

**Zeitschrift:** Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association suisse des électriciens, de l'Association des entreprises électriques suisses

**Herausgeber:** Schweizerischer Elektrotechnischer Verein ; Verband Schweizerischer Elektrizitätsunternehmen

**Band:** 75 (1984)

**Heft:** 1

**Artikel:** Méthodes de synthèse des systèmes combinatoires CMOS

**Autor:** Stauffer, A.

**DOI:** <https://doi.org/10.5169/seals-904329>

### **Nutzungsbedingungen**

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

### **Conditions d'utilisation**

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

### **Terms of use**

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

**Download PDF:** 17.03.2025

**ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>**

# Méthodes de synthèse des systèmes combinatoires CMOS

A. Stauffer

Tout système combinatoire, défini par sa table de vérité, admet des méthodes de synthèse directe de schémas à transistors CMOS. Ces méthodes évitent l'usage intermédiaire de schémas logiques et généralisent le concept de porte de transmission sous forme de branche de transmission. Elles permettent de concevoir des circuits à sortie 3-états et des circuits préchargés. Les schémas logiques obtenus ultérieurement ne servent qu'à faciliter la compréhension des fonctions réalisées par les circuits.

Für jedes kombinatorische System, das durch seine Wahrheitstabelle definiert ist, gibt es Methoden zum direkten Entwerfen der Schaltungen mit CMOS-Transistoren. Diese Syntheseverfahren gestatten es, das logische Schema zu umgehen. Sie sind eine Verallgemeinerung des Konzeptes der Transmission-Gates in Form von Transmission-Zweigen. Sie ermöglichen es, Schaltungen mit drei Ausgangszuständen und vorkonfigurierte Schaltkreise zu verwirklichen. Das nachträglich erhaltene logische Schema dient nur der besseren Verständlichkeit der in der Schaltung enthaltenen Funktionen.

Ce travail est partiellement financé par la Commission pour l'encouragement de la recherche scientifique, crédit 1085.1.

## Adresse de l'auteur

A. Stauffer, Laboratoire de systèmes logiques, EPFL, 16, chemin de Bellerive, 1007 Lausanne.

## 1. Circuit CMOS

### 1.1 Transistor nMOS

Le symbole électronique du transistor MOS à canal  $n$  (nMOS) de la figure 1 regroupe les trois électrodes de cet élément: le drain  $a$ , la grille  $b$  et la source  $c$ . Lorsqu'on applique un signal logique sur sa grille, le transistor nMOS se comporte fonctionnellement comme un interrupteur placé entre sa source et son drain. Pour  $b = 0$ , le transistor assure le blocage et correspond à un interrupteur ouvert. Pour  $b = 1$ , il réalise la conduction et s'apparente à un interrupteur fermé. Le transistor nMOS est spécialisé dans la transmission de la masse.

### 1.2 Transistor pMOS

La figure 2 donne le symbole électronique du transistor MOS à canal  $p$  (pMOS). Selon qu'il conduit pour  $b = 0$  ou qu'il bloque pour  $b = 1$ , ce transistor présente un fonctionnement équivalent à celui d'un interrupteur fermé ou ouvert. Le transistor pMOS est spécialisé dans la transmission de l'alimentation.

### 1.3 Circuit logique CMOS

Le schéma électronique d'un circuit logique combinatoire à transistors MOS complémentaires (CMOS) apparaît à la figure 3. Il se compose de deux réseaux de transistors, respectivement à canal  $n$  et à canal  $p$ , reliés le premier à la masse et le second à l'alimentation. Ces réseaux, formés chacun par la mise en parallèle de branches de transistors connectés en série, sont

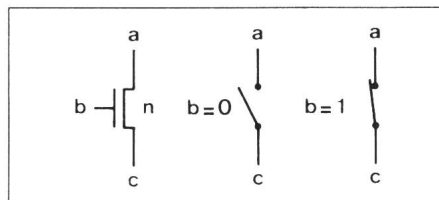


Fig. 1

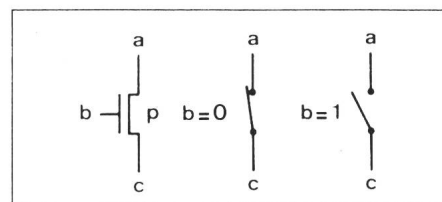


Fig. 2

commandés par les signaux d'entrée  $a$ ,  $b$  et  $c$  de telle sorte qu'ils ne conduisent jamais simultanément. La génération d'un signal de sortie  $z = 0$  résulte de la conduction du réseau  $n$  et du blocage du réseau  $p$ . Celle de  $z = 1$  découle du blocage du réseau  $n$  et de la conduction du réseau  $p$ .

D'une manière générale, la conduction d'une branche d'un des réseaux suffit à assurer la conduction du réseau entier. Le blocage de ce réseau nécessite par contre le blocage de chacune de ses branches.

Inversément, le blocage d'un des transistors d'une branche suffit à assurer le blocage de la branche entière. La conduction de cette branche nécessite par contre la conduction de l'ensemble de ses transistors.

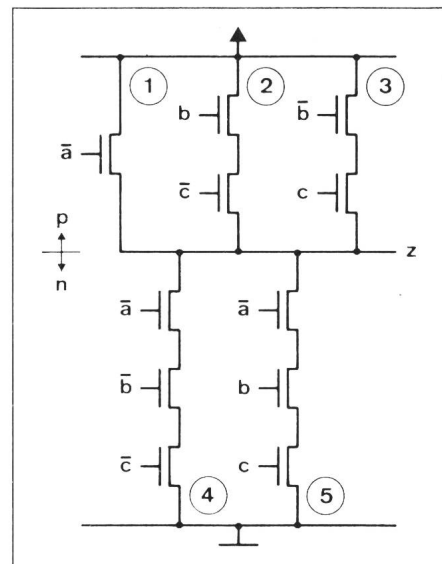


Fig. 3

### 1.4 Equations des réseaux $n$ et $p$

Les équations des réseaux

$$n(z) = \bar{a}\bar{b}\bar{c} + \bar{a}bc \quad (1)$$

$$p(z) = \bar{a} + b\bar{c} + \bar{b}c \quad (2)$$

décrivent la disposition topologique des transistors  $n$  et des transistors  $p$  de la fonction  $z$ . Dans ces équations, chacun des réseaux est défini par la somme logique des branches en parallèle, chaque branche correspond à un produit logique des transistors connectés en série et chaque transistor s'exprime par la variable logique vraie ou complémentée attachée à sa grille.

### 1.5 Equation de la fonction $z$

Le fonctionnement du circuit logique de la figure 3 peut être représenté dans une *table de Karnaugh* [1, pp. 50...73] qui exprime les valeurs de  $z$  en fonction de celles de  $a$ ,  $b$  et  $c$ .

Chacune des branches du réseau  $p$ , lorsqu'elle conduit, correspond à un bloc de 1 dans la table. Conformément au fonctionnement en interrupteur du transistor pMOS, les branches 1, 2, 3 conduisent respectivement pour  $\bar{a} = 0$ , pour  $b = 0$  et  $\bar{c} = 0$ , pour  $\bar{b} = 0$  et  $c = 0$ . Ces branches déterminent ainsi trois blocs de 1 associés à la variable  $a$ , au produit  $\bar{b}c$  et au produit  $b\bar{c}$  (fig. 4).

