

Titre: Technique hiérarchique de vérification et reconfiguration de circuits
intégrés

Auteur: Claude Cyr

Date: 1989

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Cyr, C. (1989). Technique hiérarchique de vérification et reconfiguration de circuits intégrés [Master's thesis, Polytechnique Montréal]. PolyPublie.
Citation: <https://publications.polymtl.ca/57948/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/57948/>
PolyPublie URL:

Directeurs de recherche:
Advisors:

Programme: Unspecified
Program:

UNIVERSITE DE MONTREAL

TECHNIQUE HIERARCHIQUE DE VERIFICATION
ET RECONFIGURATION DE CIRCUITS INTEGRES

par

Claude Cyr

DEPARTEMENT DE GENIE ELECTRIQUE
ECOLE POLYTECHNIQUE

MEMOIRE PRESENTE EN VUE DE L'OBTENTION
DU GRADE DE MAITRISE ES SCIENCES APPLIQUEES (M.Sc.A.)

avril 1989

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-50197-9

UNIVERSITE DE MONTREAL

ECOLE POLYTECHNIQUE

Ce mémoire intitulé:

TECHNIQUE HIERARCHIQUE DE VERIFICATION
ET RECONFIGURATION DE CIRCUITS INTEGRES

présenté par Claude Cyr

en vue de l'obtention du grade de: M. Sc. A.

a été dûment accepté par le jury d'examen constitué de:

M. Jean-Louis Houle, Ph.D., président

M. Yvon Savaria, Ph.D., directeur de recherche

Mme. Bozena Kaminska, Ph.D., examinateur

SOMMAIRE

L'augmentation de la superficie d'une puce augmente la quantité moyenne de défauts ou la probabilité de défauts d'une puce. Ceci a pour conséquence de réduire le rendement et d'augmenter le coût d'une puce utilisable (valeur commerciale). L'utilisation de redondance a été préconisée comme moyen de restreindre l'effet de l'augmentation de la superficie sur le rendement. Nous souscrivons à cette tendance et proposons HCDE: une technique hiérarchique d'amélioration du rendement de circuits intégrés de grande superficie.

Le second chapitre décrit HCDE, la technique hiérarchique d'amélioration du rendement de circuits de grande superficie par détection et exclusion de conflits forcés. Nous proposons cette technique comme solution au problème de faible rendement de circuits intégrés de grande superficie. Cette technique comporte une contribution originale: une approche analogique de vérification de circuit hautement tolérante aux défauts, vérifiable et occupant une superficie restreinte.

Au chapitre 3 nous proposons une méthode simple de calcul du rendement en présence de redondance sur la base de statistique de Poisson. Nous proposons aussi un critère

d'évaluation ou facteur de mérite basé sur l'augmentation de fonctionnalité à rendement équivalent obtenue avec HCDE.

Nous validons HCDE aux chapitres 4 et 5 au moyen de deux exemples pratiques: 1-une implantation et 2-un circuit hypothétique de grande superficie. Le quatrième chapitre décrit l'application, une architecture de système et la partie fonctionnelle d'un exemple d'implantation (IMAGE2). Ce chapitre décrit également un exemple de système de 5120 processeurs (quatre niveaux hiérarchiques CDE) afin de démontrer le potentiel de HCDE. Le chapitre 5 présente l'aspect HCDE des exemples du chapitre 4 et une analyse des gains de rendement et facteurs de mérite prévus. Nous concluons au chapitre 6 sur 1-HCDE, 2-les avantages de HCDE et 3-la faisabilité d'un système de traitement primaire d'images performant et économique.

Dans ce mémoire, nous utilisons l'expression circuit complémentaire pour référer, faute de mieux, aux parties de circuit effectuant la vérification et la reconfiguration de la partie fonctionnelle (circuit sans HCDE). L'expression "logique complémentaire" réfère à la classe de fonctionnement au niveau transistor ou chaque cellule est réalisée au moyen de deux groupements complémentaires (N et P) réalisant une fonction identique.

ABSTRACT

The desire to increase circuit area has been limited, up to now, by the low yields obtained. We hereby propose a technique to improve the yield of large area integrated circuits. It uses a hierarchical conflict detection and exclusion scheme (HCDE).

Chapter 1 contains an introduction and a review. The second chapter contains a general description of HCDE. The verification method used in HCDE which is an analog circuit that is defect tolerant, testable, and compact, constitutes an original contribution. The third chapter describes a simple method to evaluate yield using Poisson statistics, and two specific merit factors used in later chapters to evaluate yield and chip functionality gains obtained with HCDE.

In chapters 4 and 5, we validate HCDE by presenting two examples: 1-IMAGE2, an implementation, and 2-a hypothetical example of a large area device containing approximately 25 million transistors. The fourth chapter covers: 1-real time image processing (the application), 2-a global architecture for 1, 3-the IMAGE2 implementation example, and 4-a 5120 processors example with 3

vii.

reconfigurable hierarchical levels. The fifth chapter covers: 1-the HCDE aspects of the examples of chapter 4, 2-a yield and merit factors analysis, and 3-a conclusion on the examples. Chapter 6 contains our conclusions about: 1-HCDE, 2-advantages of HCDE, and 3-the feasibility of an economical real time image processing system.

REMERCIEMENTS

Nous voulons remercier ceux et celles qui, de près ou de loin, ont contribué à ce travail, dont:

Yvon Savaria, Le Centre de Recherche Informatique de Montréal, Jean-Louis Houle, Le Département de Génie Electrique de l'Ecole Polytechnique de Montréal, Robert Dufresne et Normand Allard pour leur contribution financière; l'Ecole Polytechnique de Montréal, la "Canadian Microelectronics Corporation", Madeleine Legault et Daniel Audet pour leur support matériel;

Yvon Savaria dont la direction a rendu ce travail possible; Yvon Savaria et Jean-Louis Houle sans qui le laboratoire ITGE n'aurait pas existé; Daniel Audet et Gilles Chouinard, qui ont partagé avec l'auteur les nombreuses heures vouées à la réalisation d'une puce IMAGE2 fonctionnelle et validée;

Hélène Dallaire qui a su garder le sourire, même quand la demande dépassait les capacités de production; tous ceux qui ont eu la gentillesse, aux moments critiques: de céder une console, répondre à une question, déverminer un logiciel, libérer de l'espace disque, etc...

TABLE DES MATIERES

	PAGE
SOMMAIRE	iv
ABSTRACT	vi
REMERCIEMENT	viii
CHAPITRE I - INTRODUCTION	1
CHAPITRE II - RECONFIGURATION AUTOMATIQUE HIERARCHISEE	12
II-1 préambule	12
II-2 HCDE pour un module homogène unidimensionnel	15
II-3 détection d'un module défectueux	18
II-4 registre de diagnostic	28
II-5 reconfiguration latérale	35
II-6 reconfiguration verticale	37
II-7 expansion hiérarchique	49
II-8 application	57
II-9 conclusion	66

	PAGE
CHAPITRE III - ESTIMATION DU RENDEMENT	69
III-1 introduction	69
III-2 estimation du rendement	74
III-3 estimation de la superficie	79
III-4 critère d'optimalité	83
III-5 travaux futurs	87
CHAPITRE IV - EXEMPLES, PARTIE FONCTIONNELLE	89
IV-1 introduction	89
IV-2 traitement d'images en temps réel	93
IV-3 architecture générale du système envisagé	108
IV-4 partie fonctionnelle d'IMAGE2	124
IV-5 expansion en système plus important	131
IV-6 travaux futurs	141
CHAPITRE V - EXEMPLES, PARTIE HCDE	144
V-1 introduction	144
V-2 HCDE dans IMAGE2	145
V-3 HCDE aux niveaux supérieurs	166
V-4 mesures et estimés de superficies	173
V-5 analyse du rendement et des facteurs de mérite	178
V-6 conclusion et travaux futurs	185

	PAGE
CHAPITRE VI - CONCLUSION	187
VI-1 conclusions générales	187
VI-2 avantages de HCDE	189
VI-3 système économique de traitement d'images en temps réel	193
BIBLIOGRAPHIE	195

CHAPITRE I

INTRODUCTION

Les besoins pour des circuits de plus en plus complexes dépassent les capacités des manufacturiers à proposer des procédés à granularité plus dense. Il nous a donc semblé pertinent de rechercher des méthodes douces d'augmentation de la fonctionnalité de puces. Dans ce mémoire, nous ne considérons pas les techniques modifiant la granularité ou le nombre de couches d'un procédé. Nous avons cherché des moyens d'augmenter la superficie des circuits plutôt que de modifier les procédés de fabrication.

L'augmentation de superficie d'un circuit a cependant un impact marqué sur son rendement. Le rendement est une mesure du nombre de bons circuits par rapport à la quantité de circuits fabriqués. Une partie des puces sont en effet rejetées en conséquence de défauts de fabrication hors du contrôle direct du fabricant. L'augmentation de la superficie d'une puce augmente donc la probabilité de présence ou la quantité moyenne de défauts par puce. Ceci a pour conséquence d'augmenter le coût d'une puce fonctionnelle.

2.

L'utilisation de redondance a été préconisée par plusieurs comme moyen de restreindre l'effet de l'augmentation de la superficie sur le rendement. Nous souscrivons à cette tendance et c'est dans cet esprit que nous proposons au second chapitre une technique d'amélioration du rendement de circuits intégrés. Considérant que le gain de rendement obtenu est limité par la partie intolérante aux défauts d'un circuit, nous avons prévu la possibilité d'appliquer successivement notre technique en utilisant une approche hiérarchique. Singh [1] nous [2] a précédé de peu en cette idée. Nous croyons que plusieurs techniques de tolérance aux défauts peuvent collaborer en s'appliquant hiérarchiquement sur elles-mêmes ou entre-elles, bénéficiant ainsi les unes des autres et produisant alors des circuits plus performants.

Notre technique vise d'abord des circuits comportant un certain degré de régularité. Mead et Conway [3] et d'autres auteurs (exemple: Weste et Eshraghian [4]) ont discuté des avantages de restreindre le travail de conception par l'utilisation d'approches structurées. Ils préconisent ainsi l'utilisation de techniques: de hiérarchie, de modularité, de régularité et de localité. Notre travail au laboratoire nous a convaincu des avantages de telles techniques. Ceci était prévisible, toute économie

3.

de temps de conception permet d'augmenter la probabilité de succès en amenant à une échelle humaine la description logique, la compréhension, le dessin des masques, la vérification, le déverminage et l'utilisation des circuits. Ceci est encore plus important lorsque le nombre de composants augmente. La capacité humaine à comprendre des problèmes complexes étant limitée, alors que l'envergure des circuits (nombre de transistors) croît de plus en plus rapidement (exponentiellement).

Nous reconnaissons la régularité comme une des techniques les plus efficaces pour augmenter la puissance (calcul, mémoire, communication) des circuits, tout en limitant les efforts de développement. En effet, l'augmentation du nombre de cellules répétées augmente la performance d'un circuit sans modifier sa complexité conceptuelle. Puisque les cellules sont les mêmes, seul le nombre de cellules change. Il nous est donc apparu naturel de considérer, comme premier objectif, l'amélioration du rendement de circuits de grande superficie dont la logique était régulière.

Nous sommes convaincus que cette limite n'est pas aussi restrictive qu'il semble à première vue (section 2.9). Ceci, parce que, à notre connaissance, une grande proportion

4.

des circuits de grande superficie (présents ou à venir) sont appelés à comporter une proportion importante de circuits réguliers, dont le rendement peut être amélioré en utilisant HCDE. Dans cette voie, nous mentionnons la régularité binaire (nombre de bits par mot) unidimensionnelle des unités arithmétiques des circuits microprocesseurs, micro-ordinateurs, processeurs RISC (reduced instruction set computer), coprocesseurs arithmétiques à virgule flottante (ex: familles 8087 et 68881), processeurs vectoriels, etc; la régularité binaire bidimensionnelle des mémoires vives statiques et dynamiques, des unités à logique programmables, des multiplieurs, des multiplieurs-accumulateurs, des processeurs de traitement de signal ("DSP"), etc; la régularité de voisinage/image des systèmes de traitement SIMD (single instruction multiple data: instruction commune données multiples), la régularité d'opérateur des systèmes de traitement MIMD (multiple instruction, multiple data: instructions multiples et données multiples) et de certains processeurs vectoriels, etc. Tous ces circuits sont réguliers selon les modes identifiés et notre liste n'est pas exhaustive.

