\underline{a})$ possède une intersection non nulle.

Toute fonction $M_{k\ell}(\underline{a})$ qui satisfait la table de vérité du système $U_{k\ell}(\underline{a}, y_{k\ell})$ est une fonction de masque susceptible de transformer le composant $P_k(\underline{a})$ en un vrai impliquant du sous-système individuel j_0 pour lequel il est requis; la fonction $M_{k\ell}(\underline{a})$ est directe (1 = le composant est masqué, 0 = le composant est actif); cette fonction est constructible avec les impliquants de l'inverse du sous-système individuel j_0 .

Les fonctions $Y_{k\ell}(\underline{y})$ qui entrent dans la composition de la fonction $R^{Y^*\pi}(\underline{y})$ sont les fonctions de couverture des systèmes $U_{k\ell}(\underline{a}, y_{k\ell})$.

Exemple : (les indices $k \in K$ des variables $y_{k\ell}$ correspondent à l'ordre dans lequel les composants $P_k(\underline{a})$ apparaissent dans la fonction $R^{Z^*\pi}(\underline{a}, \underline{f})$ citée en exemple à la p. 79; les systèmes $U_{k\ell}(\underline{a}, y_{k\ell})$ associés à ces variables sont présentés sur la fig. 6.1; afin d'alléger la

notation, les produits $y_{3.1} Y_{3.1}(\underline{y})$ ne sont représentés que par leurs indices : 3.1).

$$\begin{aligned}
 R^{Y^{\pi}}(\underline{y}) = & (3.1 + 14.1 + 16.1) \cdot \\
 & \cdot (3.2 + 6.2 + 8.2 + 16.2) \cdot \\
 & \cdot (3.3 + 4.3 + 13.3 + 14.3) \cdot \\
 & \cdot (3.4 + 4.4 + 6.4 + 7.4 + 12.4) \cdot \\
 & \cdot (3.5 + 4.5 + 6.5 + 7.5 + 12.5) \cdot \\
 & \cdot (2.6 + 6.6 + 7.6 + 10.6) \cdot \\
 & \cdot (2.7 + 6.7 + 7.7 + 10.7) \cdot \\
 & \cdot (1.8 + 15.8 + 17.8) \cdot \\
 & \cdot (1.9 + 5.9 + 8.9 + 17.9) \cdot \\
 & \cdot (4.10 + 5.10 + 7.10 + 11.10) \cdot \\
 & \cdot (4.11 + 5.11 + 7.11 + 11.11) \cdot \\
 & \cdot (1.12 + 2.12 + 13.12 + 15.12) \cdot \\
 & \cdot (1.13 + 2.13 + 5.13 + 7.13 + 9.13) \cdot \\
 & \cdot (1.14 + 2.14 + 5.14 + 7.14 + 9.14)
 \end{aligned}$$

La résolution de l'expression $R^{Y^{\pi}}(\underline{y})$ se fait au moyen d'une opération de concentration; cette opération consiste à remplacer le produit de deux termes par un produit simplifié :

$$y_{k_1 \ell_1} Y_{k_1 \ell_1}(\underline{y}) \cdot y_{k_2 \ell_2} Y_{k_2 \ell_2}(\underline{y}) \longrightarrow y_{k_3 \ell_3} Y_{\ell_3}(\underline{y})$$

L'opération de concentration n'est possible que dans le cas où les systèmes $U_{k_1 \ell_1}(\underline{a}, y_{k_1 \ell_1})$ et $U_{k_2 \ell_2}(\underline{a}, y_{k_2 \ell_2})$ sont compatibles (c'est-à-dire que les "1" de l'un n'ont pas d'interaction avec les "0" de l'autre, et réciproquement); une nouvelle fonction $Y_{\ell_3}(\underline{y})$ décrit la couverture du système formé par la réunion en un seul des deux systèmes concaténés.

La concentration permet de réaliser avec une fonction unique le masquage de plusieurs composants; dans le cas où $k_1 = k_2$, on obtient la réunion en une seule de deux versions différentes d'un même composant.

α_1 α_2 α_3 α_4	3.1 14.1 16.1	3.2 6.2 8.2 16.2	3.3 4.3 13.3 14.3	1.14 2.14 5.14 7.14 9.14
0 0 0 0	0 0 0			
1 0 0 0		0 0 0 0		
1 1 0 0		1 1		
0 0 1 0			0 0 0 0	
1 0 1 0	1	1 1	1 1	
0 1 1 0	1		1 1	
1 1 1 0				1 1
1 0 0 1		1 1		
0 1 0 1				
1 1 0 1				
0 0 1 1			1 1	
1 0 1 1				1 1
0 1 1 1				
1 1 1 1				0 0 0 0 0

Fig. 6.1 : systèmes associés aux composants mixtes.

Le terme obtenu par concentration ne remplace pas strictement les termes dont il est issu; il existe parmi eux de la même façon qu'un terme consensus existe parmi les impliquants qui l'engendrent.

Exemple : dans l'exemple cité, les systèmes $U_{3.1}$, $U_{3.2}$, $U_{3.3}$, $U_{7.6}$, $U_{7.7}$, $U_{1.8}$, $U_{1.9}$, $U_{7.10}$, $U_{7.11}$ et $U_{1.12}$ sont compatibles et peuvent être concentrés en un seul système U_{15} (dont la table de vérité est présentée sur la fig. 6.2) :

$$Y_{1.15} Y_{3.15} Y_{7.15} Y_{15}(y)$$

Le système U_{15} n'est compatible avec aucun des systèmes restants; il est alors utile d'expliciter sa fonction de couverture $Y_{15}(y)$:

α_1	α_2	α_3	α_4	U_{15}	$U_{9 \cdot 16}$	$U_{12 \cdot 16}$	U_{17}
0	0	0	0	0			
1	0	0	0	0			
1	1	0	0				
0	0	1	0	0			
1	0	1	0	1		0	0
0	1	1	0				
1	1	1	0	0			
1	0	0	1				
0	1	0	1	0			
1	1	0	1	0			
0	0	1	1				
1	0	1	1	0			
0	1	1	1	0			
1	1	1	1	1	0		0

Fig. 6.2 : table de vérité des systèmes U_{15} ,
 $U_{9 \cdot 16}$, $U_{12 \cdot 16}$ et U_{17} .

$$Y_{15}(\underline{y}) = y_{9 \cdot 16} Y_{9 \cdot 16}(\underline{y}) \cdot y_{12 \cdot 16} Y_{12 \cdot 16}(\underline{y}) + \dots$$

La fonction de couverture $Y_{15}(\underline{y})$, une fois explicitée, permet de construire la fonction de masque $M_{15}(\underline{a})$ qui sera appliquée à la version $\ell = 15$ des trois composants $P_1(\underline{a})$, $P_3(\underline{a})$ et $P_7(\underline{a})$ (resp. $a_2 a_4$, $\bar{a}_2 \bar{a}_4$ et $a_1 a_3$):

D'autre part, les systèmes restants $U_{12 \cdot 4}$, $U_{12 \cdot 5}$, $U_{9 \cdot 13}$ et $U_{9 \cdot 14}$ sont également compatibles et peuvent être réunis en un seul système U_{17} (dont la table de vérité est donnée ci-dessus) :

$$y_{9 \cdot 17} y_{12 \cdot 17} Y_{17}(\underline{y})$$

Finalement, les deux systèmes obtenus sont compatibles et peuvent être concentrés :

$$Y_{1 \cdot 15} Y_{3 \cdot 15} Y_{7 \cdot 15} Y_{9 \cdot 16} Y_{12 \cdot 16} Y_{9 \cdot 16}(\underline{y}) Y_{12 \cdot 16}(\underline{y}) \quad (1)$$

$$Y_{9 \cdot 17} Y_{12 \cdot 17} Y_{17}(\underline{y}) \quad (2)$$

$$Y_{1 \cdot 15} Y_{3 \cdot 15} Y_{7 \cdot 15} Y_{9 \cdot 18} Y_{12 \cdot 18} Y_{18}(\underline{y}) \quad (\text{terme concentré issu de 1 et 2})$$

avec $Y_{18}(\underline{y}) = 1$, car le système U_{18} qui lui est associé n'est composé que de "0" et de "X" (donc aucun composant n'est nécessaire pour le construire).