Chacune des branches du réseau  $n$ , lorsqu'elle conduit, correspond à un bloc de 0 dans la table. Compte tenu du fonctionnement en interrupteur du transistor nMOS, la conduction est assurée pour  $\bar{a} = 1$ ,  $\bar{b} = 1$  et  $\bar{c} = 1$  (branche 4) et pour  $\bar{a} = 1$ ,  $b = 1$  et  $c = 1$  (branche 5). Cette conduction définit ainsi deux blocs de 0 relatifs aux mintermes  $\bar{a}\bar{b}\bar{c}$  et  $\bar{a}bc$  (fig. 4).

L'équation de la fonction  $z$  réunit dans une seule expression les blocs de 0 et les blocs de 1 de la table de Karnaugh. Elle réalise la somme logique des deux mintermes multipliés par 0 avec la variable et les deux produits multipliés par 1:

$$z = [\bar{a}\bar{b}\bar{c} + \bar{a}bc]0 + [a + \bar{b}c + b\bar{c}]1 \quad (3)$$

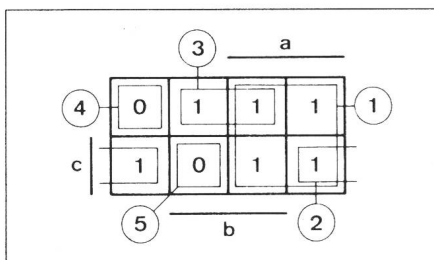


Fig. 4

Les équations (1) et (2) relatives aux réseaux  $n$  et  $p$  se retrouvent aisément à partir de l'équation de la fonction  $z$ . En effet,  $n(z)$  correspond directement à l'expression multipliée par 0 et  $p(z)$  découle de l'expression multipliée par 1 lorsqu'on complémente chacune de ses variables.

### 1.6 Méthode

L'analyse du circuit combinatoire précédent nous fournit, pour la technologie CMOS, la méthode de simplification des fonctions complètement définies:

1. introduire la fonction logique dans la table de Karnaugh;
2. rechercher visuellement et encadrer les blocs de 1 qui correspondent à des impliquants premiers de la fonction;
3. rechercher visuellement et encadrer les blocs de 0 qui correspondent à des impliquants premiers de la fonction inverse;
4. marquer d'un astérisque (\*) les impliquants premiers essentiels de la fonction et de la fonction inverse;
5. déterminer un polynôme final composé de l'ensemble des impliquants premiers essentiels et d'un ensemble minimal d'impliquants premiers non essentiels destinés à assurer la couverture des 1 et des 0 de la table qui ne sont pas inclus dans l'un ou l'autre des impliquants premiers essentiels; dans ce polynôme, les impliquants premiers de la fonction sont multipliés par 1 et ceux de la fonction inverse par 0.

La méthode de simplification proposée (comme toutes celles qui apparaîtront ultérieurement dans cet article) conduit à un polynôme comportant un nombre minimal de monômes. Elle vise donc à minimiser le nombre de branches du circuit.

### 1.7 Exemple

On cherche à effectuer la synthèse d'un circuit combinatoire défini par la forme canonique décimale de sa sortie:

$$z(a, b, c) = \Sigma 1, 2, 4, 5, 6 \quad (4)$$

Opérée dans la table de Karnaugh de la figure 5, la simplification de la fonction  $z$  fait apparaître quatre impli-

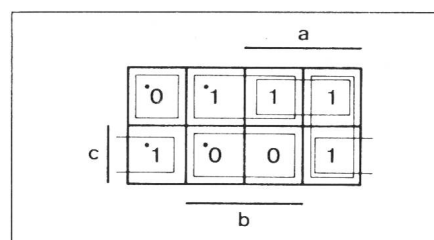


Fig. 5

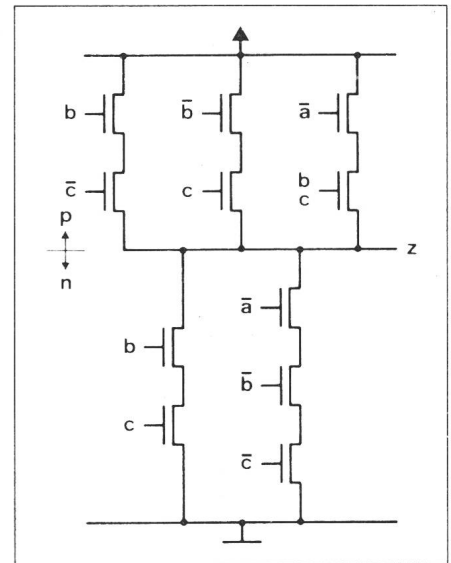


Fig. 6

quants premiers essentiels et deux impliquants premiers non essentiels. Les deux polynômes qui en résultent sont résumés dans l'équation finale de  $z$ :

$$z = [bc + \bar{a}\bar{b}\bar{c}]0 + [\bar{b}c + b\bar{c} + \left\{ \begin{matrix} a\bar{b} \\ a\bar{c} \end{matrix} \right\}]1 \quad (5)$$

Les équations des réseaux

$$n(z) = bc + \bar{a}\bar{b}\bar{c} \quad (6)$$

$$p(z) = \bar{b}c + b\bar{c} + \left\{ \begin{matrix} a\bar{b} \\ a\bar{c} \end{matrix} \right\} \quad (7)$$

découlent de la relation (5). Elles conduisent au schéma électronique de la figure 6.

### 1.8 Application: inverseur

Le symbole logique de l'inverseur apparaît à la figure 7. La table de Karnaugh, qui décrit le fonctionnement de cet opérateur, permet d'en déduire l'équation

$$z = [a]0 + [\bar{a}]1 \quad (8)$$

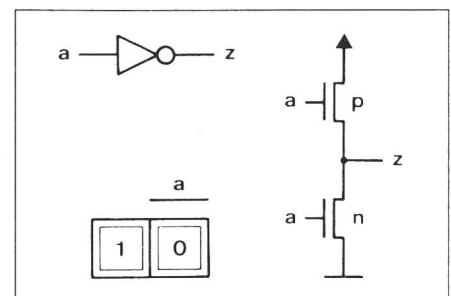


Fig. 7

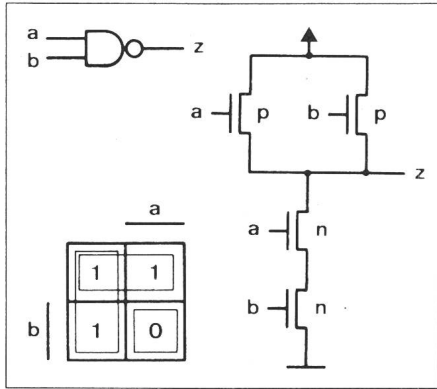


Fig. 8

Le schéma électronique de l'inverseur matérialise les équations des réseaux

$$n(z) = a; p(z) = a \quad (9)$$

### 1.9 Application: porte NAND

La figure 8 donne le symbole logique de la porte NAND. Opérée dans la table de Karnaugh, la simplification de la fonction réalisée par cet opérateur conduit à l'équation

$$z = [a b]0 + [\bar{a} + \bar{b}]1 \quad (10)$$

puis à celles des réseaux

$$n(z) = a b; p(z) = a + b \quad (11)$$

Le schéma électronique de la porte matérialise les relations (11).