Nous sommes convaincus que l'amélioration du rendement des groupements réguliers de circuits aura un effet favorable sur le rendement d'une puce, en relation

directe avec l'importance relative des superficies régulières et irrégulières de celle-ci (chapitre 5).

Le but du troisième chapitre est de présenter une formulation efficace du rendement de circuits de grande superficie auxquels on a appliqué HCDE. Nous cherchons une formulation relativement simple et pouvant s'adapter à un éventuel algorithme automatique de sélection des paramètres HCDE optimaux d'un circuit.

Savaria [5] présente une synthèse sur l'estimation du rendement sans redondance. Nous ne reprendrons pas ici l'énumération des nombreuses contributions en ce domaine. Certains auteurs ont déjà présenté des résultats sur l'estimation du rendement en présence de redondance dont Koren et al [6]. Les formulations proposées varient d'un auteur à l'autre suivant la nature des modèles statistiques utilisés. Chacune des approches cherche à tenir compte de la nature stochastique (statistique) de l'occurrence des défauts.

Certaines formulations comme celle de Schvan et al [7] estiment séparément les rendements de plusieurs modes de défauts, utilisant des paramètres D_i et α_i différents pour chacun. Certains types de défauts peuvent affecter plus

6.

qu'une couche (ex: court-circuit entre deux couches, etc) ou plus qu'un noeud (ex: 1-court-circuit entre deux lignes, 2-lignes parallèles ouvertes par une poussière commune). Par conséquent la modélisation des modes de défauts est beaucoup plus complexe qu'il apparaît à première vue à cause d'un phénomène d'interdépendance entre l'occurrence des modes.

Thibeault et al [8] ont présenté une formulation du rendement tenant compte du facteur d'entassement ("clustering") des défauts. Nous respectons ces travaux et encourageons leur poursuite. Cependant, pour valider HCDE, nous avons besoin d'une formulation simple et complète (plus que deux niveaux hiérarchiques). Nous proposons donc (section 3.2) une formulation plus simple du rendement en présence de redondance, basée sur une statistique de Poisson sans tenir compte du facteur d'entassement.

Notre formulation conduit nécessairement à des estimés pessimistes de rendement. Elle a cependant l'avantage de permettre de comparer, à faible coût de calcul, une grande quantité de solutions possibles. Elle pourra donc être utile comme partie d'un éventuel outil d'optimisation des paramètres HCDE. De plus, il était nécessaire d'avoir une méthode formelle d'estimation du rendement et, la formulation de Thibeault [8], disponible

tardivement, avait le désavantage d'être limitée à deux niveaux hiérarchiques au moment de compléter la rédaction de ce mémoire.

Nous sommes d'accord avec Thibeault [8] pour l'utilisation du facteur de mérite de Mangir [9] comme critère dans la recherche d'une solution optimale. Ce facteur donne une mesure relativement simple à calculer de la valeur d'une solution tenant compte du rendement et de la superficie. Nous proposons cependant un critère additionnel de gain de fonctionnalité (FMF: facteur de mérite fonctionnel, section 3.4). Le facteur de mérite fonctionnel est une mesure du gain de fonctionnalité conséquent à l'application de HCDE par rapport à une solution de rendement équivalent sans redondance. Ce dernier critère est essentiellement un facteur de réduction du nombre de puces par système conséquent à l'utilisation de HCDE. Il nous a semblé qu'il serait plus utile au concepteur de système en permettant de comparer deux solutions réalistes, plutôt qu'une solution HCDE réaliste avec une solution irréaliste à rendement médiocre probable dans le cas de circuits de grande superficie sans HCDE (section 3.5).

Nous présentons aux chapitres 4 et 5 une validation pratique de l'avantage de HCDE sous forme de deux exemples

8.

pratiques: une implantation et un exemple de puce de grande superficie. Aux chapitre 4 nous présentons d'abord le domaine d'application: traitement d'images en temps réel. Ce choix répondait naturellement à nos critères de sélection: 1-la puissance de calcul requise, 2-l'actualité du problème, 3-l'ampleur du marché (vision artificielle, robotique, télévision, télématique, imagerie bio-médicale, etc). Pour conserver une chance de succès commercial, nous avons imposé un critère projetant notre application en avance sur son temps. C'est pour cette raison que nous avons imposé les critères additionnels, contradictoires en 1985, traitement 1-global, 2-en temps réel et 3-économique. La faisabilité devait être démontrable au moyen d'un procédé de densité restreinte: $3\mu\text{m}$, 0.25 cm^2 .

Notre but n'était pas de développer le domaine du traitement d'images comme tel. Ce domaine est relativement vaste: nous ne présentons qu'une partie de la très volumineuse littérature sur le sujet [10 à 20]. Notre objectif essentiel était de démontrer la faisabilité d'un système de vision artificielle (opération en temps réel).

Nous effectuons au chapitre 4 un estimé des bandes passantes en utilisant la norme NTSC comme exemple tel que définie dans [21]. Nous séparons alors le domaine en trois

9.

problèmes plus simples: traitement primaire, traitement secondaire et traitement tertiaire. Nous avons ensuite déterminé un exemple de séquence d'opération afin d'estimer l'ordre de grandeur de la puissance de calcul requise pour un traitement en temps réel. Nous présentons alors une architecture globale de système de traitement d'images en temps réel et, plus précisément ensuite, un exemple de système pour le traitement primaire d'images en temps réel. L'architecture que nous préconisons est quelque peu similaire à celle proposée par Riesenbach [22] ou une architecture hautement parallèle utilisant des pipelines est proposée. La puce IMAGE2 est notre implantation d'une partie de système SIMD pour le traitement primaire d'images.

IMAGE2 visait, en relation avec ce mémoire, à valider nos propositions sur la faisabilité d'un système de traitement primaire d'images en temps réel, économique. Cette implantation permettait de mesurer les superficies réelles des circuits proposés, présentant ainsi des valeurs réalistes pour l'estimation du rendement et des facteurs de mérite prévus. IMAGE2 étant de complexité relativement restreinte (12000 transistors), n'est pas prévue pour réaliser le système puissant, en temps réel, proposé. Nous extrapolons donc sur les résultats obtenus avec IMAGE2 pour valider un concept plus performant possible si une

technologie plus dense (1 micron) et de superficie plus importante (12 cm^2) est utilisée. Nous avons prévu qu'une telle technologie serait commercialement disponible au moment de la présentation de ce mémoire.

Même en avril 1989, il n'existe pas de système de traitement primaire d'images en temps réel qui soit à la fois performant (30 à 60 images par seconde, capable d'effectuer plusieurs opérations par pixel) et économique (de \$5000 à \$10000). Il existe des systèmes en temps réel, mais ne traitant régulièrement qu'une fraction des images reçues ou à coût élevé. Notre application est plus exigeante: 1-traitement de chaque image à la fréquence des images reçues (30 à 60 images par seconde: normes NTSC, PAL, SECAM, ou HDTV) et 2-un traitement global, c'est à dire jusqu'à reconnaissance d'objet ou de scène. L'architecture de système proposée à la section 4.3 est suffisamment puissante pour répondre aux besoins les plus exigeants en puissance de calcul et suffisamment flexible (traitement primaire, secondaire et tertiaire) pour simultanément répondre à une grande variété de besoins. IMAGE2, en tant que puce pour le traitement primaire, est une étape dans le développement d'un système de traitement global.

Les chapitres 4 et 5 rencontrent ainsi cinq objectifs: 1-démontrer que HCDE peut être pratiquement implantée, 2-décrire un exemple d'implantation de HCDE (IMAGE2), 3-présenter des résultats réalistes sur les gains de rendement/fonctionnalité possibles sans tenter d'optimiser le facteur de mérite, 4-présenter un concept d'architecture de système de traitement d'image suffisamment performante et flexible pour effectuer un traitement en temps réel et 5-démontrer qu'un système économique est réalisable. Les trois premiers objectifs étayent notre démonstration des avantages qui découlent de HCDE en ce qui a trait au rendement et la fonctionnalité des circuits. Les quatrième et cinquième objectifs représentent une contribution au domaine du traitement d'images où nous croyons qu'un traitement économique, en temps réel est possible par l'utilisation de HCDE.

Nous concluons sur les résultats obtenus au chapitre 6, identifiant les avantages et limites de HCDE.

CHAPITRE2

RECONFIGURATION AUTOMATIQUE HIERARCHISEE (HCDE)

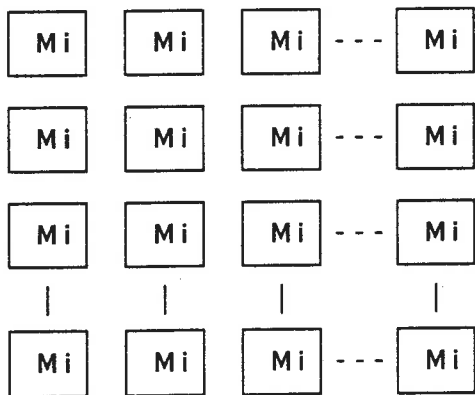
2.1-PREAMBULE

Ce chapitre décrit la technique hiérarchique d'amélioration du rendement de circuits de grande superficie par détection et exclusion de conflits forcés. Nous proposons cette technique comme solution au problème de faible rendement des circuits intégrés de grande superficie. Nous utilisons l'abréviation anglaise "HCDE" ("hierarchical conflict detection and exclusion") pour fins de clarté, respectant ainsi l'antécédence de nos publications anglaises sur le sujet [2].

HCDE vise d'abord à augmenter le rendement des parties régulières de circuits de grande surface. L'amélioration du rendement des groupements réguliers de circuits aura un effet favorable sur le rendement d'une puce, en relation directe avec l'importance relative des superficies régulières et irrégulières de celle-ci.

Voyons maintenant le fonctionnement de HCDE. Nous définissons un groupe homogène comme un circuit ou une partie de circuit composé de modules fonctionnellement

identiques. Lorsqu'au dessin des masques, les modules sont physiquement identiques et disposés suivant un ordre ou une symétrie, le groupe est alors aussi régulier. Cette condition supplémentaire est accessoire pour HCDE, néanmoins, pour faciliter la compréhension, toute figure décrivant un groupe homogène montrera des modules identiques plutôt que seulement fonctionnellement identiques. L'utilisateur de la technique demeure libre de ne pas utiliser cette seconde condition (régularité). La figure 2.1 montre un groupe homogène type formé d'une matrice bidimensionnelle de modules de type "i" correspondants (caractéristiques pour chaque matrice).



M: module

i: indice du module

FIGURE 2.1: GROUPE HOMOGENE "i" TYPE

Un groupe homogène peut avoir, dépendant du réseau d'interconnexion entre les modules, N dimensions logiques. Toutefois, les limites technologiques actuelles ne

permettent présentement (1989) pas plus que deux dimensions physiques (dessin des masques). HCDE s'applique à des modules comportant un nombre arbitraire de dimensions logiques, c'est la régularité physique (dessin des masques) et non logique (réseau d'interconnexion des modules) qui est considérée. Bien que la technologie actuelle soit limitée à deux dimensions, il n'est pas exclu qu'à long terme une technologie tridimensionnelle soit disponible, à laquelle on pourra appliquer HCDE. Nous n'excluons pas la possibilité d'appliquer HCDE à la dimension temporelle, mais n'avons pas étudié ce sujet, faute de temps.

HCDE s'applique quel que soit le nombre de dimensions, cependant, pour simplifier la description et faciliter la compréhension de la technique, nous décrivons d'abord comment elle s'applique à un groupe homogène unidimensionnel ou segment de modules (sections 2.2 à 2.6). La section 2.7 (expansion hiérarchique) présente l'utilisation de HCDE pour permettre la réalisation de circuits bidimensionnels et multidimensionnels. Enfin, la section 2.8 discute de quelques aspects complémentaires relatifs à l'application de HCDE: description d'une puce généralisée, modules composés, etc.

2.2-HCDE POUR UN MODULE HOMOGENE UNIDIMENSIONNEL

Cette section décrit HCDE pour un groupe homogène unidimensionnel. L'orientation des segments (groupes homogènes unidimensionnels) est sans importance, aussi, pour simplifier le texte, la technique est décrite pour un segment horizontal de modules (exemple, figure 2.2).