Ainsi, cinq composants mixtes sont nécessaires pour réaliser le système (cf. fig. 6.3, 6.4 et 6.5) :

composant	version	fonction
$P_1(\underline{a}) = a_2 a_4$	15	$F_1(\underline{a})$
$P_3(\underline{a}) = \bar{a}_2 \bar{a}_4$	15	$F_2(\underline{a})$
$P_7(\underline{a}) = a_1 a_3$	15	$F_1(\underline{a}) \quad F_2(\underline{a})$
$P_9(\underline{a}) = a_1 \bar{a}_2 a_3 \bar{a}_4$	18	$F_1(\underline{a}) \quad F_3(\underline{a}) \quad M_{15}(\underline{a})$
$P_{12}(\underline{a}) = a_1 a_2 a_3 a_4$	18	$F_2(\underline{a}) \quad F_3(\underline{a}) \quad M_{15}(\underline{a})$

Fig. 6.3 : attribution des composants dans le système synthétisé.

$$F_1(\underline{a}) = a_2 a_4 \overline{M_{15}(\underline{a})} + a_1 a_3 \overline{M_{15}(\underline{a})} + a_1 \bar{a}_2 a_3 \bar{a}_4$$

$$F_2(\underline{a}) = \bar{a}_2 \bar{a}_4 \overline{M_{15}(\underline{a})} + a_1 a_3 \overline{M_{15}(\underline{a})} + a_1 a_2 a_3 a_4$$

$$F_3(\underline{a}) = M_{15}(\underline{a}) = a_1 \bar{a}_2 a_3 \bar{a}_4 + a_1 a_2 a_3 a_4$$

Remarques

La méthode de synthèse multi-niveaux dont nous avons donné une description informelle ci-dessus, comme d'ailleurs la synthèse approchée à laquelle elle s'apparente, fournit un nombre énorme de solutions, dont seule une très faible proportion est réellement intéressante.

L'algorithme qui recherche ces solutions en tentant de résoudre la fonction $R^{Y_i \pi}(\underline{y})$ doit alors sélectionner, parmi le nombre considérable de combinaisons possibles, celles qui sont à même de réaliser le système plus économiquement que sur deux niveaux.

La résolution exacte de cette fonction doit être remplacée par la recherche progressive d'une solution, qui se construit en traversant successivement les fonctions partielles $R^{Y_i}(\underline{y})$; cette progression ressemble à une devinette : quels sont les termes qui peuvent être concentrés avec profit, à quel stade de cette progression l'explicitation d'une fonction $Y_{\underline{a}}(\underline{y})$ fournit-elle le résultat le plus fructueux ?

Pour répondre valablement à ces deux questions, il conviendrait d'étudier les propriétés des opérations de concentration et d'explicitation.

Finalement, remarquons que l'utilisation du masquage direct ($M(\underline{\alpha}) = 1$: composant masqué, $M(\underline{\alpha}) = 0$: composant actif) permet de réaliser les fonctions de masque $M(\underline{a})$ avec les impliquants du système inverse; il est bien sûr possible d'utiliser conjointement (ou à la place du masquage direct) le masquage inverse ($M(\underline{\alpha}) = 0$: composant masqué); dans ce cas, on effectue algébriquement l'opération qui consiste à rechercher systématiquement les mises en facteur possibles des impliquants.

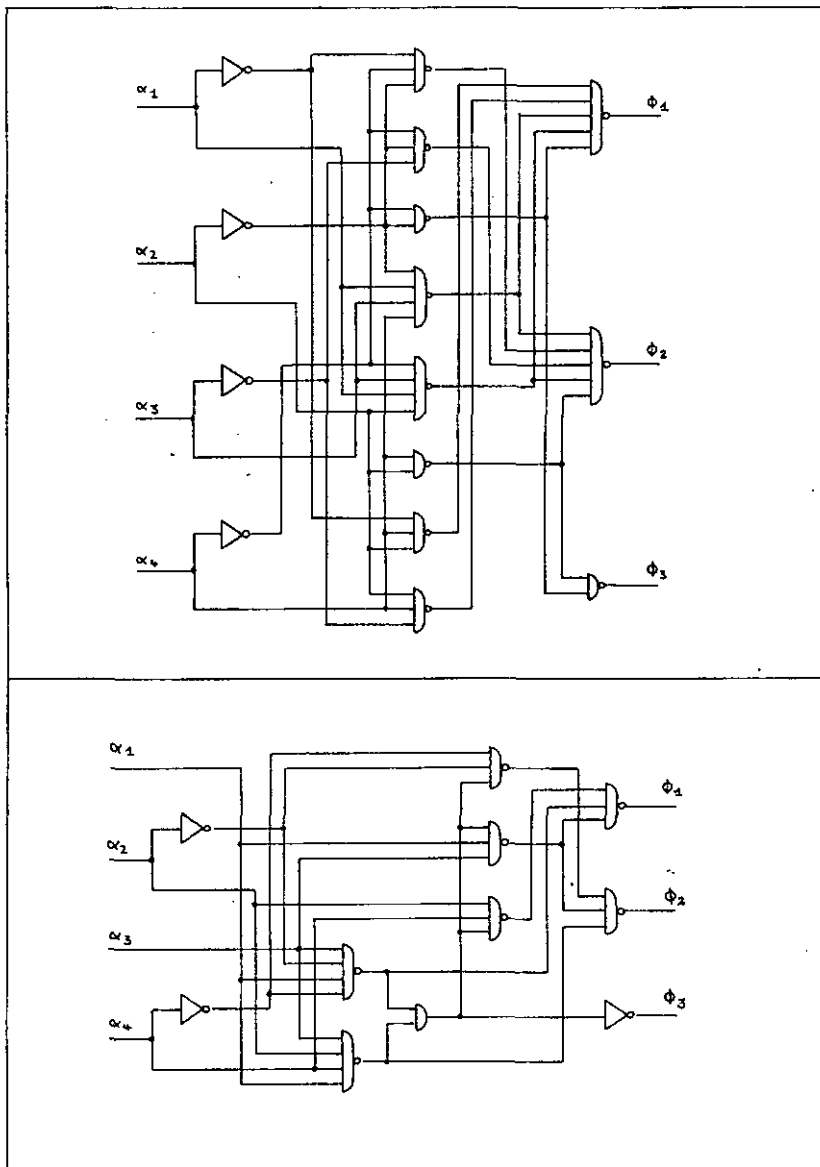


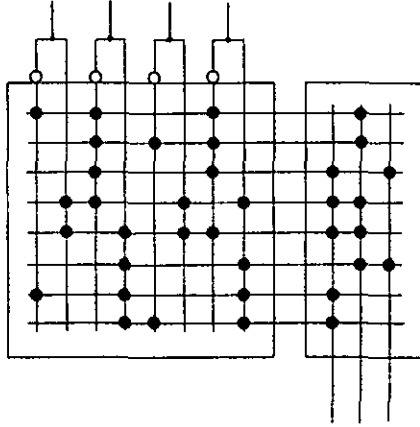
Fig. 6.4 : réalisations en portes sur deux et quatre niveaux.