### 1.10 Application: porte NOR

La porte NOR, définie par son symbole logique et sa table de Karnaugh (fig. 9), correspond à l'équation

$$z = [a + b]0 + [\bar{a} \bar{b}]1 \quad (12)$$

Le schéma électronique de cet opérateur matérialise les équations des réseaux

$$n(z) = a + b; p(z) = a b \quad (13)$$

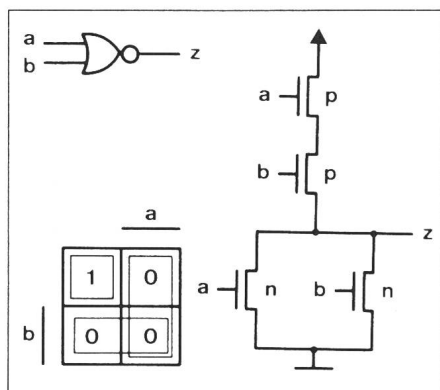


Fig. 9

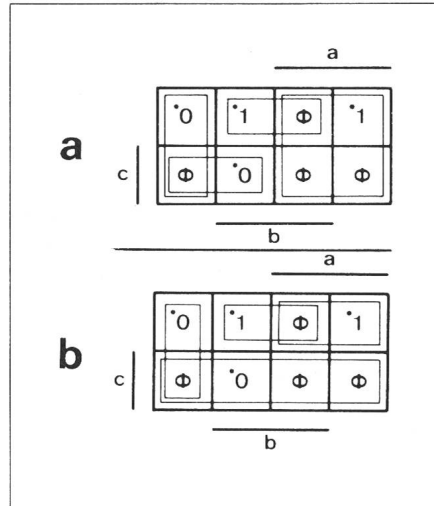


Fig. 10

### 1.11 Condition indifférente

Lorsqu'une fonction logique présente des conditions indifférentes  $\Phi$  (en anglais: don't care conditions) pour certains de ses états d'entrée, c'est-à-dire qu'elle peut prendre indifféremment la valeur 0 ou la valeur 1 pour les états en question, on parle d'une fonction incomplètement définie. Au niveau de la table de Karnaugh, une condition indifférente d'une fonction incomplètement définie correspond à une case remplie d'un symbole  $\Phi$ .

### 1.12 Méthode

Le concept de condition indifférente  $\Phi$ , lié aux fonctions incomplètement définies, modifie quelque peu la méthode de simplification des circuits logiques CMOS (§ 1.6):

1. introduire la fonction logique incomplètement définie dans une table de Karnaugh;
2. rechercher visuellement les impliquants premiers de la fonction et de la fonction inverse en commençant par encadrer les blocs de 1 et de 0 qui comportent le plus

grand nombre de cases (ces blocs n'englobent que les conditions indifférentes  $\Phi$  qui permettent d'accroître leurs dimensions);

3. marquer d'un astérisque (\*) les impliquants premiers essentiels;
4. déterminer un polynôme final composé de l'ensemble des impliquants premiers essentiels et d'un ensemble minimal d'impliquants premiers non essentiels destinés à assurer la couverture des 1 et des 0 de la table qui ne sont pas inclus dans l'un ou l'autre des impliquants premiers essentiels; dans ce polynôme, les impliquants premiers de la fonction sont multipliés par 1 et ceux de la fonction inverse par 0.

### 1.13 Exemple

On cherche à simplifier la fonction incomplètement définie

$$z(a, b, c) = \Sigma 2, 4 + \Phi 1, 5, 6, 7 \quad (14)$$

En commençant par le plus grand bloc de 1, la simplification effectuée dans la table de Karnaugh de la figure 10a nous conduit à la relation

$$z = [\bar{a} \bar{b} + \bar{a} c]0 + [a + b \bar{c}]1 \quad (15)$$

En partant du plus grand bloc de 0, on passe par la table de la figure 10b pour aboutir à l'expression

$$z = [c + \bar{a} \bar{b}]0 + [a \bar{c} + b \bar{c}]1 \quad (16)$$

La matérialisation des deux équations de la fonction  $z$  engendrent des schémas électroniques de complexité comparable (fig. 11a et 11b).

## 2. Circuit à branches de transmission

### 2.1 Branches de transmission

Le schéma électronique du circuit combinatoire de la figure 12 comporte

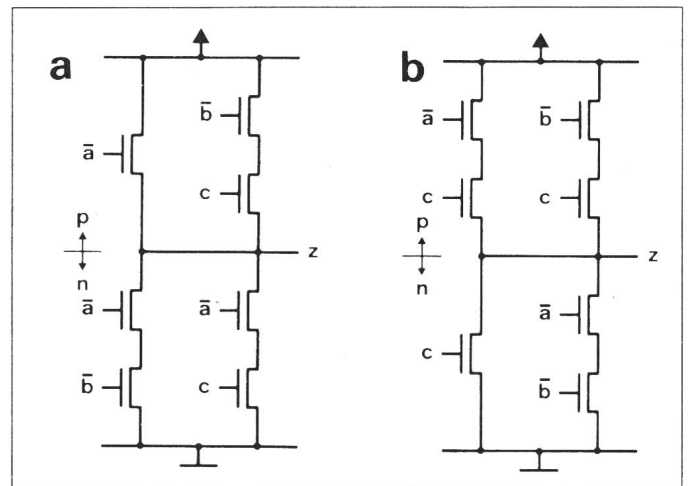


Fig. 11

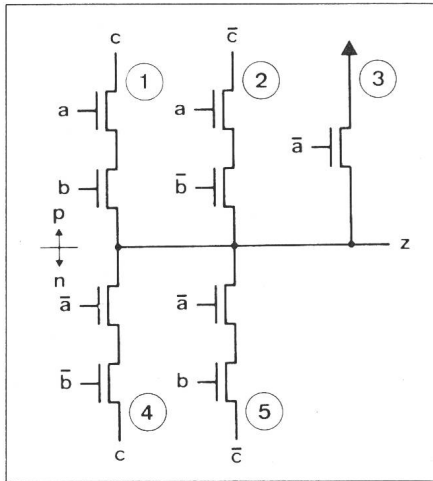


Fig. 12

des branches de transmission, c'est-à-dire des branches qui ne sont plus reliées à la masse ou à l'alimentation mais à une variable logique. Dans les équations des réseaux  $n$  et  $p$  de ce circuit, chaque branche de transmission correspond au produit logique des transistors connectés en série multiplié par la variable logique:

$$n(z) = [\bar{a} \bar{b}] c + [\bar{a} b] \bar{c} \quad (17)$$

$$p(z) = [a b] c + [a \bar{b}] \bar{c} + \bar{a} \quad (18)$$

Représenté dans une table de Karnaugh (fig. 13), le fonctionnement du circuit détermine un bloc qui dépend de la variable  $c$  (branches 1 et 4), un bloc qui dépend de la variable  $\bar{c}$  (branches 2 et 5) et un bloc de 1 (branche 3). L'équation de la fonction  $z$  réalise la somme logique des deux produits et de la variable relatifs aux trois blocs, multipliés respectivement par  $c$ , par  $\bar{c}$  et par 1:

$$z = [\bar{a} \bar{b}] c + [\bar{a} b] \bar{c} + [a] 1 \quad (19)$$

D'une manière générale, l'équation d'une fonction  $z$  permet toujours de retrouver celles de ses réseaux  $n(z)$  et  $p(z)$ :  $n(z)$  correspond aux expressions de  $z$  multipliées par des variables et par 0;  $p(z)$  découle des expressions de  $z$  multipliées par des variables et par 1,