M: module

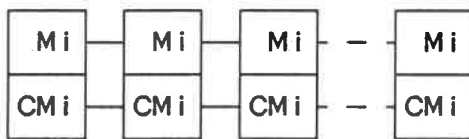
i: indice du module

FIGURE 2.2: GROUPE HOMOGENE "i" TYPE COMPORTANT n MODULES

La tolérance aux défauts est obtenue en ajoutant à chaque segment un nombre "r" (remplacement/rechange) déterminé de modules supplémentaires, identiques à ceux du segment original (pas de module spécial à concevoir), dont le rôle est de suppléer aux modules défectueux. Des multiplexeurs sont ajoutés aux entrées et aux sorties pour commander la reconfiguration du segment. Un groupe homogène peut alors tolérer r défauts à des modules différents et demeurer fonctionnel en tant que segment de n modules. Il n'y a pas de distinction entre les modules originaux et ceux de remplacement, nous avons un segment physiquement plus long ($n+r = n^*$ modules) de r modules correspondant au nombre de défauts à des modules différents pouvant être tolérés par

le segment. De l'extérieur du segment, le nombre de modules est le même que pour l'original (n modules). L'approche est transparente à l'utilisateur du segment: les bornes d'entrée/sortie du segment ne sont pas affectées par le circuit complémentaire (celui de HCDE).

HCDE prend avantage de la régularité d'un circuit en effectuant simultanément la vérification, le diagnostic et la reconfiguration de tous les modules d'un groupe homogène donné. Ceci est réalisé en ajoutant à chaque module un circuit complémentaire dont le rôle est d'effectuer localement ces opérations (vérification, diagnostic et reconfiguration) pour le module correspondant (figure 2.3).

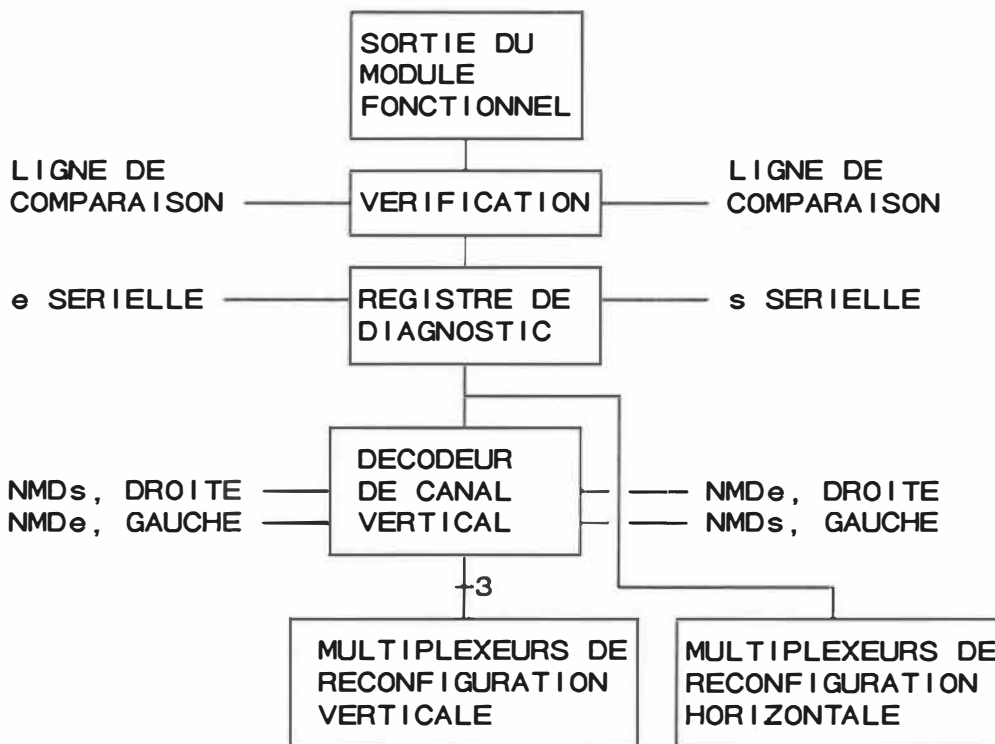


M: module
 i: indice du module
 CM: circuiterie complémentaire du M_i

FIGURE 2.3: MODULE HOMOGENE UNIDIMENSIONNEL "i" TYPE

La logique du circuit complémentaire est la même pour tous les modules, ce qui simplifie l'ajout de HCDE à des circuits déjà complexes. Signalons qu'une partie importante de ce circuit complémentaire sera intolérante aux défauts (kill area) alors que le segment peut tolérer r modules

défectueux. Conséquemment, un défaut dans la circuiterie complémentaire d'un module pourra invalider le segment entier plutôt que le module concerné seulement. Il est donc avantageux de minimiser la complexité du circuit intolérant aux défauts. La figure 2.4 montre un diagramme bloc du circuit complémentaire type d'un module de segment homogène.



NMD: nombre de modules défectueux à.
 suffixe e/s: entrée/sortie.

FIGURE 2.4: DIAGRAMME BLOC DU CIRCUIT COMPLEMENTAIRE D'UN MODULE DE SEGMENT

Certaines de ces cellules sont invariantes quelle que soit la valeur r (nombre de modules de rechange par segment): cellule de comparaison, registre de diagnostic, multiplexeurs de reconfiguration latérale (bypass). Seuls les circuits de reconfiguration verticale (décodeur de reconfiguration longitudinale et multiplexeurs de reconfiguration correspondants) varient selon la valeur de r (section 2.6).

La section 2.3 décrit l'aspect vérification, la section 2.4 décrit l'aspect diagnostic, la section 2.5 décrit la reconfiguration latérale, la section 2.6 décrit la reconfiguration verticale (décodage et multiplexage) d'un module typique. La section 2.7 complète la section 2.6 en décrivant certaines variantes lorsque le nombre de modules de rechange augmente.

2.3-DETECTION D'UN MODULE DEFECTUEUX

Cette section décrit le fonctionnement de la cellule de vérification complémentaire au module d'un segment. Chaque module est complété d'une cellule de vérification.

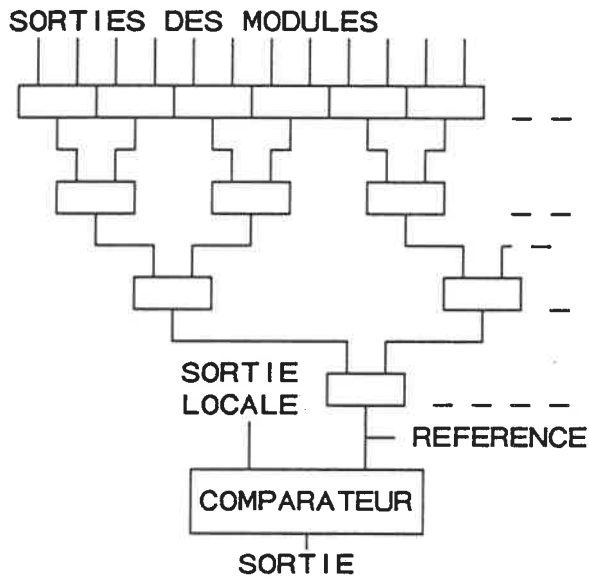
Puisque tous les modules d'un segment homogène sont identiques, il est possible d'en effectuer la vérification

en comparant leurs sorties entre elles si des vecteurs identiques sont injectés à leurs entrées. Il n'est pas du ressort de ce mémoire de décrire toutes les méthodes de génération de vecteurs de vérification; ce sujet a déjà été amplement couvert par d'autres. Citons simplement les approches exhaustives, pseudo-exhaustives [23] et algorithmiques comme PODEM [24], etc.

Dans HCDE, des vecteurs de vérification identiques sont injectés simultanément aux entrées de tous les modules, un circuit doit être prévu à cet effet. Les modules fonctionnels (sans défauts) auront des séquences de sortie identiques, alors que ceux qui sont défectueux auront des erreurs à leur sortie. Ces erreurs peuvent être détectées en comparant, à chaque cycle, les sorties des modules. Ceci évite d'avoir à générer artificiellement une séquence de vecteurs de signature, éliminant ainsi la nécessité de réaliser un circuit supplémentaire pour cette fonction. De plus, le problème de masquage ("aliasing") est complètement éliminé.

Un module défectueux peut alors être identifié en utilisant un comparateur à logique majoritaire, considérant comme valide la sortie de la majorité. Un signal de référence serait alors généré en compilant le nombre de

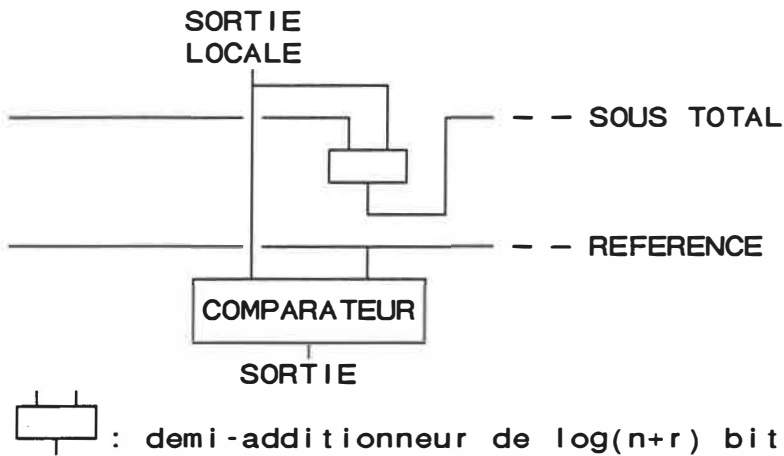
sorties à "1" des modules d'un segment et en comparant cette valeur au nombre de modules/2 ($(n+r)/2$) du segment homogène. La valeur de référence est "1" si le nombre de sorties "1" est supérieur à $(n+r)/2$ et "0" dans le cas contraire. La figure 2.5 montre un exemple d'arbre additionneur permettant de déterminer la valeur de référence.



 : additionneur de $\log(n+r)$ bits

FIGURE 2.5: DIAGRAMME BLOC D'UN COMPARATEUR A LOGIQUE MAJORITAIRE NUMERIQUE

"a"



 : demi-additionneur de $\log(n+r)$ bits

FIGURE 2.5b: DIAGRAMME BLOC D'UN COMPAREUR LOGIQUE "b"
REGULIER (circuit pour un module)

Il est évidemment inutile de déterminer plusieurs fois la valeur de référence, aussi le générateur de valeur de référence sera certainement distribué aux modules, manifestant alors une complexité de $\log(n+r)$ demi-additionneurs par module. Savaria [5] présente une manière d'implanter quasi régulièrement ce type de circuit. Le délai de propagation est proportionnel à $\log(n+r)$ puisque la réalisation est en arbre. Une structure absolument régulière est possible si l'on utilise plutôt une chaîne de retenue (figure 2.5a), cependant le délai de propagation devient alors rapidement excessif lorsque le nombre de modules $(n+r)$ augmente. Nous continuons donc avec la première solution (générateur de référence en arbre).

La complexité du circuit de chaque module devient de $\log(n+r)$ demi-additionneurs: un additionneurs de $\log(n+r)$ bits par module. La complexité d'un demi-additionneur est relativement élevée (1 ou exclusif et une porte ET: 14 transistors par bit d'additionneur en CMOS) pour des modules de complexité restreinte. La superficie requise lorsque le nombre de modules est élevé réduirait le gain de rendement dans le cas de modules de faible superficie. Lorsque la distance entre les modules sera élevée, le nombre de lignes séparant les modules ($\log(n+r)$) pourra occuper une superficie importante. De plus, le dessin des masques doit être ajusté à chaque augmentation du nombre de modules pour réaliser l'arbre. Dans les deux cas, le circuit n'est pas tolérant aux défauts.

Nous proposons une solution alternative originale qui utilise un circuit de comparaison majoritaire de type analogique (figure 2.6). Un tampon par module participe à la détermination du niveau logique d'une ligne de comparaison commune à tous les modules. Cette ligne de comparaison réalise élégamment et à peu de frais la détermination d'une sortie valide et peut ainsi servir directement de référence (sortie valide). Nous posons l'hypothèse que, pour chaque cycle de vérification des modules, une majorité de ceux-ci seront fonctionnels et auront donc des sorties identiques.