PLA exacte

2 niveaux

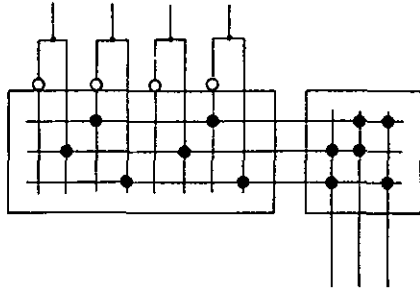
AND : 8 x 8

OR : 8 x 3



PLA approchée

2 niveaux

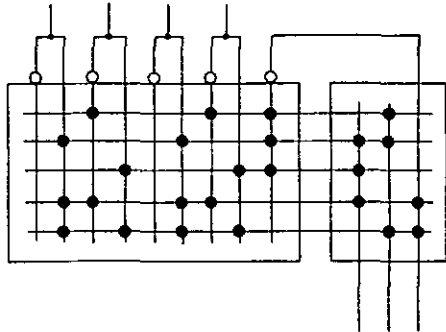


PLA exacte

4 niveaux

AND : 9 x 5

OR : 5 x 3



VII. CONCLUSION

Le formalisme algébrique développé dans le cadre de ce travail comprend la description d'une relation d'ordre au moyen d'une fonction booléenne caractéristique d'une part, et de l'autre le traitement simultané d'un ensemble de fonctions booléennes au moyen d'une fonction composée unique.

La synthèse classique d'un système logique combinatoire à sortie unique est effectuée par la formulation successive de deux modèles paramétrés dont on calcule l'adéquation aux données du problème au moyen des fonctions caractéristiques correspondantes. La synthèse d'un système à sortie multiple apparaît alors comme le calcul d'un ensemble de relations qui décrivent l'adéquation d'un ensemble de modèles paramétrés aux données spécifiques de chaque sortie; ce calcul est effectué de façon simultanée avec les moyens développés à cet effet.

Les procédés de synthèse classique, présentés ordinairement sous leur forme algorithmique, apparaissent alors comme des cas particuliers de la méthode ci-dessus, ce qui entraîne leur justification théorique complète dans le cadre de ce formalisme algébrique.

Finalement, les méthodes algébriques développées ici permettent l'ouverture vers le développement théorique de procédés de synthèse nouveaux; dans ce cas cependant, le problème majeur consiste à contourner une caractéristique inhérente à toute méthode algébrique : celle de fournir toutes les solutions d'un problème donné. La mise en œuvre de ces nouvelles méthodes ne se conçoit donc qu'avec l'utilisation d'algorithmes heuristiques.

◀ Fig. 6.5 : réalisation en PLA des exemples cités.

REMERCIEMENTS

Je tiens à remercier M. J.-J. Monbaron pour le soin qu'il a pris pour m'initier au domaine faisant l'objet de ce travail, ainsi que pour le soutien qu'il m'a apporté tout au long de son exécution.

Ma reconnaissance va également à M. F. Pellandini pour la diligence dont il a fait preuve lors de la direction de ce travail.

Je remercie aussi MM. W. Guggenbühl, A. Robert, A. Shah et Th. Stutz pour l'intérêt qu'ils ont porté à ce travail et les suggestions intéressantes qu'ils m'ont faites à ce sujet ainsi que, dans le cas de M. Stutz, pour les compléments bibliographiques qu'il m'a fournis.

Le travail faisant l'objet de la présente thèse a été accompli à l'Institut de Microtechnique de l'Université de Neuchâtel, et a été financé par le Fonds National Suisse pour la Recherche Scientifique, dans le cadre du projet n° 2'888-0.80.

APPENDICE A : NOTATIONS

1. Ensembles d'indices

I, J, \dots ensembles finis d'indices distincts (exemple :
nombres naturels $I = \{1, 2, 3, 4, 5\}$);

I', I_k, \dots sous-ensembles de I ;

\emptyset ensemble vide;

$i \in I$ l'indice i est un élément de l'ensemble I ;

\cap, \cup intersection, resp. réunion d'ensembles;

\subset, \subseteq inclusion stricte, resp. large d'ensembles;

\bar{I}_k complément du sous-ensemble I_k par rapport à
l'ensemble I ($I_k \cap \bar{I}_k = \emptyset$, $I_k \cup \bar{I}_k = I$);

$|I|$ cardinal de l'ensemble I (nombre d'éléments
dont il est composé).

2. Eléments de l'algèbre de Boole

$0, 1$ éléments constants de l'algèbre de Boole (éléments
neutres de la disjonction, resp. de la conjonction);

a_n, x_k, \dots variables booléennes indépendantes, identifiées
par un indice;

- $\underline{a}, \underline{x}, \dots$ jeux de variables booléennes indépendantes, en relation implicite avec un ensemble d'indices ($\underline{a} = \{a_n | n \in \mathbb{N}\}$, $\underline{x} = \{x_k | k \in K\}$, etc.);
- α_n, ξ_k, \dots valeurs particulières que l'on peut attribuer aux variables booléennes indépendantes a_n , resp. x_k (tout élément de l'algèbre de Boole peut être une valeur particulière : 0, 1, variable ou expression booléennes);
- $\underline{\alpha}, \underline{\xi}, \dots$ ensemble de valeurs particulières que l'on attribue aux variables \underline{a} , resp. \underline{x} ;
- $B(\underline{a}), V(\underline{a}, f), \dots$ fonctions booléennes des variables \underline{a} , resp. \underline{a} et f ;
- $B(\underline{\alpha}), V(\underline{\alpha}, \phi), \dots$ valeurs que prennent les fonctions $B(\underline{a})$ ou $V(\underline{a}, f)$ lorsque l'on donne des valeurs particulières à leurs arguments.

3. Relations et opérations booléennes

- $A(\underline{x}) = B(\underline{x})$ égalité des deux expressions $A(\underline{x})$ et $B(\underline{x})$;
- $A(\underline{x}) \leq B(\underline{x})$ implication de l'expression $B(\underline{x})$ par l'expression $A(\underline{x})$;
- $\overline{A(\underline{x})}$ complément (ou inverse) de l'expression $A(\underline{x})$;
- $A(\underline{x}) + B(\underline{x})$ disjonction des expressions $A(\underline{x})$ et $B(\underline{x})$ (appelée également somme booléenne ou, lorsque le risque de confusion est exclu, somme);

$$A(\underline{x}) \cdot B(\underline{x})$$

conjonction des expressions $A(\underline{x})$ et $B(\underline{x})$
 (appelée également produit booléen, ou simplement produit, lorsque le risque de confusion est exclu; la conjonction est aussi notée par la juxtaposition de ses arguments, surtout lorsque ceux-ci sont des variables : par exemple $a_1 a_2 a_3 a_4$);

$$\sum_{j \in J_k} f_j$$

disjonction de toutes les variables f_j dont l'indice appartient au sous-ensemble J_k (définie par récurrence à partir de $\sum f_j = 0$, $j \in \emptyset$);

$$\prod_{j \in J_k} f_j$$

conjonction de toutes les variables f_j dont l'indice appartient au sous-ensemble J_k (définie par récurrence à partir de $\prod f_j = 1$, $j \in \emptyset$).