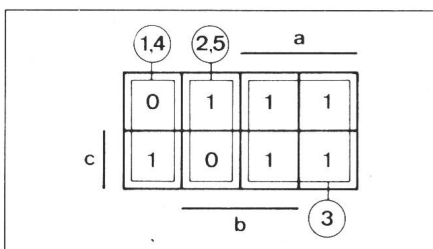


Fig. 13

lorsqu'on complémente chacune de leurs variables placées entre crochets.

### 2.2 Méthode

Le concept de blocs dépendant d'une variable, lié aux branches de transmission, permet de prolonger les méthodes de simplification des fonctions complètement ou incomplètement définies (§ 1.6 et 1.12):

1. introduire la fonction logique résultant de la simplification par blocs de 0 et de 1 dans une table de Karnaugh;
2. rechercher visuellement et encadrer tous les blocs dont les 0 et les 1 ne dépendent que d'une variable en éliminant tous ceux qui sont totalement inclus dans un bloc plus grand;
3. choisir parmi ces blocs ceux qui permettent de supprimer simultanément un impliquant premier de la fonction et un impliquant premier de la fonction inverse dans le polynôme final obtenu lors de la simplification par blocs de 0 et de 1;
4. déterminer un polynôme final composé de l'ensemble des blocs choisis et des impliquants premiers qui n'ont pas été supprimés; dans ce polynôme, les blocs choisis, les impliquants premiers de la fonction et les impliquants premiers de la fonction inverse sont respectivement multipliés par la variable dont ils dépendent, par 1 et par 0.

### 2.3 Exemple

Pour illustrer la méthode, prenons la fonction

$$z = [b c + \bar{a} \bar{b} \bar{c}] 0 + [\bar{b} c + b \bar{c} + a \bar{b}] 1 \quad (20)$$

qui correspond à une des solutions obtenues lors de la simplification d'une fonction complètement définie (§ 1.7). Introduite dans la table de Karnaugh de la figure 14, cette fonction admet deux blocs de quatre cases et trois blocs de deux cases qui ne dépendent que d'une variable. Un choix judicieux parmi ces blocs permet de supprimer au mieux deux impliquants premiers de la fonction et deux impliquants premiers de la fonction inverse. La figure 15 propose une des trois solutions possibles. Cette solution correspond à l'équation

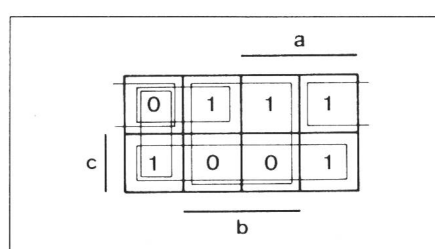


Fig. 14

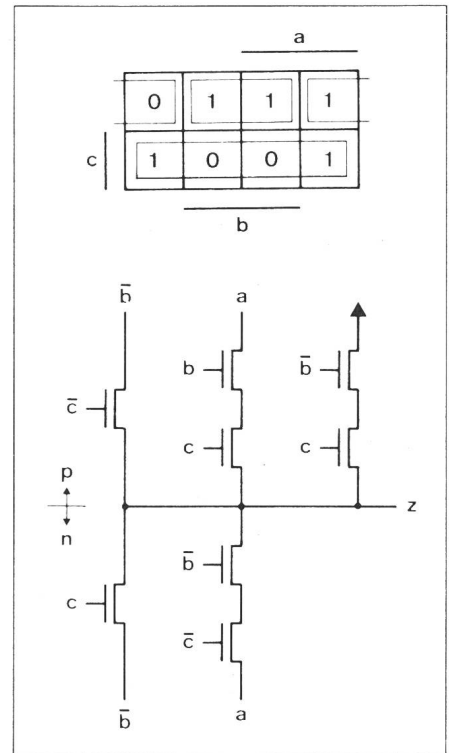


Fig. 15

$$z = [c] \bar{b} + [\bar{b} \bar{c}] a + [b \bar{c}] 1 \quad (21)$$

Elle définit les équations des réseaux

$$n(z) = [c] \bar{b} + [\bar{b} \bar{c}] a \quad (22)$$

$$p(z) = [\bar{c}] \bar{b} + [b c] a + \bar{b} c \quad (23)$$

et conduit au schéma électronique indiqué.

Pour une fonction donnée, le choix d'une solution avec ou sans branches de transmission dépend des critères d'intégration. La première solution comporte néanmoins toujours un nombre plus faible de transistors.

### 2.4 Propriété

Les équations des réseaux  $n$  et  $p$  d'une fonction donnée  $z$  et de l'inverse  $\bar{z}$  de cette fonction sont liées par les relations

$$n[z(\bar{x}_i)] = p[\bar{z}(x_i)] \quad (24)$$

$$p[z(\bar{x}_i)] = n[\bar{z}(x_i)] \quad (25)$$

dans lesquelles  $x_i$  et  $\bar{x}_i$  représentent respectivement les variables vraies et complémentées des fonctions en question.

### 2.5 Exemple

Prenons la fonction de l'équation (19), son inverse

$$\bar{z} = [\bar{a} \bar{b}] \bar{c} + [\bar{a} b] c + [a] 0 \quad (26)$$

et illustrons la propriété en vérifiant que les équations des réseaux de ces fonctions satisfont bien les relations (24) et (25). On obtient effectivement

$$n(z) = [\bar{a} \bar{b}] c + [\bar{a} b] \bar{c} \quad (27)$$

$$p(z) = [a b] c + [a \bar{b}] \bar{c} + \bar{a} \quad (28)$$

$$n(\bar{z}) = [\bar{a} \bar{b}] \bar{c} + [\bar{a} b] c + a \quad (29)$$

$$p(\bar{z}) = [a b] \bar{c} + [a \bar{b}] c \quad (30)$$

### 2.6 Application: fonction majorité

Si l'on ne dispose que des variables sous forme vraie pour réaliser la fonction majorité de trois variables:

$$\begin{aligned} \text{MAJ}(a, b, c) = & \\ & [\bar{a} \bar{b} + \bar{a} \bar{c} + \bar{b} \bar{c}] 0 \\ & + [a b + a c + b c] 1 \end{aligned} \quad (31)$$

il est préférable d'effectuer la synthèse de la fonction complémentaire MAJ(a, b, c) puis de l'inverser. On est ainsi conduit à l'utilisation d'un inverseur au lieu de trois.