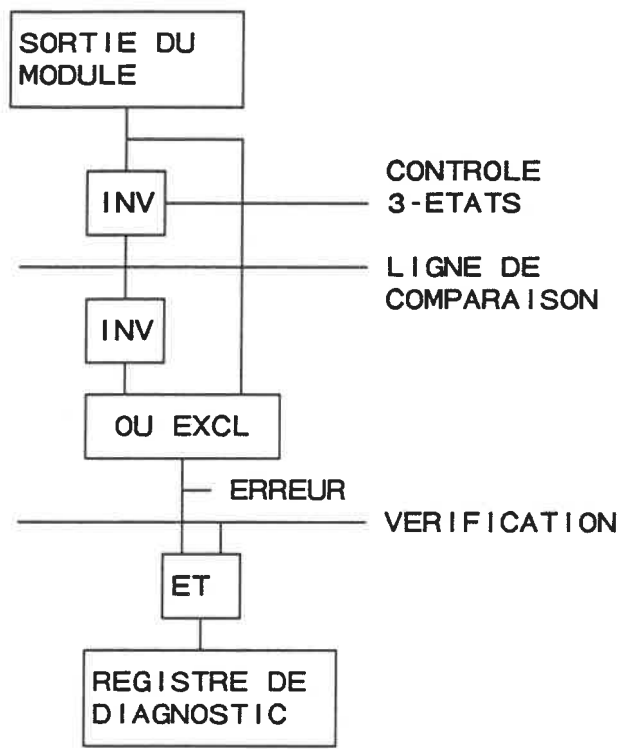
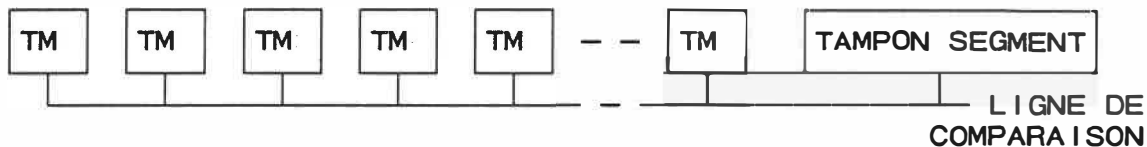


FIGURE 2.6: DIAGRAMME DE LA CELLULE DE VERIFICATION

Les défauts de modules apparaîtront sous forme d'erreurs, lorsque les vecteurs de vérification appropriés seront appliqués à leurs entrées. Lorsqu'il y a une erreur, les modules défectueux ont une sortie qui est en "conflit" avec celle de la majorité de modules valides à la ligne de comparaison. Ces modules défectueux n'auront pas suffisamment de force pour renverser la valeur logique (niveau analogique) de la ligne de comparaison (vote majoritaire fonctionnel) à une valeur fausse (figure 2.7). Ce sera le cas si le rendement des modules individuels est largement supérieur à 50%. Alors, la valeur de la ligne de

comparaison sera celle d'une sortie valide votée par la majorité de modules fonctionnels. Une faute peut alors être détectée par simple comparaison de la sortie de tous les modules d'un segment avec leurs pairs. Tous les modules sont comparés simultanément en considérant la valeur de la ligne de comparaison comme une sortie valide, un module dont la sortie est différente de l'homologue analogique (référence) sur la ligne de comparaison, se considère en faute.



TM: tampon module (3 états)

FIGURE 2.7: LIGNE DE COMPARAISON D'UN SEGMENT DE MODULES

Cette technique analogique, dont le coût d'implantation est linéaire avec le nombre de modules évite les problèmes d'irrégularité du câblage et de complexité résultant d'une solution logique purement numérique. Le circuit occupant une surface considérablement réduite (seulement 12 transistors en CMOS ou jusqu'à 5 portes élémentaires pour d'autres technologies) sera moins sujette aux défauts. De plus, le délai de propagation pour obtenir la valeur de référence sera nécessairement plus court et est constant quelle que soit la valeur de $(n+r)$. La technique

permet d'effectuer les tests à la vitesse maximale d'opération du circuit, ce qui serait clairement impossible si n est grand avec la technique purement logique. Le résultat local (pour un module) apparaissant au noeud "ERREUR", est disponible pour stockage dans le registre de diagnostic du module (section suivante).

La vérification des modules ne peut être considérée valide que si les cellules de vérification peuvent, elles aussi, être démontrées valides. Une difficulté est le grand nombre d'entrées qui commandent la ligne de comparaison. Nous avons considéré ce problème et la cellule de comparaison/vérification peut effectivement être vérifiée en désactivant (3ième état) les tampons de sortie des modules qui commandent la ligne de comparaison. Ceci permet de réduire la consommation pendant l'opération normale et permet aussi d'injecter un signal indépendant sur la ligne de comparaison via un autre tampon 3-états au niveau du segment (figure 2.7, ci-haut). Le second bit de vérification doit être forcé à travers le module. On peut ainsi injecter un vecteur exhaustif (2 entrées: 4 vecteurs requis) couvrant tous les états possibles des entrées des cellules de comparaison. On peut ensuite effectuer la lecture des résultats individuels des cellules de vérification en effectuant une lecture sérielle du registre de diagnostic

(section 2.4). Les cellules de vérification sont vérifiées depuis le niveau segment. Une vérification exhaustive des cellules de vérification des modules est ainsi possible au niveau hiérarchique supérieur suivant (segment de modules).

Un autre problème, lui-aussi dû au grand nombre d'entrées qui commandent la ligne de comparaison, est la vérification des tampons de sortie des modules à cette ligne. Il ne sera pas nécessaire de vérifier les tampons de sortie des PEs si le tampon de sortie de l'ensemble des PEs (du segment) est plus fort que le tampon de sortie bloqué à 1 d'un ou deux PEs. Il suffit d'ajouter la contrainte supplémentaire: chaque segment doit comporter suffisamment de modules fonctionnels pour que l'ensemble soit plus fort $r+2$ modules défectueux. r pour couvrir les modules défectueux, $+1$ pour couvrir un tampon de sortie défectueux et $+1$ pour que le signal de comparaison soit valide même s'il y a trois modules défectueux. Les tampons fonctionnels auront alors toujours suffisamment de force commune pour que la valeur de la ligne de comparaison soit valide. Cette contrainte assure un fonctionnement normal du système de vérification des PEs même si quelques tampons sont défectueux. Il n'est alors pas nécessaire de vérifier ces tampons. Le système de vérification peut ainsi être considéré vérifiable même si les tampons à la ligne de

comparaison ne sont pas vérifiés. La disposition des cellules de vérification leur confère une propriété de tolérance aux défauts, des défauts à ces cellules n'affecteront pas le segment mais seulement le module associé. Ces cellules ne font pas partie du circuit du segment. Pour résumer, dans cette cellule, seule la ligne de comparaison est intolérante aux défauts.

Une cellule de vérification et diagnostic par segment effectue un diagnostic des sorties des cellules de vérification via le canal de communication sériel des registres de diagnostic (un par module). Un diagnostic indépendant pour chaque module du segment doit être réalisé. La cellule compare tour-à-tour les sorties de chaque module avec la valeur valide prévue. Lorsqu'une erreur est détectée, un 1 est forcé à l'entrée sérielle du registre de diagnostic du segment, 0 dans le cas contraire. Lorsque les sorties de chaque module ont ainsi été lues et diagnostiquées, un chargement parallèle de l'étage de sortie (3ième étage) des registres de diagnostic (voir figure 2.8) des modules du segment est effectué (voir section 2.4). Cette vérification requiert $4 \times (1+n+r)$ cycles. Il n'est pas nécessaire de vérifier séparément les registres de diagnostic, ceux-ci étant automatiquement vérifiés par l'essai des vecteurs de vérification des cellules de

vérification.

Une séquence de vérification normale consiste à d'abord vérifier les registres de diagnostic et les cellules de vérification et ensuite les modules eux-mêmes.

2.4-REGISTRE DE DIAGNOSTIC

Le registre de diagnostic remplit les fonctions suivantes: 1-stocker le diagnostic du module associé, 2-permettre la vérification des cellules de comparaison, 3-injecter les résultats du diagnostic (niveau segment) de ces mêmes cellules de vérification et 4-lire et charger sériellement le registre de diagnostic pour permettre des remises en fonction rapide après un arrêt temporaire.

La figure 2.8 montre le diagramme bloc d'un registre de diagnostic typique pour HCDE. Les étages 1 et 2 peuvent être dynamiques, le troisième étage doit être statique. Les étages 1 et 3 doivent pouvoir être remis à 0.

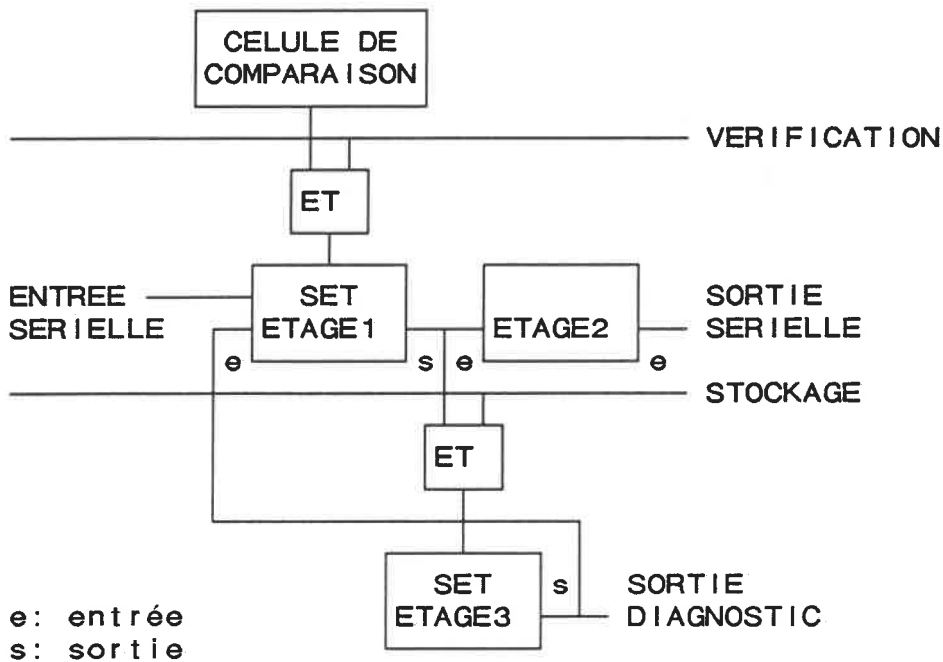


FIGURE 2.8: DIAGRAMME BLOC D'UN REGISTRE DE DIAGNOSTIC TYPIQUE

Le premier étage est l'étage crucial de ce registre; son état peut être modifié de quatre façons différentes: 1-mise-à-0, 2- chargement sériel de la donnée de sortie du second étage du registre du module précédent (chargement sériel), 3- mise-à-1 conditionnelle à un second signal de commande (VERIFICATION) et 4- transfert de la donnée du troisième étage au premier étage (pour lecture du diagnostic). Les deux premiers modes ne requièrent pas d'explications supplémentaires. Le troisième mode permet le chargement des résultats de la cellule de comparaison. La sortie de la cellule de comparaison passe par une porte ET qui sert à bloquer les transitoires inopportunes à la sortie

de la cellule de comparaison. Sur réception du signal de commande de vérification (chargement du registre), la sortie de la porte ET passe à 1 si la sortie de la cellule de comparaison est à 1. Ceci force une mise à 1 du premier étage du registre de diagnostic. Un chargement de donnée aurait effacé les diagnostics précédents. Il est indispensable que le chargement du résultat de la cellule de comparaison soit une mise-à-1, plutôt qu'un chargement de donnée, évitant ainsi la perte des erreurs précédemment diagnostiquées. Pour éviter que le premier étage n'augmente démesurément la consommation, le signal de commande mise-à-0 du premier étage du registre de diagnostic, doit être maintenu pendant les périodes prolongées d'opération en mode normal (après l'initialisation de la puce).

Le troisième étage stocke à long terme les diagnostics (module et cellules de vérification) du module local; sa sortie est le diagnostic du module. Cet étage, contrairement aux deux autres, doit donc être statique. Son état peut être modifié de deux façons différentes: 1-mise-à-0 et 2-mise-à-1 conditionnelle en réponse à un second signal de commande (STOCKAGE). Ce second mode est similaire au troisième mode du premier étage et est lui-aussi réalisé en utilisant une porte ET. La différence

31.

majeure est que cet étage est statique afin de permettre un stockage permanent du diagnostic. Le second étage est simplement un étage de propagation utilisé pour la lecture/écriture sérielle, d'un module au suivant, du registre de diagnostic (RD).