APPENDICE B : DEFINITIONS DE BASE, AXIOMES ET PROPRIETES

DE L'ALGEBRE DE BOOLE

L'algèbre de Boole est un ensemble composé d'éléments en nombre illimité, entre lesquels une relation, l'égalité, et trois opérations de base, la disjonction, la conjonction et l'inversion, sont définies par dix axiomes.

1. Eléments de l'algèbre de Boole

Les éléments de l'algèbre de Boole se répartissent en trois catégories :

a) deux éléments constants distincts, notés 0 et 1, qui sont les éléments neutres respectivement de la disjonction et de la conjonction (axiomes n° 7 et 8);

b) les variables booléennes indépendantes, en nombre arbitraire, auxquelles on peut librement assigner des valeurs particulières (notamment les éléments 0 et 1, mais aussi tout autre élément de l'algèbre de Boole);

c) les fonctions booléennes, qui peuvent prendre une valeur déterminée (élément de l'algèbre de Boole) lorsque l'on assigne une valeur particulière à leur(s) argument(s); nous ne considérerons que les fonctions booléennes dont les arguments sont des variables booléennes indépendantes.

2. Définitions et axiomes

L'égalité entre deux éléments de l'algèbre de Boole est définie par récurrence :

a) $X = X$, un élément quelconque de l'algèbre de Boole est égal à lui-même;

b) $0 \neq 1$, les deux éléments constants ne sont pas égaux;

c) $A(x) = B(x) \iff A(\xi) = B(\xi) \quad \forall \xi$
 deux expressions booléennes sont égales si les valeurs qu'elles prennent sont égales pour toute valeur que l'on peut assigner à leurs arguments.

Les opérations de disjonction, de conjonction et d'inversion sont définies par les dix axiomes suivants, que vérifie tout élément X, Y ou Z de l'algèbre de Boole :

$$1. X + Y = Y + X$$

$$2. XY = YX$$

$$3. X + (Y + Z) = (X + Y) + Z$$

$$4. X(YZ) = (XY)Z$$

$$5. X + (YZ) = (X + Y)(X + Z)$$

$$6. X(Y + Z) = (XY) + (XZ)$$

$$7. X + 0 = X$$

$$8. X \cdot 1 = X$$

$$9. X + \bar{X} = 1$$

$$10. X \cdot \bar{X} = 0$$

3. Propriétés de base

Tout élément X ou Y de l'algèbre de Boole possède les propriétés suivantes :

11. $X + X = X$

12. $X \cdot X = X$

13. $X + 1 = 1$

14. $X \cdot 0 = 0$

15. $X + (XY) = X$

16. $X(X + Y) = X$

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

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

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

20. $X = (X + Y)(X + \overline{Y})$

21. $X + Y = Y \iff XY = X$

22. $X + Y = 0 \iff (X = 0 \text{ et } Y = 0)$

23. $XY = 1 \iff (X = 1 \text{ et } Y = 1)$

24. $Y = \overline{X} \iff (X + Y = 1 \text{ et } XY = 0)$

25. $\overline{\overline{X}} = X$

26. $X = Y \iff \overline{X} = \overline{Y}$

27. $\overline{\overline{X + Y}} = \overline{X} \cdot \overline{Y}$

28. $\overline{\overline{XY}} = \overline{X} + \overline{Y}$

Ces propriétés sont utilisées couramment sans référence explicite lorsque l'on travaille avec des expressions booléennes; elles se démontrent simplement à partir des axiomes n° 1 à 10; par exemple 11. :

$$\begin{aligned}
 X + X &= (X + X) \cdot 1 && \text{(axiome n° 8)} \\
 &= (X + X) \cdot (X + \overline{X}) && \text{(axiome n° 9)} \\
 &= X + (X\overline{X}) && \text{(axiome n° 5)} \\
 &= X + 0 && \text{(axiome n° 10)} \\
 &= X && \text{(axiome n° 7)}
 \end{aligned}$$

4. Définition et propriétés de l'implication

L'implication d'un élément Y par un autre élément X est notée $X \leq Y$; elle est définie par :

$$29. X \leq Y \iff \bar{X} + Y = 1$$

La terminologie est justifiée par la propriété :

$$A(\underline{x}) \leq B(\underline{x}) \iff [A(\underline{\xi}) = 1 \implies B(\underline{\xi}) = 1] \forall \underline{\xi}$$

L'implication jouit des propriétés ci-dessous, démontrables à partir des axiomes n° 1 à 10, ainsi que des théorèmes n° 11 à 28 :

$$30. X \leq Y \iff 'X\bar{Y} = 0$$

$$31. X \leq Y \iff X + Y = Y$$

$$32. X \leq Y \iff XY = X$$

$$33. X \leq X$$

$$34. (X \leq Y \text{ et } Y \leq X) \iff X = Y$$

$$35. (X \leq Y \text{ et } Y \leq Z) \implies X \leq Z$$

$$36. 0 \leq X \leq 1$$

$$37. X \leq Y \iff \bar{Y} \leq \bar{X}$$

$$38. (X \leq Y \text{ et } V \leq W) \implies X + V \leq Y + W$$

$$39. (X \leq Y \text{ et } V \leq W) \implies XV \leq YW$$

$$40. (X \leq Z \text{ et } Y \leq Z) \iff X + Y \leq Z$$

$$41. (X \leq Y \text{ et } X \leq Z) \iff X \leq YZ$$

$$42. X \leq X + Y$$

$$43. XY \leq X$$

5. Propriétés des fonctions booléennes

Une fonction booléenne $F(\underline{x})$ est définie par les valeurs $F(\underline{\xi})$ qu'elle prend pour chaque ensemble de valeurs $\underline{\xi}$ que l'on assigne à ses arguments \underline{x} . Toute fonction booléenne possède une expression explicite ne faisant intervenir que ses arguments et les trois opérations de base de l'algèbre de Boole (admis sans démonstration).

On appelle impliquant d'une fonction booléenne $F(\underline{x})$ un produit $P_k(\underline{x})$ de variables \underline{x} niées ou non, vérifiant les trois conditions :

a) les variables qui composent le produit appartiennent à l'ensemble des arguments de $F(\underline{x})$;

b) chacune des variables qui composent le produit n'y apparaît qu'une seule fois ;

c) le produit implique la fonction : $P(\underline{x}) \leq F(\underline{x})$

On appelle forme disjonctive d'une fonction son expression sous forme d'une somme d'impliquants. Toute fonction booléenne possède au moins une forme disjonctive (on obtient une forme disjonctive en effectuant tous les produits que contient une expression, jusqu'à ce qu'il ne reste plus que des produits de variables).