## 3. Circuit à sortie 3-états

### 3.1 Etat à haute impédance

Le multiplexage entre plusieurs circuits logiques se simplifie si chacun de ces circuits est pourvu d'une sortie à 3-états capable de générer un état à haute impédance  $\nabla$  en plus des états logiques 0 et 1. Au niveau de la table de Karnaugh, un état à haute impédance du circuit correspond à une case remplie d'un symbole  $\nabla$ .

### 3.2 Méthode

Le concept d'état à haute impédance  $\nabla$ , lié aux circuits à sortie 3-états, permet de prolonger les méthodes de simplification précédentes (§ 1.6, 1.12 et 2.2):

1. introduire la fonction logique du circuit à sortie 3-états dans une table de Karnaugh;
2. appliquer les méthodes propres aux systèmes combinatoires sans sortie 3-états en procédant de sorte que chacun des  $\nabla$  de la table n'appartienne ni à un bloc de 0, ni à un bloc de 1, ni à un bloc qui dépend d'une variable.

### 3.3 Exemple

La table de Karnaugh de la figure 16 illustre la simplification par blocs de 0 et de 1 d'une fonction z réalisée par un

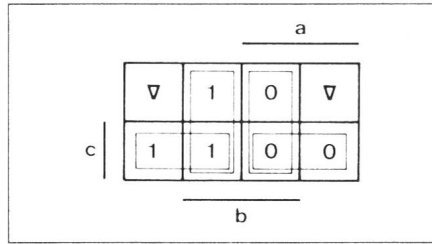


Fig. 16

circuit logique à sortie 3-états. Les quatre blocs de cette simplification conduisent à l'équation

$$z = [a b + a c] 0 + [\bar{a} b + \bar{a} c] 1 \quad (32)$$

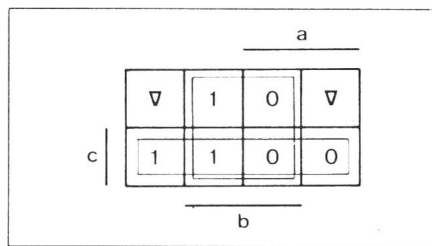


Fig. 17

Dans la table de Karnaugh de la figure 17, deux blocs, dont les 0 et les 1 ne dépendent que de la variable  $\bar{a}$ , ont permis de supprimer les quatre blocs de 0 et de 1 de la simplification précédente. L'équation de la fonction devient

$$z = [b] \bar{a} + [c] \bar{a} = [b + c] \bar{a} \quad (33)$$

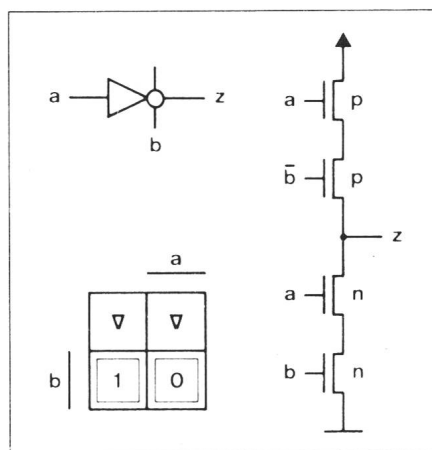


Fig. 18

### 3.4 Application: inverseur à sortie 3-états

Le symbole logique de l'inverseur à sortie 3-états apparaît à la figure 18. La table de Karnaugh, qui décrit le fonctionnement de cet opérateur, permet d'en déduire l'équation

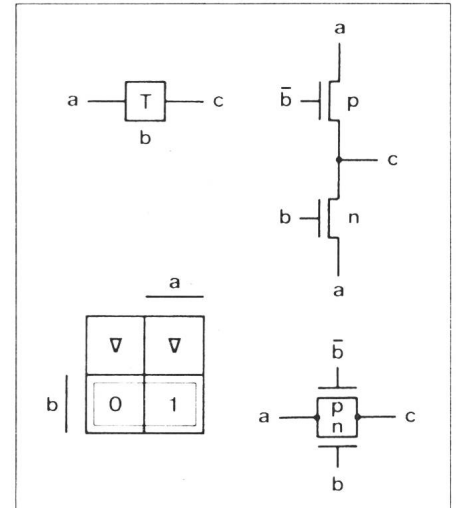


Fig. 19

$$z = [a b] 0 + [\bar{a} b] 1 \quad (34)$$

Le schéma électronique de l'inverseur à sortie 3-états matérialise les équations des réseaux

$$n(z) = a b; p(z) = a \bar{b} \quad (35)$$

### 3.5 Application: porte de transmission

La figure 19 donne le symbole logique de la porte de transmission. Si l'on considère a comme la variable d'entrée et b comme la variable de commande de cet élément, la table de Karnaugh de la figure 19 précise les états de sa variable de sortie c. Le bloc, dont le 0 et le 1 ne dépendent que de a, détermine l'équation de c:

$$c = [b] a \quad (36)$$

puis celles des réseaux n et p:

$$n(c) = [b] a; p(c) = [\bar{b}] a \quad (37)$$

Le schéma électronique de la figure 19 matérialise les relations (37). Par simple transformation graphique, on trouve le schéma électronique final de la porte de transmission. Les rôles symétriques joués par les variables a et c de ce schéma indiquent bien qu'il s'agit d'un élément électronique bidirectionnel équivalent à un interrupteur, dont l'ouverture et la fermeture sont commandés par la variable b.

### 3.6 Condition impossible

La fonction logique réalisée par un circuit présente des conditions impossibles (en anglais: don't happen conditions) lorsqu'il existe un certain nombre d'états qui ne peuvent jamais



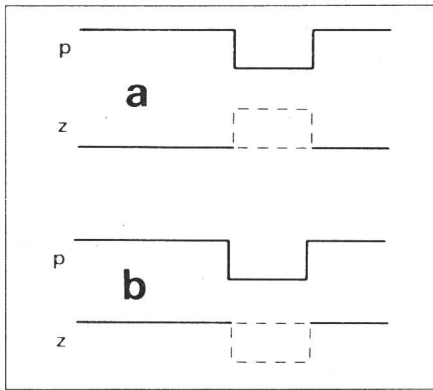


Fig. 20

se présenter à l'entrée du circuit en question. Au niveau de la table de Karnaugh, la condition impossible correspond à une case remplie d'un tiret (-). Elle peut être remplacée par 0, 1,  $\nabla$  ou 0 et 1 simultanément lors de la synthèse.

## 4. Circuit préchargé

### 4.1 Principe de fonctionnement

Un circuit préchargé à 0 est un système combinatoire dont le fonctionnement en deux phases dépend de la variable de précharge  $p$ . Lorsque  $p = 1$ , la variable de sortie  $z$  est préchargée à 0 par conduction du réseau  $n$  (fig. 20a). Lorsque  $p = 0$ , en fonction du blocage des réseaux  $n$  et  $p$  ou de la conduction du réseau  $p$ ,  $z$  conserve temporairement l'état 0 grâce à la capacité de sortie du circuit ou prend l'état 1 par transmission de l'alimentation.