Une séquence typique de vérification des cellules de vérification procéderait comme suit: 1-remise à 0 du premier étage du RD (registre de diagnostic), 2-injection du vecteur de vérification à l'entrée de la cellule de comparaison, 3-chargement conditionnel du premier étage du RD par la ligne de commande "VERIFICATION", 4-transfert de la valeur du premier étage au second étage du RD, 5- n+r cycles de lecture-écriture sérielle du RD (une cellule au niveau segment programme à chaque cycle la valeur appropriée du RD), 6-chargement parallèle pour tous les modules des résultats (diagnostic) dans le troisième étage (pour stockage longue durée). Cette séquence (1 à 6) est répétée pour chacun des quatre vecteurs de vérification des cellules de vérification des modules requis. Les cellules de vérification (comparaison et canal de donnée au registre de diagnostic) sont alors vérifiées et diagnostiqués (du moins pour les fautes bloquées-à 1 ou 0).

La vérification et le diagnostic des modules sont beaucoup plus simples: 1-mise-à-0 du premier étage du RD, 2-injection du vecteur de vérification à l'entrée du module, 3-chargement conditionnel du premier étage du RD, 4-chargement conditionnel du troisième étage du RD. Les troisième et quatrième cycles sont répétés pour chaque vecteur de vérification et peuvent être effectués pendant la même phase d'horloge dans ce mode. Chaque vecteur de vérification ne requiert donc qu'un seul cycle. A ce point-ci, les modules et leur cellule de vérification ont été vérifiés et diagnostiqués. Un "1" à la sortie du troisième étage du registre de diagnostic indique que le module ou la cellule de vérification est défectueux et que le module local doit être exclu du segment.

A ce point, les modules défectueux (incluant leur cellule de vérification) sont identifiés. Le segment comporte n^* modules dont r peuvent être défectueux. Les branchements extérieurs au segment de M_s (modules) doivent être fixés pour que la reconfiguration soit transparente à l'utilisateur. Il n'est pas garanti que r modules soient défectueux. Dans HCDE, r modules par segment doivent être exclus pour que les signaux d'interconnexion entre les modules demeurent cohérents. S'il y a moins que r modules défectueux, certains modules seront inutilisés; nous n'avons

pas développé de circuiterie pour différencier un module défectueux d'un module inutilisé. En conséquence, un module inutilisé doit être déclaré défectueux (pseudo-défectueux) en forçant son registre de diagnostic à "1". La différence entre un module pseudo-défectueux et un module physiquement défectueux est que le pseudo-défectueux est sans défaut et qu'il pourra, plus tard, remplacer un module alors fraîchement devenu défectueux et détecté lors d'une vérification (ex: remise en marche, etc.) du système.

Dans HCDE, le choix des modules inutilisés n'a pas d'importance. Nous avons donc voulu favoriser une solution simple utilisant le mode sériel du registre de diagnostic et une paire de compteurs. Le premier compteur compte le nombre de modules et le second compte le nombre de modules fonctionnels (non-défectueux). Ce second compteur est d'abord chargé de la valeur n et décrémente de 1 à chaque fois que la sortie du RD est 0 jusqu'à ce qu'il atteigne l'état 0, où il demeure alors bloqué (à 0). Lorsque le second compteur est 0, un 1 est chargé à l'entrée du RD, quelle que soit la valeur de la sortie du RD. Tous les modules subséquents à l'état 0 du second compteur seront donc considérés pseudo-défectueux, qu'ils le soient réellement ou pas. Si lorsque le premier compteur arrive à 0, le second n'est pas à 0, il y a alors plus que r modules

34.

défectueux et le segment doit être rejeté. Lorsque le premier compteur est 0, la programmation des modules inutilisés est complète.

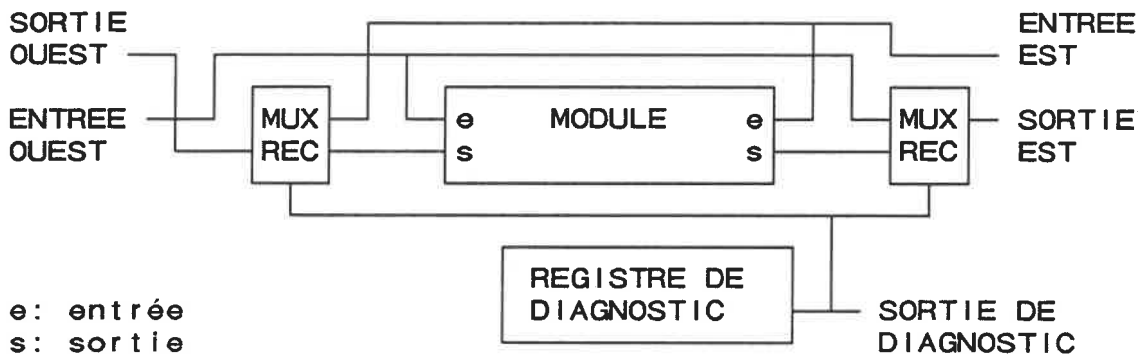
La séquence est la suivante: 1-chargement de n dans le compteur de modules fonctionnels, 2-transfert des valeurs des troisièmes étages des RDs aux premiers étages (sur Phi1) et propagation aux seconds (sur Phi2), 3- n+r cycles de lecture et écriture sérielle du RD, module par module. Deux comptages simultanés (nombre de modules lus et nombre de modules fonctionnels détectés) sont effectués, le registre sériel est rechargé avec une valeur diagnostique correspondant à celle de la table 2.1. 4-Le troisième étage est finalement chargé conditionnellement pour stocker à long terme la condition inutilisé = pseudo-défectueux = défectueux.

TABLE 2.1: TABLE DE VERITE
POUR DETERMINER LES MODULES INUTILISES

COMPTEUR DE MODULES FONCTIONNELS		REGISTRE DE DIAGNOSTIC	
COMPTE DESCENDANT	DECREMENT	SORTIE	ENTREE
n à 1	non	1	1
n à 1	oui	0	0
0	non	X	1

2.5-RECONFIGURATION LATÉRALE

A ce point, les modules et leur circuiterie de vérification et de diagnostic ont été vérifiés, les modules défectueux et inutilisés ont été identifiés (registre de diagnostic). La circuiterie de reconfiguration a pour rôle de choisir et de brancher les modules opérationnels. Ceci se fait en débranchant ou contournant ceux qui sont déclarés défectueux. Horizontalement, on évite (bypass) les modules défectueux. Il n'est pas nécessaire de modifier le branchement aux entrées latérales car le module est défectueux de toute façon; il suffit donc de remplacer la sortie invalide par celle du module valide (sortie du voisin précédent). Chaque signal de sortie est dévié au moyen d'un multiplexeur à deux entrées commandé par la sortie du troisième étage (statique) du registre de diagnostic (figure 2.9), ceci quel que soit le nombre de modules (n) ou le nombre de modules de rechange (r) par segment. Si le nombre de sorties horizontales le justifie, des économies seront réalisées en remplaçant les multiplexeurs par des paires d'interrupteurs partageant les signaux de commande (un signal module normal et un signal module défectueux), un seul inverseur est requis pour tous les multiplexeurs. Suivant cette méthode, le système requiert un inverseur et deux interrupteurs par sortie horizontale.



MUX REC: multiplexeur de reconfiguration horizontale

FIGURE 2.9: EXEMPLE DE RECONFIGURATION HORIZONTALE

La tolérance aux défauts de ces multiplexeurs est relative à l'état du RD du module. Il est nécessaire de cerner les types de défauts pour identifier le degré de tolérance. Les chemins utilisés sont intolérants aux défauts. Les chemins inutilisés doivent être en circuit ouvert. Conséquemment, certains types de défauts aux interrupteurs ouverts: canal ouvert, contact de canal absent, ligne reliant les polarités de canal ouvertes, lignes d'accès aux signaux ouvertes; sont sans effet. Les lignes de commande de l'interrupteur, incluant les grilles de commande, doivent être valides. Nous concluons que les interrupteurs ouverts sont partiellement tolérants aux défauts, tandis que les interrupteurs fermés sont entièrement intolérants. Conséquemment, 50% du circuit est partiellement tolérant aux défauts de type ligne ouverte.

La vérification des interrupteurs de reconfiguration des modules fait partie du segment. Une défectuosité à ce niveau invalide donc le segment plutôt que les modules seulement.

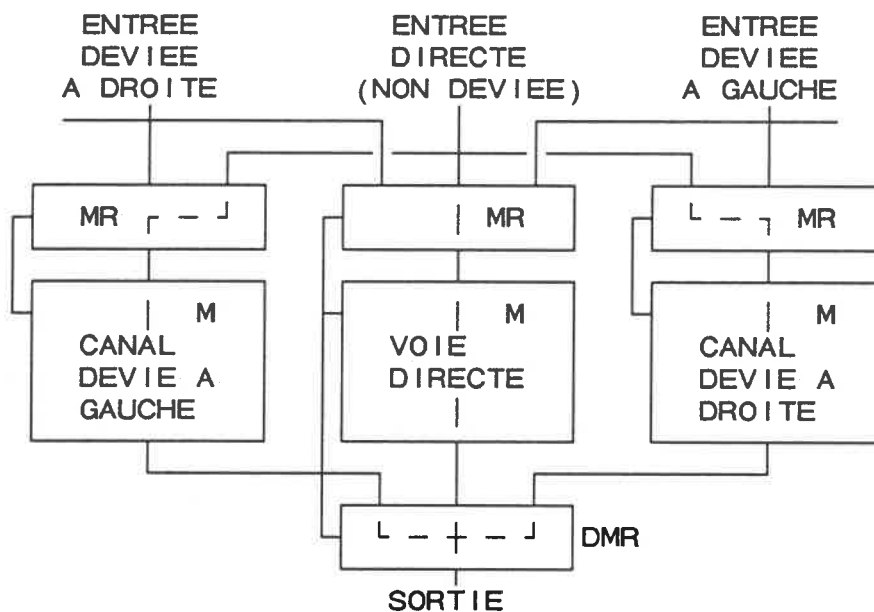
2.6-RECONFIGURATION VERTICALE

2.6a- INTRODUCTION ET MULTIPLEXEURS

Tous les canaux non-horizontaux sont considérés verticaux: nord, sud, nord-est, nord-ouest, sud-est, sud-ouest, tout canal qui n'est pas est ou ouest. Leur reconfiguration est un peu plus complexe que celle des canaux horizontaux parce qu'elle est affectée par la valeur de r (nombre de modules de rechange). Le circuit de reconfiguration verticale comporte deux sections principales: un bloc décisionnel logique déterminant les branchements et un bloc formé de multiplexeurs effectuant la déviation des signaux. Les chemins de données verticaux sont reconfigurés en déviant les entrées et sorties, vers la gauche ou la droite, suivant le nombre de modules défectueux dans chaque direction (exemple: figure 2.10).

L'utilisation de décodeurs identiques dans les modules d'un même segment assure la cohérence du circuit de reconfiguration qui est transparent à l'utilisateur. Pour

l'utilisateur du segment, les entrées et les sorties sont fixées, le circuit de reconfiguration garantit que s'il n'y a pas plus que r défauts dans le circuit tolérant et aucun défaut dans le circuit intolérant, la fonctionnalité du circuit est conservée.



M: module

MR: multiplexeur de reconfiguration verticale (entrée de module)

DMR: démultiplexeur de reconfiguration verticale (sortie de module)

FIGURE 2.10: EXEMPLE DE DEVIATION DE CANAL DE DONNEES VERTICAL

(cas deux modules de remplacement: $r=2$)

Une difficulté, avec le circuit de reconfiguration verticale, est qu'il dépend de la valeur de r (nombre de modules de remplacement). Considérons d'abord les conséquences sur la complexité du bloc multiplexeur.