Exemple : $B(\underline{x}) = \bar{x}_1 \cdot (\bar{x}_2 x_4 + x_2 \bar{x}_3) + x_1 \cdot (\bar{x}_2 x_3 + x_2 \bar{x}_4 + \bar{x}_3 x_4)$

on obtient une forme disjonctive en effectuant les deux parenthèses :

$$B(\underline{x}) = \bar{x}_1 \bar{x}_2 x_4 + \bar{x}_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_4 + x_1 \bar{x}_3 x_4$$

On appelle impliquant premier d'une fonction un impliquant $P(\underline{x})$ qui n'est impliquant d'aucun autre impliquant $P'(\underline{x})$ de cette fonction :

$$\left. \begin{array}{l} P(\underline{x}) \leq F(\underline{x}) \\ \text{premier} \end{array} \right\} \iff \left\{ \begin{array}{l} \nexists P'(\underline{x}) \neq P(\underline{x}) \text{ tq.} \\ P(\underline{x}) \leq P'(\underline{x}) \leq F(\underline{x}) \end{array} \right.$$

Exemple : dans l'expression $B(\underline{x})$ ci-dessus,

$$x_1 x_2 \bar{x}_4 \leq B(\underline{x}) \quad \text{est premier; par contre,}$$

$$\bar{x}_1 \bar{x}_2 x_4 \leq B(\underline{x}) \quad \text{n'est pas premier, car :}$$

$$\bar{x}_1 \bar{x}_2 x_4 \leq \bar{x}_2 x_4 \leq B(\underline{x})$$

Propriétés : un impliquant qui en implique un autre possède plus de variables que lui; un impliquant premier est par conséquent composé d'un nombre minimum de variables.

Nous appellerons forme complète la forme disjonctive d'une fonction $F(\underline{x})$ qui fait apparaître tous ses impliquants premiers, et seulement ceux-ci; toute fonction booléenne possède une et une seule forme complète.

Exemple :
$$B(\underline{x}) = \bar{x}_2 x_4 + x_2 \bar{x}_3 + \bar{x}_3 x_4 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_4 + x_1 \bar{x}_3 x_4$$

Une forme disjonctive irrédondante est une somme d'impliquants premiers dont aucun n'est impliquant de la somme des autres.

Exemple : la fonction $B(x)$ possède deux formes irrédondantes :

$$B(x) = \bar{x}_2 x_4 + x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_4$$

$$B(x) = \bar{x}_2 x_4 + x_2 \bar{x}_3 + x_1 x_3 \bar{x}_4$$

6. Moyens de calcul

Il est souvent utile de rechercher la forme complète d'une fonction booléenne; un des moyens qui permet d'y parvenir part de son expression sous forme conjonctive. A l'opposé de la forme disjonctive, la forme conjonctive est un produit de sommes de variables, niées ou non; on l'obtient en effectuant toutes les sommes, jusqu'à ce qu'il ne reste plus que des sommes de variables.

Exemple : $B(x) = (x_2 + x_3 + x_4) \cdot (x_1 + \bar{x}_3 + x_4) \cdot (\bar{x}_2 + \bar{x}_3 + \bar{x}_4)$

Théorème B.1

Une forme conjonctive est un produit de formes complètes.

Théorème B.2

En effectuant le produit de deux formes complètes, on obtient aussi une forme complète, après élimination des impliquants non premiers.

Ces deux théorèmes sont simples à démontrer; ils fournissent le moyen de calculer sous forme complète n'importe quelle fonction booléenne :

Corollaire B.3

Le calcul sous forme disjonctive d'une fonction exprimée sous forme conjonctive fournit la forme complète de cette fonction, après élimination des impliquants non premiers.

Cette méthode de calcul d'une forme complète est appelée algorithme d'inversion, car on obtient sans calcul une expression conjonctive à partir d'une forme disjonctive de l'inverse d'une fonction (règle de Morgan).

Exemple : $\overline{B(x)} = \overline{x_2} \overline{x_3} \overline{x_4} + \overline{x_1} x_3 \overline{x_4} + x_2 x_3 x_4$

$$B(x) = (x_2 + x_3 + x_4) \cdot (x_1 + \overline{x_3} + x_4) \cdot (\overline{x_2} + \overline{x_3} + \overline{x_4})$$

$$\begin{aligned} B(x) = & x_1 \overline{x_2} x_3 + \overline{x_2} \overline{x_3} \overline{x_4} + \overline{x_1} \overline{x_2} x_4 + \overline{x_2} \overline{x_3} \overline{x_4} + \overline{x_2} x_4 + \\ & + \overline{x_1} \overline{x_2} \overline{x_3} + x_2 \overline{x_3} + \overline{x_1} \overline{x_3} \overline{x_4} + \overline{x_1} \overline{x_3} \overline{x_4} + \overline{x_3} x_4 + \\ & + x_1 x_2 \overline{x_4} + \overline{x_2} \overline{x_3} \overline{x_4} + x_1 x_3 \overline{x_4} \end{aligned}$$

$$B(x) = x_1 \overline{x_2} x_3 + \overline{x_2} x_4 + x_2 \overline{x_3} + \overline{x_3} x_4 + x_1 x_2 \overline{x_4} + x_1 x_3 \overline{x_4}$$

La méthode la plus simple pour effectuer les produits d'une expression conjonctive consiste à procéder itérativement [55]; cependant, il existe une méthode beaucoup plus rapide qui procède récursivement, en fractionnant l'expression à calculer par factorisation successive de toutes les variables qui la composent; ce processus arborescent s'applique également à la méthode des consensus [33].

APPENDICE C : CODAGE NUMERIQUE D'UNE TABLE DE VERITE

La synthèse automatisée par ordinateur de systèmes logiques combinatoires complexes fait intervenir le codage numérique des tables de vérité; mentionnons deux méthodes pour effectuer cette représentation.

1. Codage numérique à caractère binaire

La méthode généralement utilisée pour réaliser le codage d'une table de vérité se base sur la parenté qui existe entre l'objet à représenter et le support du codage, tous deux de nature binaire.

Cette méthode de codage associe deux bits du support (les bits du support sont groupés en mots-machine, octets, ou autres) à chaque bit de la table de vérité; chaque ligne de la table est ainsi représentée par deux groupes de mots-machine, l'un pour coder la polarité des bits définis de la table, l'autre pour coder la présence de ces bits (soit leur définition ou leur indéfinition). Plusieurs mots-machine sont nécessaires pour coder une ligne de la table, lorsque le nombre de bits que contient un mot-machine est inférieur à celui que contient une ligne de celle-ci (cf. fig. C.1).

Ce format de représentation est fixe : le nombre de mots-machine utilisés par ligne de la table est constant pour un problème donné; les bits de la table sont identifiables par la position qu'ils occupent dans le mot-machine ou dans le groupe de mots-machine.