Pour le circuit préchargé à 1 par conduction du réseau  $p$ , lorsque  $p = 1$ , la conservation temporaire de l'état 1 résulte du blocage des deux réseaux et le passage à l'état 0 de la conduction du réseau  $n$ , lorsque  $p = 0$  (fig. 20b).

Un circuit préchargé ne constitue donc plus un système combinatoire statique dont la fonction logique reste constamment disponible mais un système combinatoire dynamique dont la fonction logique n'apparaît qu'à l'annulation de la variable de précharge.

Pour un nombre élevé de variables et au prix d'une plus grande complexité d'emploi, le circuit préchargé utilise approximativement la moitié des transistors du circuit sans précharge équivalent.

### 4.2 Méthode

La table de vérité d'un circuit préchargé correspond à celle du circuit sans précharge équivalent pour  $p = 0$  et définit l'état de précharge pour  $p =$

1. La simplification du circuit s'opère dans une table de Karnaugh:

1. introduire la fonction logique du circuit préchargé dans la table de Karnaugh;
2. souligner les 0 ou les 1 de la table relatifs à la précharge;
3. souligner les 0 ou les 1 de la table qui diffèrent de la valeur préchargée;
4. appliquer les méthodes propres aux systèmes combinatoires sans précharge en considérant que chacun des 0 ou des 1 non soulignés de la table peut également correspondre à un état à haute impédance  $\nabla$ ; pour les circuits à branches de transmission, il n'est cependant plus nécessaire que les blocs dépendant d'une variable suppriment chacun un bloc de 0 et un bloc de 1; de plus, les branches de transmission sont superflues dans le réseau de précharge, le transistor de précharge assure seul la transmission de la masse ou de l'alimentation.

### 4.3 Application:

#### porte NAND préchargée à 1

Définie par sa table de Karnaugh (fig. 21), la porte NAND préchargée à 1 correspond à l'équation

$$z = [a b \bar{p}] 0 + [p] 1 \quad (38)$$

Le schéma électronique de la porte matérialise les équations des réseaux

$$n(z) = a b \bar{p}; p(z) = \bar{p} \quad (39)$$

### 4.4 Application:

#### porte NAND préchargée à 0

Effectuée dans la table de Karnaugh de la figure 22, la simplification de la

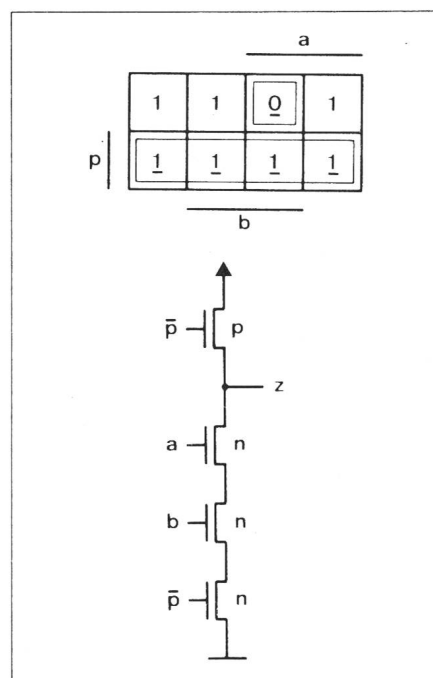


Fig. 21

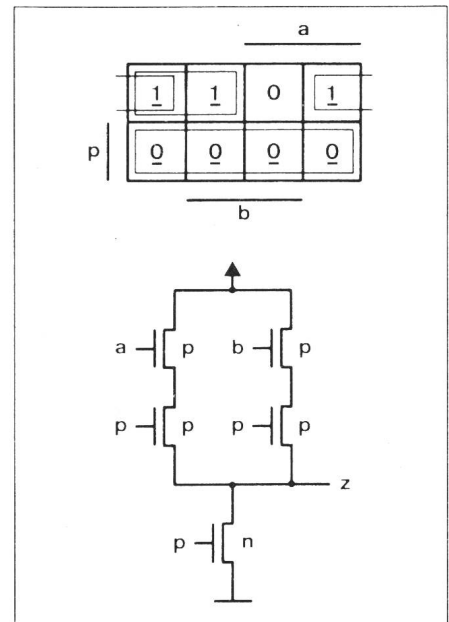


Fig. 22

porte NAND préchargée à 0 conduit à l'équation de la fonction

$$z = [p] 0 + [\bar{a} \bar{p} + \bar{b} \bar{p}] 1 \quad (40)$$

puis à celles des réseaux

$$n(z) = p; p(z) = a p + b p \quad (41)$$

Le schéma électronique matérialise les relations (41).

### 4.5 Minimisation du nombre de transistors

Dans un circuit préchargé, le nombre des transistors commandés par la variable de précharge peut toujours se réduire à deux. Le schéma électronique du circuit préchargé se ramène alors à ces deux transistors et à l'un des réseaux  $n$  ou  $p$  du circuit sans précharge équivalent. Pour la porte NAND préchargée à 0, l'obtention d'un schéma de ce type (fig. 23) passe

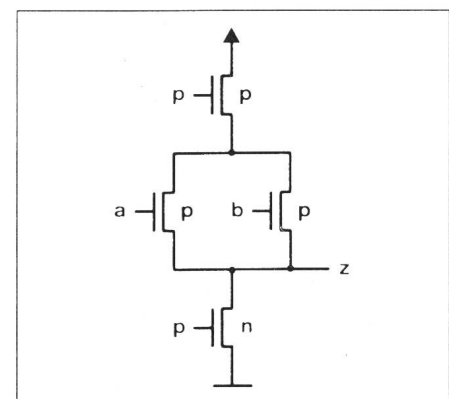


Fig. 23

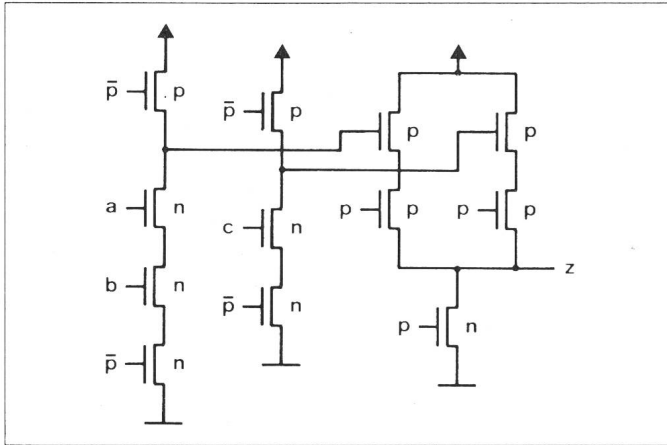


Fig. 24

par la mise en évidence de la variable de précharge  $p$  dans la relation (41):

$$p(z) = p(a + b) \quad (42)$$

De façon générale, les mises en évidence opérées dans les équations des réseaux d'un circuit quelconque permettent toujours de minimiser le nombre de transistors du circuit en question. Cette minimisation s'effectue cependant par adjonction de lignes d'interconnexion autres que celles propres à la fonction, à la masse et à l'alimentation.