Ce bloc comprend autant de multiplexeurs qu'il y a d'entrées et autant de démultiplexeurs qu'il y a de sorties. Le nombre d'entrées des multiplexeurs et le nombre de sorties des démultiplexeurs sont les mêmes: $r+1$. Ainsi un module peut occuper sa position normale et n'importe laquelle de r positions suppléantes. Chacune de ces positions correspond à des positions qui auraient été occupés par des modules si HCDE n'avait pas été utilisée. Dévier un canal de données de une position vers la gauche est équivalent, pour un module, à utiliser les entrées/sorties du module voisin de droite et vice versa. Le même raisonnement s'applique pour des déviations de distance plus grande que 1, il suffit d'appliquer un facteur de distance identique à chacune des entrées d'un module; fonctionnellement, un module remplace un module voisin situé à une distance définie. en ce qui concerne le bloc multiplexeurs/démultiplexeurs, la valeur de r n'affecte que le nombre d'entrées par mux/démux: $r+1$ entrées/sorties.

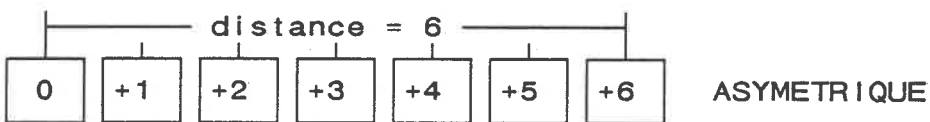
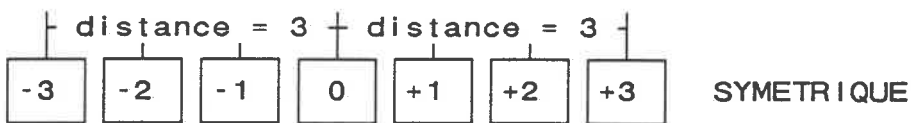
Nous avons envisagé plusieurs approches possibles quant-à la localisation relative des positions de suppléance. Nous en avons retenu trois: 1-déviations asymétrique, 2-déviations symétrique et 3-approche mixte de déviations. Nous proposons l'approche mixte comme solution valide quelle que soit la valeur de r . L'approche symétrique

est intéressante dans le cas $r=2$ seulement. Voyons d'abord les deux approches élémentaires, pour présenter ensuite l'approche mixte. L'approche asymétrique procède en ne déviant que selon une seule direction (gauche ou droite). L'approche symétrique procède en déviant la moitié de la distance mais suivant deux directions possibles (gauche et droite).

L'approche asymétrique a l'avantage de minimiser la complexité de la description du nombre de modules défectueux d'un segment lorsque la valeur de r est élevée. En effet, le nombre maximal de modules défectueux est de r . Avec l'approche asymétrique, un seul vecteur (une seule direction) de $\log(r)$ bits est requis pour indiquer le nombre de modules défectueux. Avec l'approche symétrique, deux vecteurs (deux directions) de $\log(r)$ bits sont requis pour indiquer le nombre de modules défectueux dans chaque direction, pour un total de $2 \log(r)$ bits. Le nombre de lignes requis par l'approche symétrique est supérieur lorsque r est plus grand que 2 (table 2.2). L'approche symétrique a cependant l'avantage de réduire la distance de déviation maximale de moitié, restreignant ainsi les délais de propagation encourus (figure 2.11).

TABLE 2.2: COMPARAISON DES APPROCHES SYMETRIQUES ET ASYMETRIQUES (complexité du vecteur de défautuosité, r variable)

r	asymétrique	symétrique
1	1	
2	2	2
3	2	
4	3	4
5	3	
6	3	4
7	3	
8	3	6



0 indique le module considéré
 x indique la distance de déplacement

FIGURE 2.11: COMPARAISON DES APPROCHES SYMETRIQUE ET ASYMETRIQUE (distances de déviation maximales, cas $r=6$)

Il serait intéressant de profiter des avantages combinés des deux approches. Ceci est possible en utilisant l'approche asymétrique, mais en décalant les branchements d'entrée aux modules de $r/2$ positions (figure 2.12). Cette approche mixte combine les avantages de l'approche asymétrique (vecteur de description de complexité $\log(r)$) tout en coupant les distances de déviation en deux (avantage

de l'approche symétrique). La superficie d'interconnexion est pratiquement la même dans chaque cas (asymétrique, symétrique, mixte).

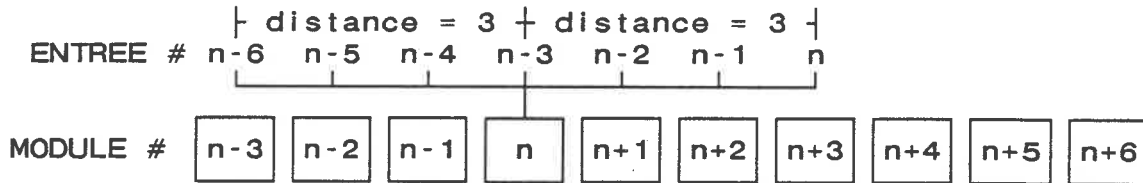


FIGURE 2.12: EXEMPLE DE L'APPROCHES MIXTE (cas: $r=6$)

Seuls des modules de faible rendement justifieront l'utilisation de valeurs élevées de r . Vu les tendances actuelles d'évolution de la qualité des procédés, des modules de faible rendement sont nécessairement des modules de superficie importante. Par conséquent, bien que la complexité du décodeur décisionnel augmente avec r , la superficie de ces décodeurs demeurera restreinte relativement à celle du circuit fonctionnel. Pour r plus petit que 3, les trois approches sont valables. Dans la pratique, l'approche symétrique n'est donc intéressante que dans le cas $r=2$ (le nombre de lignes descriptives des modules défectueux, est identique pour les trois approches lorsque $r=2$). Nous limitons donc l'approche symétrique au cas $r=2$ (table 2.2). Pour des raisons de simplicité, l'approche mixte est suffisante pour présenter une solution valide quelle que soit la valeur de r .

Comme dans le cas des interrupteurs de reconfiguration latérale, seuls les chemins utilisés sont intolérants aux défauts. Les chemins inutilisés doivent être en circuit ouvert. Conséquemment, certains types de défauts aux interrupteurs ouverts: canal ouvert, contact de canal absent, ligne reliant les polarités de canal ouvertes, lignes d'accès aux signaux ouvertes; sont sans effet. Les lignes de commande de l'interrupteur, incluant les grilles de commande, doivent être valides. Nous concluons que les interrupteurs ouverts sont partiellement tolérants aux défauts, tandis que les interrupteurs fermés sont entièrement intolérants. Conséquemment, une fonction $(r-1)/r$ du circuit est tolérant à certains défauts de type ligne ouverte.

La vérification des interrupteurs de reconfiguration des modules fait partie du segment. Leur défectuosité invalide donc le segment plutôt que les modules seulement.

2.6b- DECODEUR DECISIONNEL, APPROCHE SYMETRIQUE, (cas $r=2$)

L'approche symétrique n'étant avantageuse que lorsque $r=2$ (2 modules de remplacement), nous considérons ce cas particulier seulement. Le chapitre 5 décrit un exemple

d'implantation de cette approche (symétrique, $r=2$). Chaque module peut occuper trois positions différentes (voir figure 2.11): ne pas dévier le canal de communication, dévier le canal vers la droite, ou le dévier vers la gauche. Le décodeur confronte la sortie du registre de diagnostic local avec deux signaux additionnels provenant des décodeurs des modules voisins immédiats (droite et gauche): "au moins un module défectueux à droite" et "au moins un module défectueux à gauche". Le décodeur produit 5 sorties: dévier le chemin vertical vers la droite, dévier le chemin vertical vers la gauche, ne pas dévier le chemin vertical (table 2.3) et les sorties module défectueux à droite et module défectueux à gauche requises pour propagation aux modules voisins. Les trois premières sorties sont des signaux mutuellement exclusifs qui commandent des triplets d'interrupteurs servant de multiplexeurs de reconfiguration verticale. Les deux derniers signaux prolongent une chaîne de propagation ("daisy chain") déterminant la présence ou l'absence de modules défectueux dans chaque direction. Ceci est facilement réalisé en effectuant un OU de l'entrée correspondante (droite ou gauche) avec la valeur de sortie du registre de diagnostic local.

Ici encore, comme dans le cas de la reconfiguration horizontale, des économies sont réalisables en utilisant une

seule cellule de décodage pour tous les multiplexeurs/démultiplexeurs d'un module; par conséquent, dans le cas de circuit CMOS, les multiplexeurs à 3 entrées sont des triplets d'interrupteurs partageant un décodeur commun.

TABLE 2.3: TABLE DE VERITE DU DECODEUR DECISIONNEL (cas $r=2$)

AU MOINS UN MODULE DEFECTUEUX A		SORTIE DU REGISTRE DE DIAGNOSTIC LOCAL	
DROITE	GAUCHE	FONCTIONNEL	DEFECTUEUX
non	non	sans importance	sans importance
non	oui	dévier à droite	dévier à droite
oui	oui	ne pas dévier	segment defectueux
oui	non	devier à gauche	dévier à gauche

L'état "sans importance" doit être défini de manière identique pour tous les modules; que ce soit "dédié à droite", "dédié à gauche" ou "non dévié". L'économie de transistors est une justification suffisante pour déterminer un choix. Dans le cas d'IMAGE2 (section 4.4), nous avons utilisé: sans importance = dévier à gauche. La figure 2.13 illustre quelques exemples de conditions de déviation de canaux de communication en utilisant cette approche (cas IMAGE2, $r=2$, $n=8$). Lorsque les modules sont minces, les

délais encourus par HCDE se réduisent aux délais à travers les interrupteurs.

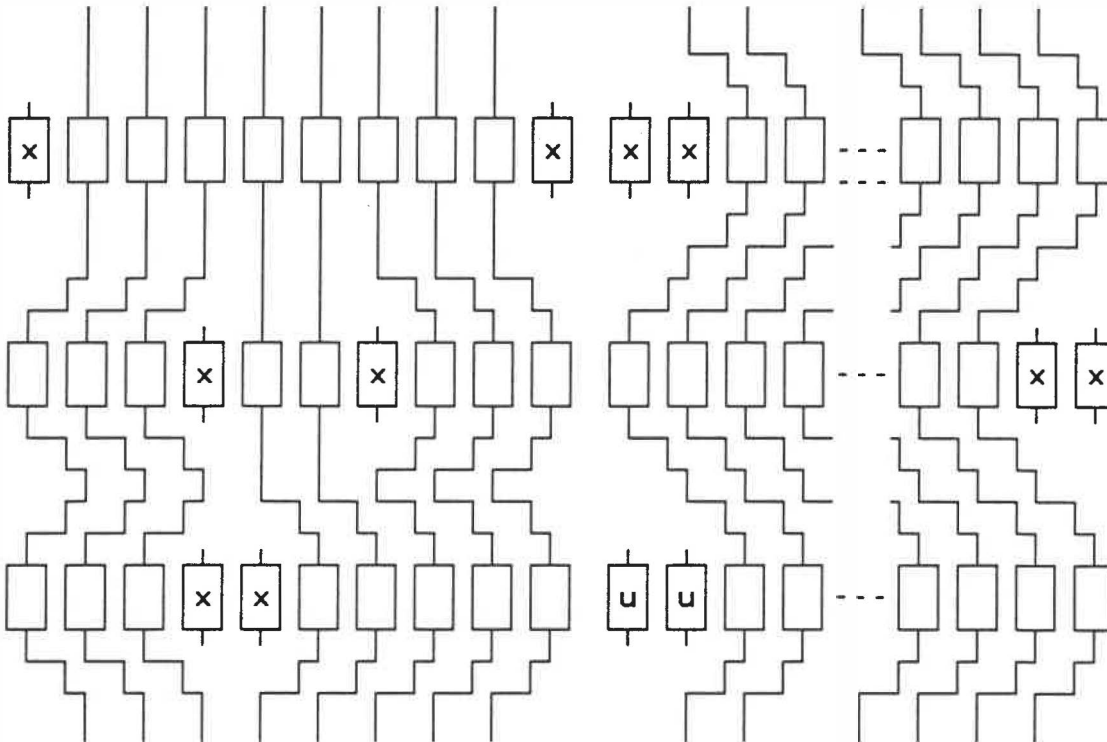


FIGURE 2.13: EXEMPLES DE DEVIATION DE CANAUX VERTICAUX
(cas $r=2$, $n=8$)

2.6c- DECODEUR DECISIONNEL, APPROCHES ASYMETRIQUE ET MIXTE

Considérons maintenant le décodeur de reconfiguration verticale pour r arbitraire et décodeur mixte (asymétrique). Un vecteur est requis pour indiquer le nombre de modules défectueux. Plus particulièrement, définissons ce vecteur "mr" (modules de rechange disponibles) comme indiquant le

nombre de modules de rechange restant. Cette dernière définition a l'avantage de réduire les différences entre les décodeurs lorsque r varie. En effet, la condition "aucun module de rechange disponible" est alors la même (0) quelle que soit la valeur de r ; seul le nombre de bits du vecteur m_r est différent.