Cette méthode de codage possède l'avantage de la compacité : il est en effet difficile d'imaginer une représentation plus dense d'une table

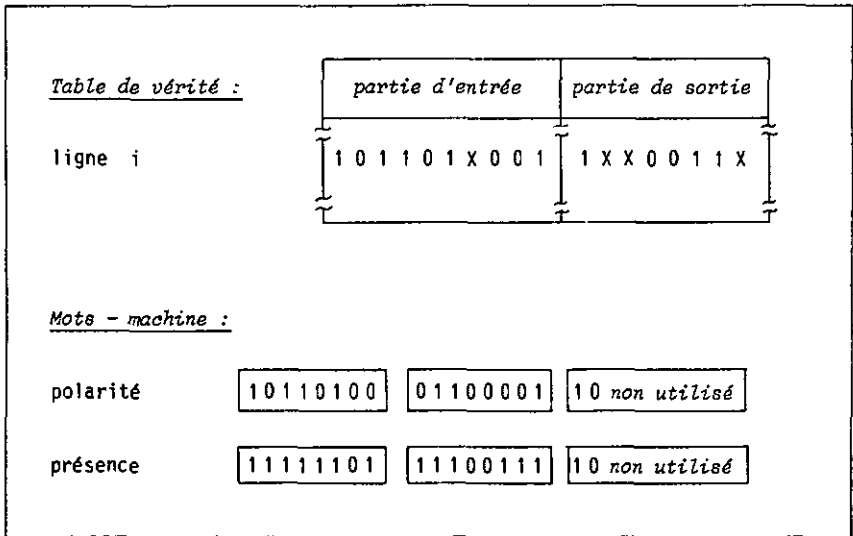


Fig. C.1 : codage numérique à caractère binaire d'une table de vérité.

de vérité, surtout si celle-ci est fortement définie; un autre avantage de cette méthode est la rapidité de traitement, car la plupart des ordinateurs disposent d'opérations logiques parallèles entre les mots-machine.

Ces avantages destinent cette représentation à caractère binaire aux algorithmes performants de synthèse classique.

2. Codage numérique à caractère algébrique

La deuxième méthode de codage numérique est une mise en pratique de la description algébrique d'une table de vérité (cf. chap. III, § 1).

Cette méthode utilise un mot-machine pour coder chaque bit défini de la table; l'un des bits de chaque mot (par exemple le signe) sert à spécifier la polarité de la variable codée; le reste du mot est

utilisé pour identifier cette variable. Un séparateur marque la fin de chaque ligne de la table de vérité (cf. fig. C.2).

L'identification des variables, avec cette méthode, étant explicite, on peut se passer de leur identification par la position et, par conséquent, d'un format fixe pour chaque ligne; ainsi, les bits indéfinis de la table ("X", ou "don't care") sont simplement omis par le procédé de transcription.

L'expression numérique formée par la suite des mots-machine obtenus réalise une représentation algébrique de l'expression logique qui était à coder; cette représentation possède l'avantage de ne jamais mentionner explicitement les états indéfinis. Ceci a pour conséquence que la place

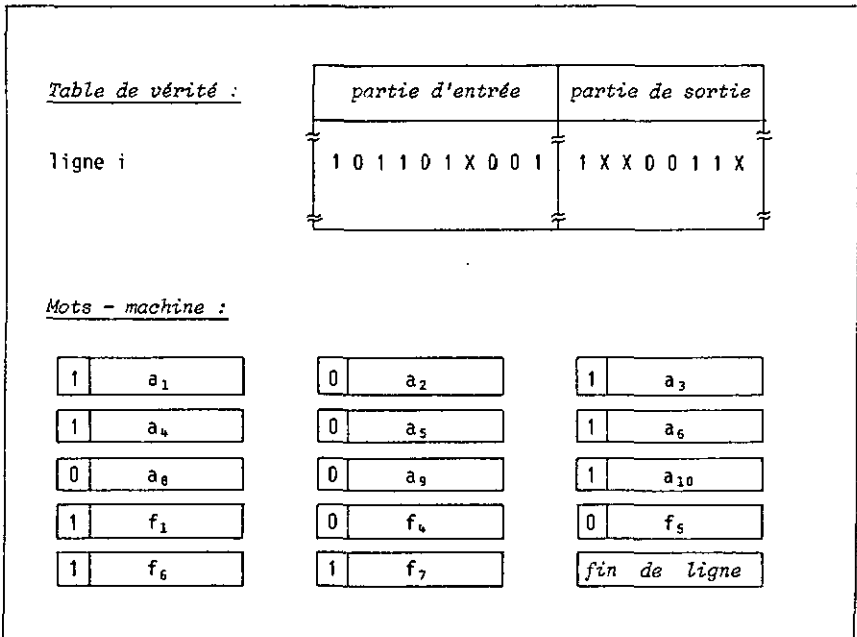


Fig. C.2 : codage numérique à caractère algébrique d'une table de vérité.

requis pour le stockage de l'information, ainsi que la durée d'exécution des algorithmes qui utilisent cette méthode ne dépendent que du contenu en information du problème à traiter, quel que soit le nombre des variables qui le composent.

Cependant, sa faible compacité, la relative lenteur des procédés qui l'utilisent due à l'absence d'opérations parallèles, ainsi que la complexité accrue de manipulation due à son format variable, rendent cette méthode de représentation à caractère algébrique peu intéressante pour les algorithmes de synthèse classique.

Par contre, sa nature algébrique la destine à l'implantation des algorithmes de synthèse qui font intervenir un grand nombre de variables booléennes auxiliaires, comme c'est le cas des méthodes décrites au chapitre VI.

Finalement, la méthode de représentation algébrique offre un exemple intéressant d'application pratique du formalisme algébrique.

APPENDICE D : PERMUTATION Σ - Π GRACE A LA CONDITION DE COHERENCE

Soit à démontrer la permutation des opérateurs somme et produit dans la dérivation de la fonction $R^{\underline{A}}(\underline{a}, \underline{y}, \underline{f})$ (chap. V, § 2, p. 59), en utilisant la condition de cohérence d'écriture de la table de vérité (chap. III, § 2) :

$$\prod_{j \in J} \sum_{i \in I} E_i(\underline{a}) \cdot (S_{ij}^A(f_j) + \bar{f}_j) = \sum_{i \in I} \prod_{j \in J} E_i(\underline{a}) \cdot (S_{ij}^A(f_j) + \bar{f}_j)$$

simplifions la notation :

$$E_i(\underline{a}) + E_i \quad \text{et} \quad (S_{ij}^A(f_j) + \bar{f}_j) + S_{ij}$$

écrivons alors la première expression sous forme développée :

$$\begin{aligned} & \prod_{j \in J} \sum_{i \in I} E_i \cdot S_{ij} = \\ & = (E_1 S_{11} + E_2 S_{21} + E_3 S_{31} + \dots + E_i S_{i1}) \\ & \quad (E_1 S_{12} + E_2 S_{22} + E_3 S_{32} + \dots + E_i S_{i2}) \\ & \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\ & \quad (E_1 S_{1j} + E_2 S_{2j} + E_3 S_{3j} + \dots + E_i S_{ij}) \\ & = \prod_j E_1 S_{1j} + \prod_j E_2 S_{2j} + \prod_j E_3 S_{3j} + \dots + \prod_j E_i S_{ij} \end{aligned}$$

seuls restent les produits verticaux; en effet, par la condition de cohérence, tout produit de termes diagonaux est soit nul, soit inclus dans l'un des produits verticaux.