#### 4.6 Systèmes combinatoires dynamiques

La conception de systèmes combinatoires dynamiques repose sur l'emploi de circuits logiques préchargés.

La première façon de procéder [2] consiste à utiliser le signal généré par un circuit préchargé à 1 comme variable d'entrée d'un circuit préchargé à 0. La figure 24 illustre ainsi un système combinatoire dynamique obtenu par assemblage d'une porte NAND préchargée à 1, d'un inverseur préchargé à 1 et d'une porte NAND préchargée à 0.

La seconde manière de faire [3] privilégie l'emploi des transistors  $n$  au détriment de celui des transistors  $p$ . Elle revient à n'utiliser que des circuits préchargés à 1 dont on complémente la sortie par un inverseur sans précharge. La figure 25 représente un système combinatoire dynamique de ce type qui réalise la même fonction logique que celui de la figure précédente. Ce système se compose d'une porte NAND préchargée à 1, d'une porte NOR préchargée à 1 et de deux inverseurs sans précharge.

Dans les systèmes combinatoires dynamiques considérés, un circuit pré-

chargé peut se simplifier si toutes ses variables d'entrée sont de nature dynamique. Un tel circuit admet la suppression de l'ensemble des transistors commandés par la variable de précharge, à l'exception du transistor qui assure la précharge. Les deux transistors  $p$  commandés par  $p$  de la porte NAND préchargée à 0 (fig. 24) sont ainsi superflus.

## 5. Schémas logiques

### 5.1 Circuit CMOS

Les variables d'entrée d'un circuit CMOS apparaissent sous la même forme, vraie ou complétement, dans les réseaux  $n$  et  $p$  que dans le terme multiplié par 0 de l'équation du circuit. La fonction réalisée par le circuit CMOS correspond de plus à l'inverse du terme multiplié par 0. Pour obtenir le schéma logique d'un circuit CMOS, il suffit par conséquent de partir de l'équation du circuit et de représenter l'inverse du terme multiplié par 0.

L'équation (3) du circuit CMOS de la figure 3 conduit ainsi au schéma logique de la figure 26.

### 5.2 Circuit à branches de transmission

Le schéma logique d'un circuit à branches de transmission généralise le concept de porte de transmission (§ 3.5) et l'applique à l'ensemble des branches du circuit.

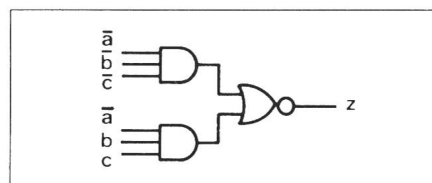


Fig. 26

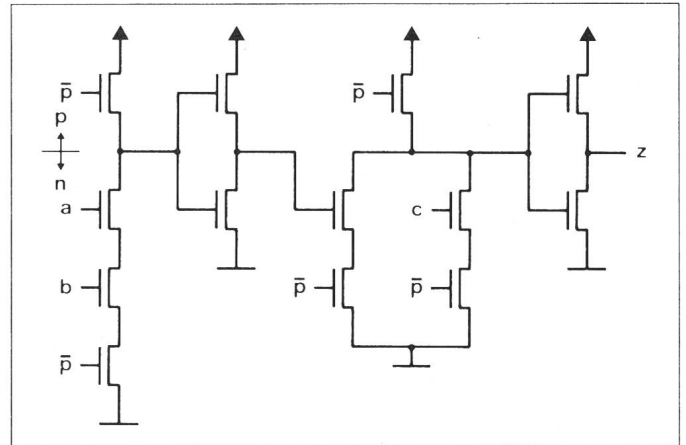


Fig. 25

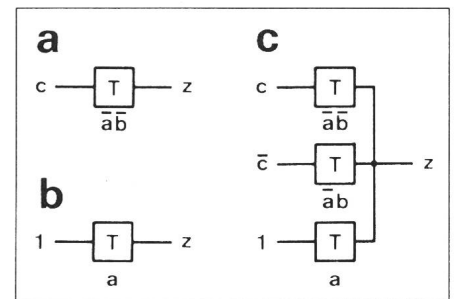


Fig. 27

Une paire de branches de transmission, déterminée dans la table de Karnaugh par un bloc dont les 0 et les 1 ne dépendent que d'une variable, correspond à une porte de transmission commandée par la variable ou l'expression qui définit le bloc. La porte de transmission de la figure 27a représente ainsi les branches 1 et 4 du circuit de la figure 12.

Une branche, déterminée dans la table de Karnaugh par un bloc de 0 ou par un bloc de 1, correspond à une porte de transmission qui ne comporte respectivement que des transistors  $n$  ou des transistors  $p$ . Cette porte, commandée par la variable ou l'expression qui définit le bloc, est connectée à la masse ou à l'alimentation. La branche 3 du circuit de la figure 12 conduit ainsi à la porte de transmission de la figure 27b.

Pour obtenir le schéma logique d'un circuit à branches de transmission, il suffit par conséquent de partir de l'équation du circuit et de représenter chacun de ses termes par une porte de transmission commandée par la variable ou l'expression placée entre crochets et connectée à la variable ou à la constante placée hors des crochets.

L'équation (19) du circuit à branches de transmission de la figure 12 en-



gendre ainsi le schéma logique de la figure 27c.

### 5.3 Circuit à sortie 3-états

Le schéma logique d'un circuit à sortie 3-états reprend et généralise le symbolisme adopté pour l'inverseur à sortie 3-états (§ 3.4). Ce schéma spécifie en particulier le domaine binaire du circuit, c'est-à-dire l'ensemble des états d'entrée pour lesquels sa sortie vaut 0 ou 1.

Pour obtenir le schéma logique d'un circuit à sortie 3-états sans branches de transmission, il suffit de partir de l'équation du circuit, de représenter l'inverse du terme multiplié par 0 et de préciser le domaine binaire de la fonction en calculant la somme logique des expressions placées entre crochets. L'équation (32) définit ainsi le domaine binaire

$$B(z) = ab + ac + \bar{a}b + \bar{a}c = b + c \quad (43)$$

et correspond au schéma logique de la figure 28a.

On peut aussi expliciter le domaine binaire en procédant à une mise en évidence dans l'équation du circuit:

$$z = [b + c]([a]0 + [\bar{a}]1) \quad (44)$$

Le schéma logique correspondant (fig. 28b) découle alors d'un autre terme multiplié par 0.

Lorsque le circuit à sortie 3-états comporte des branches de transmission, on procède par portes de transmission. L'équation (33) conduit ainsi au schéma logique de la figure 28c.