Une première fonction du décodeur est alors de soustraire 1 au vecteur m_r d'entrée lorsque le module local est défectueux. Une seconde fonction est de déterminer le nombre de positions de déplacement (déviations). En ceci, l'approche asymétrique a l'avantage que m_r peut aussi être défini comme déterminant le nombre de positions de déplacement. En effet, chaque fois qu'un module est défectueux, il est horizontalement exclu du segment et la valeur de m_r baisse de 1; il devient alors nécessaire de varier de 1 le déplacement des connexions verticales et vu que m_r a été modifié par le décodeur du module défectueux, il n'est pas nécessaire de recalculer cette modification d'état.

La direction de l'asymétrie n'a pas d'importance; considérons le cas où les modules de rechange sont à gauche. La figure 2.14 montre un décodeur asymétrique/mixte, la table 2.4 montre une table de vérité généralisée pour la

cellule de comptage; la figure 2.15 montre l'évolution de la valeur du vecteur mr après que les modules inutilisés aient été marqués défectueux.

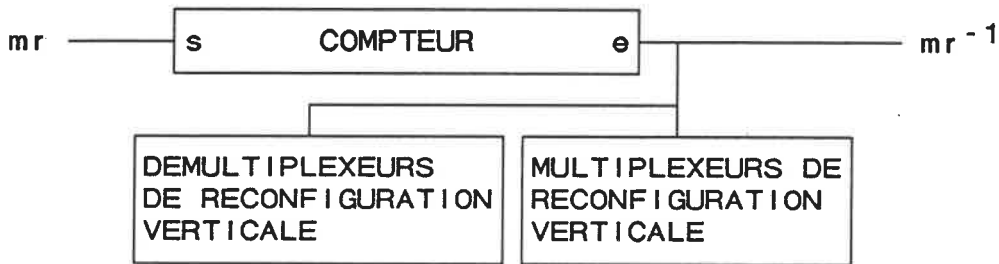


FIGURE 2.14: DIAGRAMME BLOC DU DECODEUR DECISIONNEL ASYMETRIQUE/MIXTE

TABLE 2.4: TABLE DE VERITE DU DECODEUR ASYMETRIQUE ($r=6$)

mr d'entrée	module défectueux	mr de sortie	déviat. locale	non-reconfigurable
r	oui	r-1	r	0
r	non	r	r	0
j	oui	j-1	j	0
j	non	j	j	0
0	oui	X	0	1
0	non	0	0	0

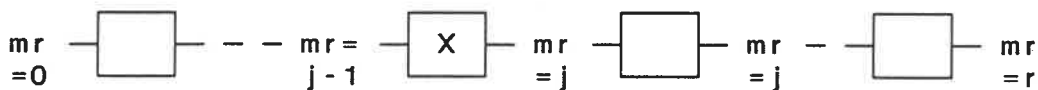


FIGURE 2.15: EVOLUTION DE mr DANS LA CHAÎNE DE DECODEURS DECISIONNELS

Les compteurs des modules peuvent être remplacés par des registres sériels et un additionneur sériel commun aux modules du segment soustrayant 1 pour chaque module. Cette

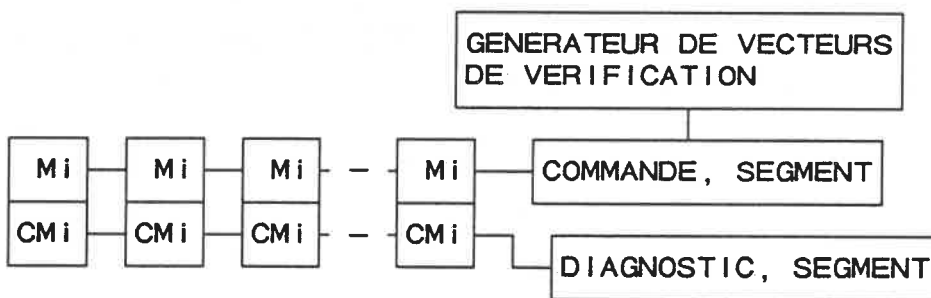
approche réduit la superficie intolérante aux défauts des segments et est ainsi supérieure du point de vue rendement. Le calcul des déplacements n'a à être effectué qu'une seule fois: après que les modules ont été diagnostiqués. Ceci ne coûte que $k \cdot (n+r)$ cycles ou k est le nombre de bits du vecteur m_r et $(n+r)$ est le nombre de modules (fonctionnels + remplacement) par segment. L'approche sérielle sera donc avantageuse pour des valeurs élevées de $n+r$. Elle sera encore plus avantageuse lorsque r sera élevé car elle permet de réduire le nombre de lignes de $\log(r)$ à une seule. Dans ce cas, des registres statiques doivent être prévus pour stocker m_r localement.

2.7-EXPANSION HIERARCHIQUE

Maintenant que nous savons comment appliquer CDE (détection et exclusion de conflit) à un groupe homogène unidimensionnel (segment), voyons comment reconfigurer des groupes bidimensionnels.

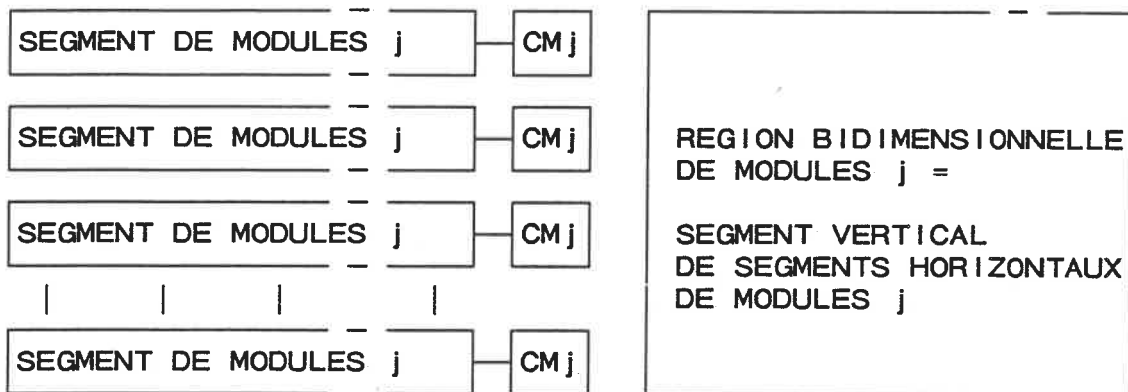
Nous appliquons CDE à chaque segment de modules tel qu'expliqué dans les sections 2.2 à 2.6. Puis nous considérons ces segments reconfigurés comme des modules au niveau hiérarchique supérieur; ceci inclut le circuit complémentaire du premier niveau, le circuit de commande du

segment, ainsi que le circuit complémentaire CDE (second niveau, figure 2.16). Un segment de modules de second niveau est constitué d'une région de modules du premier niveau si les segments sont juxtaposés perpendiculairement (verticalement dans notre exemple) à l'orientation de ceux du premier niveau hiérarchique. CDE s'applique à ce niveau aussi car ce segment vertical est homogène et le rendement des modules du second niveau est élevé (amélioré par CDE appliqué au premier niveau). Nous avons alors (figure 2.17) un segment vertical de modules-segments ou une région bidimensionnelle de modules élémentaires.



M: module
 i: indice du module
 CM: circuit complémentaire du Mi

FIGURE 2.16: DIAGRAMME BLOC D'UN SEGMENT AVEC CDE



CM_j: commande du segment horizontal de modules j

FIGURE 2.17: formation d'une région (segment de segments de modules)

Tout segment de modules auquel on applique la technique de détection et exclusion de conflits décrite aux sections 2.2 à 2.6 est un niveau hiérarchique CDE. Le plus petit module non reconfigurable mais servant de module de base d'un segment de modules est aussi un niveau hiérarchique CDE. Une hiérarchie HCDE est l'ensemble des niveaux hiérarchiques CDE d'un groupe homogène auquel s'ajoute le plus petit module non-reconfigurable répété (le module élémentaire du premier niveau hiérarchique). La figure 2.18 montre un exemple de groupe homogène bidimensionnel auquel HCDE a été appliqué de manière à former trois niveaux hiérarchiques: 1-modules non-reconfigurables, 2-segment horizontal de modules et 3-segment vertical de segments horizontaux de modules et l'ensemble des circuits irréguliers complémentaires au

groupe.

Tous les niveaux hiérarchiques CDE, excepté le plus bas, comprennent un segment homogène de modules et un module irrégulier complémentaire (circuit de diagnostic des cellules de vérification et registres de diagnostic des segments, circuit complémentaire du segment, circuit de commande ou autre commune à tous les modules, etc). Le module du niveau hiérarchique CDE inférieur comprend la partie fonctionnelle du module plus son circuit complémentaire (figure 2.18).

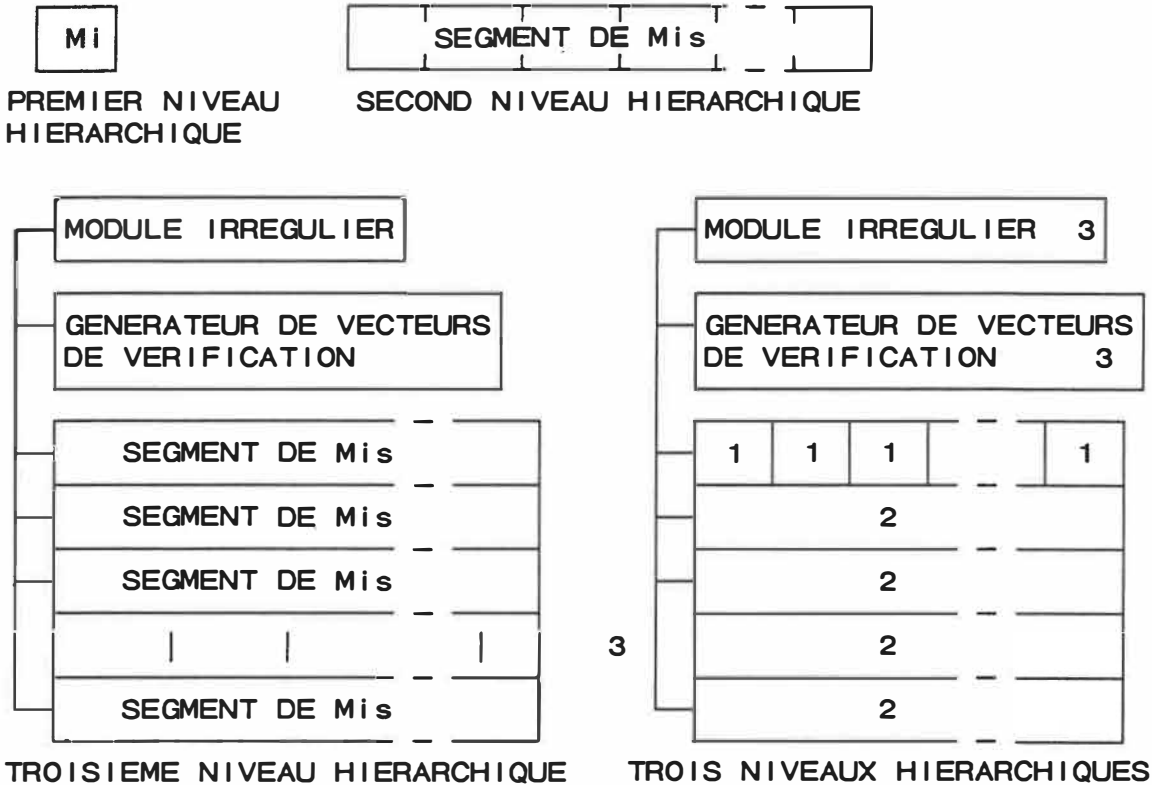


FIGURE 2.18: EXEMPLE DE CIRCUIT AVEC 3 NIVEAUX HIERARCHIQUES CDE

Le circuit complémentaire (vérification, diagnostic et reconfiguration) décrit aux sections 2.2 à 2.6 pour le premier niveau hiérarchique s'applique aussi aux niveaux supérieurs. Une vérification et reconfiguration à l'échelle du système pour un circuit comportant trois niveaux hiérarchiques procéderait comme suit.