La condition de cohérence nous fournit la propriété suivante :

$$E_1 \cdot E_2 \neq 0 \quad \Rightarrow \quad S_{1j} = S_{2j} \quad \forall j$$

Considérons alors un produit diagonal de l'expression développée ci-dessus :

$$\begin{aligned} & E_2 \cdot S_{21} \cdot E_1 \cdot S_{12} \cdot E_1 \cdot S_{13} \cdot E_1 \cdot S_{14} \cdot \dots \cdot E_1 \cdot S_{1j} = \\ & = E_1 \cdot E_2 \cdot S_{21} \cdot S_{12} \cdot S_{13} \cdot S_{14} \cdot \dots \cdot S_{1j} \end{aligned}$$

si $E_1 \cdot E_2 = 0$ ce terme est nul; par contre, si $E_1 \cdot E_2 \neq 0$, on a $S_{21} = S_{11}$, et ce terme croisé est égal à :

$$\begin{aligned} & = E_1 \cdot E_2 \cdot S_{11} \cdot S_{12} \cdot S_{13} \cdot S_{14} \cdot \dots \cdot S_{1j} \\ & \leq E_1 \cdot S_{11} \cdot S_{12} \cdot S_{13} \cdot S_{14} \cdot \dots \cdot S_{1j} = \prod_j E_1 \cdot S_{1j} \end{aligned}$$

Ce terme est donc inclus dans le premier produit vertical (il l'est d'ailleurs également dans le second), ce qui permet de l'omettre. Il est dès lors évident que tout terme qui ne résulte pas du produit de facteurs juxtaposés verticalement dans la formule développée ci-dessus subit le même sort, s'il n'est pas nul.

BIBLIOGRAPHIE

- [1] J. Kuntzmann : *"Algèbre de Boole"*
Dunod, Paris (1968).

- [2] D. L. Dietmeyer : *"Logic Design of Digital Systems"*
Allyn & Bacon, Boston (1971).

- [3] M. Davio, J.-P. Deschamps, A. Thayse :
"Discrete and switching functions"
McGraw Hill, New York (1978).

- [4] P. L. Hammer, S. Rudeanu :
"Boolean Methods in Operations Research"
Springer, Berlin (1968).

- [5] A. Blake : *"Canonical expressions in Boolean algebra"*
Ph. D. Thesis, Univ. of Chicago, Ill. (1938).

- [6] C. E. Shannon : *"A Symbolic Analysis of Relay and
Switching Circuits"*
Trans. AIEE 57 (1938) 713 - 723.

- [7] D. E. Muller : *"Application of Boolean Algebra to
Switching Circuit Design and Error Detection"*
IRE Trans. EC-3 (1954) 6 - 12.

- [8] R. S. Polansky : *"Simplification of Multiple-Output
Switching Functions"*
Rep. 7849-M-330, Electronic Systems Lab., MIT, Mass. (1959).

- [9] T. C. Bartee : *"Automatic design of logical networks"*
Proc. West. Joint Computer Conf. (Mars 1959) 103 - 107.
- [10] T. C. Bartee : *"Computer Design of Multiple-Output Logical Networks"*
IRE Trans. EC-10 (1961) 21 - 30
- [11] G. C. Vandling : *"The Simplification of Multiple-Output Switching Networks Composed of Unilateral Devices"*
IRE Trans. EC-9 (1960) 477 - 486.
- [12] M. J. Ghazala (Gazalé) : *"Irredundant Disjunctive and Conjunctive Forms of a Boolean Function"*
IBM Journal 1 (1957) 171 - 176.
- [13] T. Rado : *"Comments on the Presence Function of Gazalé"*
IBM Journal 6 (1962) 268 - 269.
- [14] J. P. Roth : *"Algebraic Topological Methods for the Synthesis of Switching Systems" I*
Trans. Am. Math. Soc. 88 (1958) 301 - 326.
- [15] J. P. Roth, E. G. Wagner : *"Algebraic Topological Methods for the Synthesis of Switching Systems"*
IBM Journal 4 (1959) 326 - 345.
- [16] J. P. Roth, E. G. Wagner : *"A Calculus and an Algorithm for a Logic Minimization Problem Together with an Algorithmic Notation"*
IBM Research RC 2280 (Nov. 1968), New York.
- [17] J. P. Roth : *"Programmed Logic Array Optimization"*
IEEE Trans. C-27 (1978) 174 - 176.

- [18] M. Karnaugh : *"The Map Method for Synthesis of Combinational Logic Circuits"*
Trans. AIEE Comm. Electron. 72 (1953) 593 - 598.
- [19] W. V. Quine : *"The problem of simplifying truth functions"*
Am. Math. Monthly 59 (1952) 521 - 531.
- [20] W. V. Quine : *"A way to simplify truth functions"*
Am. Math. Monthly 62 (1955) 627 - 631.
- [21] E. J. McCluskey Jr. : *"Minimization of Boolean Functions"*
Bell Syst. Techn. J. 35 (1956) 1417 - 1444.
- [22] T. H. Mott : *"Determination of the Irredundant Normal Forms of a Truth Function by Iterated Consensus of the Prime Implicants"*
IRE Trans. EC-9 (1960) 245 - 252.
- [23] S. R. Petrick : *"A Direct Determination of the Irredundant Forms of a Boolean Function from the Set of Prime Implicants"*
Techn. Rept. No. 56-110, Air Force Cambridge Research Center, Bedford, Mass. (1956).
- [24] P. Tison : *"Recherche des termes premiers d'une fonction booléenne - applications"*
Automatisme 9 (1964) 15 - 18.
- [25] P. Tison : *"Théorie des consensus"*
Thèse, Univ. de Grenoble (1965).
- [26] P. Tison : *"Generalization of Consensus Theory and Application to the Minimization of Boolean Functions"*
IEEE Trans. EC-16 (1967) 446 - 456.

- [27] J. F. Gimpel : *"A Reduction Technique for Prime Implicant Tables"*
IEEE Trans. EC-14 (1965) 535 - 541.
- [28] F. Luccio : *"A Method for the Selection of Prime Implicants"*
IEEE Trans. EC-15 (1966) 205 - 212.
- [29] R. M. Bowman, E. S. McVey : *"A Method for the Fast Approximate Solution of Large Prime Implicant Charts"*
IEEE Trans. C-19 (1970) 169 - 173.
- [30] J. R. Slagle, C.-L. Chang, R. C. T. Lee :
"A New Algorithm for Generating Prime Implicants"
IEEE Trans. C-19 (1970) 304 - 310.
- [31] T. L. Dollhoff, B. L. Weinberg : *"A Result on Set Extraction and Application to Covering-Closure Tables"*
IEEE Trans. C-21 (1972) 603 - 606.
- [32] T. L. Dollhoff : *"Program cuts logic design costs"*
Electronic Design 9 (1974) 186 - 189.
- [33] B. Reusch : *"Generation of Prime Implicants from Subfunctions and an Unifying Approach to the Covering Problem"*
IEEE Trans. C-24 (1975) 924 - 930.
- [34] A. Dunworth, A. van der Knaap : *"An efficient implementation of the sharp product operation for multiple-output switching cubes"*
Digital Processes 3 (1977) 161 - 176.
- [35] G. C. S. Bown : *"Use of n-cubes in the design of logic circuits"*
Electronic Letters 12 (1976) 240.