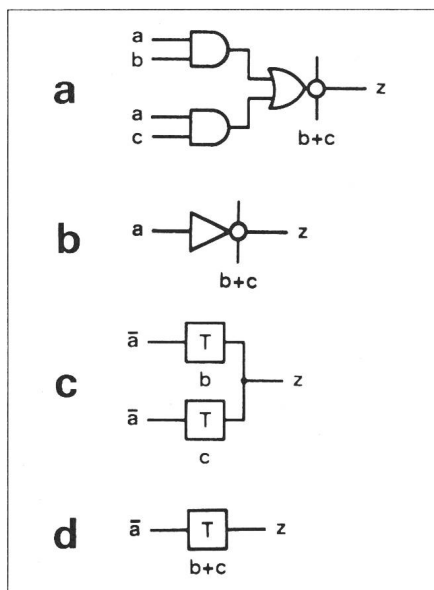


Fig. 28

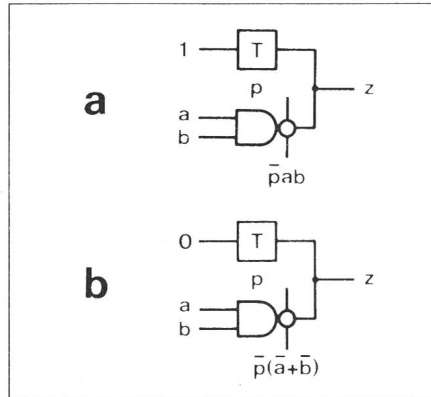


Fig. 29

On peut également réécrire l'équation du circuit en regroupant les termes qui résultent de blocs dépendant d'une même variable (équation 33). Le schéma logique correspondant apparaît à la figure 28d.

### 5.4 Circuit préchargé

Le schéma logique d'un circuit préchargé applique le concept de porte de transmission au réseau de précharge et le symbolisme des circuits à sortie 3-états à l'autre réseau.

Pour obtenir le schéma logique d'un circuit préchargé à 1 sans branches de transmission, il suffit de dessiner une porte de transmission commandée par la variable de précharge  $p$  et connectée à 1 puis, à partir de l'équation du circuit, de représenter l'inverse du terme multiplié par  $\bar{p}$  en considérant le terme multiplié par 0 comme domaine binaire. L'équation (38) de la porte NAND préchargée à 1 (§ 4.3) correspond ainsi au schéma logique de la figure 29a.

Le schéma logique d'un circuit préchargé à 0 sans branches de transmission se réduit au dessin d'une porte de transmission commandée par  $p$  et connectée à 0 puis à la représentation, sous forme d'une fonction inversée, du terme multiplié par  $\bar{p}$  en considérant le terme multiplié par 1 comme domaine binaire. L'équation (40) de la porte NAND préchargée à 0 (§ 4.4) conduit ainsi au schéma logique de la figure 29b.

Lorsque le circuit préchargé comporte des branches de transmission, on procède par portes de transmission.

### 5.5 Equivalence fonctionnelle

Opérée dans la table de Karnaugh de l'inverseur à sortie 3-états (fig. 30), la simplification par un bloc dépendant de la variable  $a$  conduit à l'équation

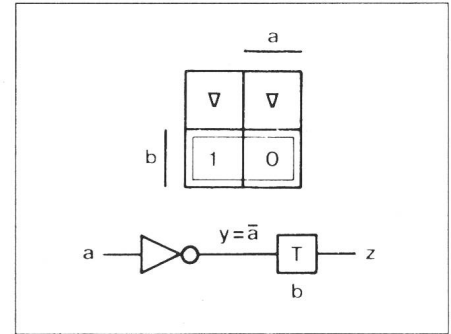


Fig. 30

$$z = [b] \bar{a} \quad (45)$$

puis au schéma logique de la figure 30. Ce circuit, fonctionnellement équivalent à l'inverseur à sortie 3-états de la figure 18, comporte toutefois une variable interne  $y = \bar{a}$  supplémentaire. D'une manière générale, lorsque la sortie d'un circuit CMOS attaque une porte de transmission d'un circuit à branches de transmission, on peut toujours supprimer une variable interne en ramenant la porte de transmission dans le circuit CMOS et obtenir ainsi un circuit à sortie 3-états. Au niveau des équations, cela revient à introduire l'expression du circuit CMOS dans celle de la porte de transmission. Pour l'inverseur à sortie 3-états de la figure 30, l'inverseur et la porte de transmission vérifient respectivement les relations

$$y = [a]0 + [\bar{a}]1 \quad (46)$$

$$z = [b]y \quad (47)$$

L'équation de l'inverseur à sortie 3-états de la figure 18 s'obtient en substituant la variable  $y$  de la relation (47) par son expression (46):

$$\begin{aligned} z &= [b]([a]0 + [\bar{a}]1) \\ &= [ab]0 + [\bar{a}b]1 \end{aligned} \quad (48)$$

Le circuit à sortie 3-états (fig. 31a), obtenu par la mise en série de deux portes de transmission, réalise la fonction globale représentée dans la table de Karnaugh de la figure 31b. Effectuée à l'aide d'un bloc qui dépend de la variable  $c$ , la simplification de cette fonction détermine l'équation

$$z = [ab]c \quad (49)$$

qui correspond au schéma logique de la figure 31c. En d'autres termes, il suf-

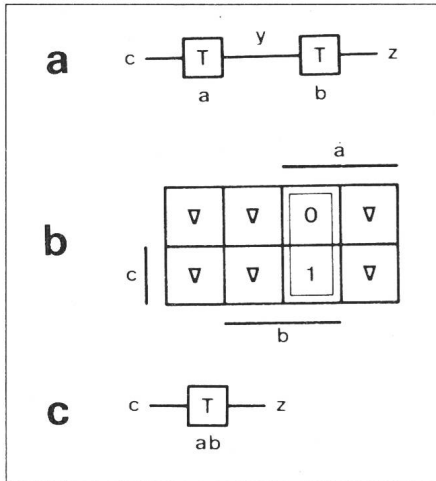


Fig. 31

fit de commander une porte de transmission par le produit logique des variables de commande de deux portes de transmission connectées en série pour supprimer une variable interne tout en conservant l'équivalence fonctionnelle du circuit. Au niveau des équations, il convient d'introduire l'expression de la première porte de transmission dans celle de la seconde. Dans le cas de la figure 31a, la suppression de la variable interne  $y$  dans les relations

$$y = [a] c \quad (50)$$

$$z = [b] y \quad (51)$$

propres aux deux portes de transmission conduit bien à l'équation du circuit à sortie 3-états de la figure 31c:

$$z = [b]([a] c) = [a b] c \quad (52)$$

#### Bibliographie

- [1] *D. Mange*: Analyse et synthèse des systèmes logiques. Traité d'électricité de l'EPFL. Vol. V. St-Saphorin, Editions Georgi, 1978.
- [2] *N.F. Goncalves and H. De Man*: NP-CMOS: A race-free-dynamic CMOS technique for pipelined logic structures. Eighth European Solid-State Circuits Conference, Brussel, 1982 (ESSCIRC '82). Digest of technical papers p. 141..144.
- [3] *R.H. Krambeck, C.M. Lee and H.S. Law*: High-speed compact circuits with CMOS. IEEE Journal of Solid-State Circuits 17 (1982) 3, p. 614..619.