- [36] P. Weiner, T. F. Dwyer : *"Discussion of Some Flaws in the Classical Theory of Two-Level Minimization of Multiple-Output Switching Networks"*
IEEE Trans. C-17 (1968) 184 - 186.
- [37] V. D. Kazakov : *"The minimization of logical functions of a large number of variables"*
Automation and Remote Control (Mars 1967) 1161 - 1165,
traduit de : Avtomatika i Telemekhanika 23 (1962) 1237 - 1242
(en russe).
- [38] S. J. Hong, R. G. Cain, D. L. Ostapco :
"MINI : A Heuristic Approach for Logic Minimization"
IBM J. Res. Dev. 18 (1974) 443 - 458.
- [39] Z. Arevalo, J. G. Bredeson : *"A Method to Simplify a Boolean Function into a Near Minimal Sum-of-Products for Programmable Logic Arrays"*
IEEE Trans. C-27 (1978) 1028 - 1039.
- [40] A. Meo : *"On the Determination of the ps Maximal Implicants of a Switching Function"*
IEEE Trans. EC-14 (1965) 830 - 840.
- [41] J. F. Gimpe] : *"The Minimization of TANT Networks"*
IEEE Trans. EC-16 (1967) 18 - 38.
- [42] E. Cerny, M. A. Marin : *"A Computer Algorithm for the Synthesis of Memoryless Logic Circuits"*
IEEE Trans. C-23 (1974) 455 - 465.
- [43] E. Cerny, M. A. Marin : *"An Approach to Unified Methodology of Combinational Switching Circuits"*
IEEE Trans. C-26 (1977) 745 - 756.

- [44] J. Beister : *"Die Optimierung dreistufiger Schaltnetze - ein zweistufiges Bündelminimierungsproblem"*
Digital processes 5 (1979) 73 - 97.
- [45] D. Mange : *"Arbres de décision pour systèmes logiques câblés ou programmés"*
8u11. ASE/UCS 69 (1978) 1238 - 1243.
- [46] E. Cerny, D. Mange, E. Sanchez :
"Synthesis of Minimal Binary Decision Trees"
IEEE Trans. C-28 (1979) 472 - 482.
- [47] E. Sanchez, A. Thayse : *"Implementation and transformation of algorithms based on automata"*
Philips J. Res. 36 (1981) 159 - 172.
- [48] A. Thayse : *"Programmable and hard-wired synthesis of discrete functions" I*
Philips J. Res. 36 (1981) 40 - 73.
- [49] A. Thayse : *"P-Functions : A New Tool for the Analysis and Synthesis of Binary Programs"*
IEEE Trans. C-30 (1981) 126 - 134.
- [50] R. L. Ashenurst : *"The Decomposition of Switching Functions"*
Proc. Int. Symp. on Theory of Switching, Annals of Computation Lab., Harvard Univ. 29 (1959) 74 - 116.
- [51] E. L. Lawler : *"Minimal Boolean expressions with more than two levels of sums and products"*
Proc. AIEE 3rd Annual Symp. on Switching Theory and Logical Design, New York (1962).

- [52] E. L. Lawler : *"An Approach to Multilevel Boolean Minimization"*
J. Ass. Computing Machinery 11 (1964) 283 - 295.
- [53] B. Hazeltine : *"Construction of Irredundant Multilevel Switching Functions"*
IEEE Trans. EC-13 (1964) 468 - 470.
- [54] T. Ibaraki, S. Muroga : *"Synthesis of Networks with a Minimum Number of Negatives Gates"*
IEEE Trans. C-20 (1971) 49 - 58.
- [55] J.-J. Monbaron : *"Description des systèmes logiques combinatoires par des fonctions booléennes caractéristiques permettant le calcul des impliquants premiers et des couvertures irrédundantes au moyen d'un algorithme unique adapté aux petits ordinateurs"*
Thèse, Univ. de Neuchâtel (1980).
- [56] N. Peguiron : *"Ensemble de programmes pour la synthèse de systèmes logiques combinatoires - description et mode d'emploi"*
Rapport No. 90 EC 02-002, Institut de Microtechnique de l'Université de Neuchâtel (1983).

TABLE DES MATIERES

	<i>page</i>
<u>Résumé</u>	1
<u>I. Introduction</u>	3
1. Définitions	3
2. Notion de synthèse	4
3. Procédés de synthèse	6
4. But du présent travail	9
<u>II. Description et calcul d'une relation d'ordre</u>	11
1. Définitions	11
2. Exemples d'utilisation	14
<i>Recherche d'une forme irrédondante</i>	14
<i>Démonstration de théorèmes</i>	16
3. Application à la synthèse d'un système simple	18
<i>Recherche des impliquants premiers du système</i>	19
<i>Recherche d'une couverture minimale</i>	21
<u>III. Description algébrique d'une table de vérité</u>	25
1. Fonction de vérité	25
2. Condition de cohérence	29
3. Transformations de la table de vérité	30
<i>Bornes d'un sous-système individuel</i>	30

	<i>page</i>
<i>Borne inférieure d'un système</i>	31
<i>Borne supérieure d'un système</i>	32
<u>IV. Traitement simultané de fonctions booléennes</u>	35
1. Définitions	35
2. Obtention des solutions communes	37
<i>Impliquants de la fonction-produit</i>	37
<i>Impliquants de la fonction-somme</i>	39
3. Règles de transformation	43
<i>Diagramme de transformation</i>	45
4. Aspect algorithmique :	47
<u>V. Application à la synthèse classique sur deux niveaux d'un système logique combinatoire à sortie multiple</u>	49
1. Recherche des impliquants premiers du système	50
<i>Formation directe de la fonction composée</i>	51
<i>Interprétation du résultat</i>	53
2. Recherche des couvertures irrédondantes globales	56
<i>Formation directe de la fonction composée</i>	58
<i>Obtention du résultat</i>	61
<i>Interprétation du résultat</i>	62
3. Remarques	66
<u>VI. Autres applications du formalisme algébrique</u>	69
1. Composants premiers	69
<i>Calcul à partir de la table de vérité</i>	70

	<i>page</i>
2. Synthèse approchée de systèmes	72
<i>Fonction de couverture généralisée</i>	72
<i>Interprétation des impliquants obtenus</i>	74
<i>Réduction du nombre de solutions obtenues</i>	76
3. Synthèse multi-niveaux	77
<i>Recherche des composants mixtes</i>	77
<i>Synthèse à quatre niveaux avec les composants mixtes</i>	80
<i>Remarques</i>	86
<u>VII. Conclusion</u>	89
<u>Remerciements</u>	91
<u>Appendice A : notations</u>	93
1. Ensembles d'indices	93
2. Eléments de l'algèbre de Boole	93
3. Relations et opérations booléennes	94
<u>Appendice B : définitions de base, axiomes et propriétés</u> <u>de l'algèbre de Boole</u>	97
1. Eléments de l'algèbre de Boole	97
2. Définitions et axiomes	98
3. Propriétés de base	98
4. Définition et propriétés de l'implication	100
5. Propriétés des fonctions booléennes	101
6. Moyens de calcul	103

	<i>page</i>
<u>Appendice C : codage numérique d'une table de vérité</u>	105
1. Codage numérique à caractère binaire	105
2. Codage numérique à caractère algébrique	106
<u>Appendice D : permutation $\Sigma - \Pi$ grâce à la condition <u>de cohérence</u></u>	 109
<u>Bibliographie</u>	111
<u>Table des matières</u>	119