On vérifie d'abord le premier niveau hiérarchique. Ceci se fait en vérifiant les circuits de vérification et de

diagnostic locaux des modules via la chaîne de registres de diagnostic du segment tel que décrit aux sections 2.3 et 2.4. Tous les modules (premier niveau hiérarchique) du groupe (segment de segments) sont vérifiés simultanément tel que décrit à la section 2.3. Insistons sur le fait que tous les modules de tous les segments sont vérifiés en même temps. Le nombre de vecteurs requis dépend donc de la complexité du module élémentaire plutôt que de celle du groupe, il en résulte des économies de temps de vérification proportionnelles au nombre de modules total. Les modules inutilisés sont ensuite identifiés tel que décrit à la section 2.4. Les segments sont alors automatiquement reconfigurés. A ce point tous les modules d'un même segment et identifiés fonctionnels sont démontrés identiques. Cependant la validité des circuits de commande des segments demeure à être confirmée par la vérification du second niveau hiérarchique: segment de modules.

La vérification du second niveau hiérarchique vise à démontrer la validité des interconnexions entre les modules des segments, du circuit complémentaire des segments (CDE) et de tout circuit de commande partagé par les modules (premier niveau hiérarchique) d'un segment. Les segments étant identiques, CDE s'applique de nouveau à ce niveau hiérarchique. On vérifie d'abord les cellules de diagnostic

et de vérification des segments de modules. On vérifie ensuite le circuit de commande. Ceci débute par l'injection des vecteurs de vérification et l'échantillonnage de la sortie d'un des modules (idéalement le plus éloigné du circuit irrégulier, pour valider aussi les lignes de commande). On vérifie alors les interconnexions entre les modules des segments. On identifie enfin les segments inutilisés. La reconfiguration du second niveau est elle aussi automatique et la région de modules est alors complète. Les segments de modules déclarés fonctionnels sont ainsi démontrés identiques. Au troisième niveau hiérarchique on vérifie les interconnexions entre les segments des régions et le circuit irrégulier du groupe (dans le cas d'un groupe à trois niveaux).

La vérification procède ainsi du niveau hiérarchique le plus bas vers les niveaux supérieurs, un niveau à la fois, jusqu'à ce que tous les niveaux, sauf le dernier, soient vérifiées et reconfigurés. Le niveau supérieur n'étant jamais régulier, il ne peut être reconfiguré par HCDE et des techniques complémentaires pourront être appliquées pour obtenir des résultats supérieurs (du point de vue du rendement) à ce niveau aussi. Le concepteur voudra réduire au minimum la complexité du circuit irrégulier (commande, canaux de données) du niveau hiérarchique

supérieur. La conception du dernier niveau hiérarchique aura les conséquences les plus directes sur le rendement final d'un circuit.

Ainsi, la simplification de la complexité de la vérification et de la reconfiguration, par utilisation de circuits distribués, obtenue au niveau hiérarchique inférieur (niveau module) est conservée aux niveaux supérieurs subséquents. L'usage de l'approche HCDE rend l'auto-testabilité et l'auto-reconfigurabilité applicable à tout niveau de complexité de circuit intégré (ou autre) envisageable. Il suffit de maintenir l'orthogonalité entre les divers niveaux hiérarchiques si on désire une reconfigurabilité multidimensionnelle.

L'exemple ci-haut couvre trois niveaux hiérarchiques; seule une spécification de rendement définie limite l'application de HCDE à d'autres niveaux hiérarchiques.

Dans la mesure où le rendement d'un niveau hiérarchique donné peut être contrôlé (maintenu élevé), on peut augmenter la superficie d'un circuit en appliquant un niveau hiérarchique CDE supplémentaire. L'augmentation de la superficie n'est plus freinée que par les limites physique des circuits monolithiques (tranche, voir aussi section

2.10). En subdivisant le problème de la reconfiguration multidimensionnelle en plusieurs problèmes unidimensionnels, nous obtenons une technique structurée applicable à un nombre arbitraire de dimensions et pouvant déborder les frontières des boîtiers, des cartes, des cabinets...

2.8-APPLICATION

2.8a-CRITERES D'APPLICABILITE DE HCDE

HCDE s'applique à tout circuit rencontrant les critères suivants:

1- le circuit doit être régulier.

2- le cellule élémentaire ou module doit être vérifiable.

3- le module doit avoir un rendement suffisant. Bien qu'un rendement minimal de 50% soit spécifié, des rendements supérieurs à 80% produiront des résultats supérieurs (section 2.3).

Le circuit complémentaire d'un module est presque entièrement vérifiable pour les fautes bloqué à 1 et 0 et les transitions. Le canal de données sériel du registre de diagnostic est couvert à 100%. Le troisième étage du

registre de diagnostic est couvert à 100% des états (0,1) et transitions (0 à 1 et 1 à 0) possibles. L'entrée conditionnelle et de remise à 0 des premiers et troisième étages du registre de diagnostic permettent de vérifier chaque transition de chaque étage du registre de diagnostic. Chaque transition de chaque entrée de la cellule de comparaison et la ligne de comparaison sont vérifiables à 100%. Le tampon de sortie du PE à la ligne de comparaison est tolérant aux défauts. Le multiplexeur de vérification est entièrement vérifiable pour chaque transition de chaque noeud pour chaque entrée affectant ce noeud.

Un interrupteur de reconfiguration inutilisé peut être ouvert sans que le fonctionnement du circuit soit affecté. Un court-circuit avec une ligne fonctionnelle peut cependant affecter le fonctionnement du circuit. Nous n'avons pas trouvé de méthode pour la vérification de ce dernier mode de défaut.

2.8b-MODULES COMPOSES

Si les modules sont trop petits pour justifier l'utilisation de HCDE sur chaque module, quelques modules peuvent être groupés, formant alors des modules-composés plus importants. La géométrie de ce regroupement est à la

discrétion du concepteur qui pourra tenir compte de la superficie, de la géométrie et du nombre de points d'entrée/sortie aux modules-composés ainsi formés (figure 2.19) selon les caractéristiques du problème particulier. Ceci permet d'optimiser la superficie relative du circuit complémentaire par rapport à celle des modules, maximisant ainsi l'avantage du point de vue rendement ou fiabilité. Cette décision affectera aussi possiblement le nombre de niveaux hiérarchiques requis.

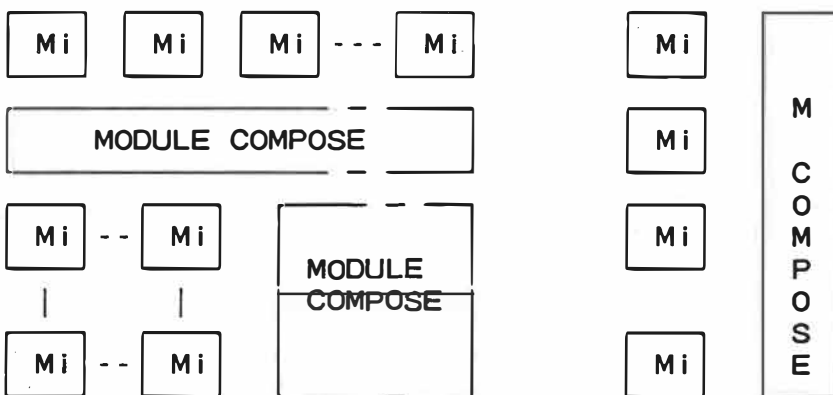


FIGURE 2.19: EXEMPLES DE MODULES-COMPOSES

2.8c-HCDE POUR UN CIRCUIT INTEGRE GENERALISE

Considérons un circuit intégré généralisé comportant des parties irrégulières et homogènes. L'ensemble des groupes homogènes et irréguliers constituent un niveau hiérarchique hétérogène. La figure 2.20 montre une représentation d'un circuit hétérogène représentatif de la

majorité des circuits existants ou réalisables (mémoire, microprocesseur, co-processeur, etc). On note la présence de plusieurs groupes homogènes distincts et de modules irréguliers.



FIGURE 2.20: PUCE TYPIQUE, ASPECT HETEROGENE

Du point de vue de HCDE, tous les modules irréguliers peuvent être groupés en un seul module commun (figure 2.21). Tous les autres modules seront donc au moins homogènes et possiblement réguliers. Les groupes de bornes d'entrée/sortie de même type sont homogènes. HCDE est applicable à tous les groupes homogènes ainsi formés.



* H: HOMOGENE

FIGURE 2.21: SEPARATION DES PARTIES IRREGULIERES ET DES GROUPES HOMOGENES

HCDE s'applique à tout groupe homogène mais s'appliquera plus avantageusement au plus grand groupe homogène possible pour un module donné. Autrement dit, si deux groupes homogènes sont formés de modules identiques et que l'on néglige les considérations relatives au dessin des

masques, il sera avantageux pour le rendement de l'ensemble de regrouper ces groupes en un seul plus important. Nous appellerons un groupe homogène propre le plus grand groupe homogène possible d'un type de module donné.

En résumé, un circuit typique est subdivisé en une section irrégulière et une série de groupes homogènes propres déterminés. HCDE s'applique à chaque groupe homogène, propre ou non. C'est l'estimation des avantages sur le rendement qui détermine s'il est avantageux d'appliquer HCDE à un groupe homogène donné.

2.8d-RESEAUX HETEROGENES

Lorsque les matrices de modules ne sont pas de dimension suffisante pour justifier l'utilisation de plusieurs niveaux hiérarchiques mais que la structure du réseau ou la géométrie de l'empaquetage nécessite un arrangement bidimensionnel, la partie irrégulière du réseau de communication peut être exclue des modules permettant alors la création d'un groupe homogène (exemple: figure 2.22).

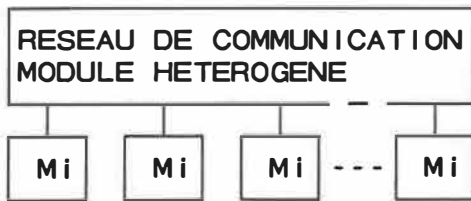


FIGURE 2.22: EXEMPLE DE GROUPE HOMOGENE
AVEC RESEAU DE COMMUNICATION ARBITRAIRE

Un segment homogène de modules n'a pas à être physiquement unidimensionnel, il peut être bidimensionnel au dessin des masques. On pourra économiser des niveaux hiérarchiques CDE en arrangeant les décodeurs décisionnels sous d'autres formes que des segments: serpentín, spirale, etc. Ceci est avantageux lorsque les canaux de communication entre les modules sont larges (occupant une grande surface), que le nombre de modules par rangée est bas et que r est élevé. Le nombre total de modules de rechange peut alors être réduit en groupant en un seul pseudo-segment des segments distincts, faisant circuler le signal m_r (section 2.6c) en serpentín (exemple: figure 2.23). Ces arrangements en pseudo-segments de modules se comporteront de manière identique à celle de segments si les branchements des multiplexeurs de reconfiguration tiennent compte de l'ordre des modules: toutes les entrées/sorties sont traitées comme des entrées de type vertical ($r+1$ positions).

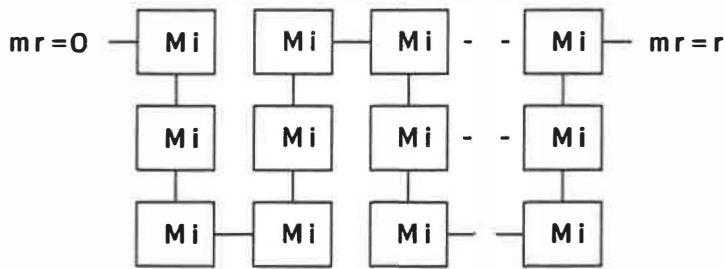


FIGURE 2.23: EXEMPLE DE PSEUDO-SEGMENT CHAINON CDE EN SERPENTIN.

La nature fréquemment régulière des instances des bornes d'entrée, de sortie et d'entrée-sortie à trois états, permet la formation de groupes homogènes de chaque type, permettant ainsi l'amélioration du rendement des bornes d'interconnexion. Il en est de même en ce qui concerne les barres de distribution des signaux de commande et de données si leur superficie justifie l'utilisation de HCDE.

2.8e-TECHNOLOGIES COMPLEMENTAIRES

HCDE est compatible avec d'autres techniques d'amélioration de rendement. Cette section vise à donner un aperçu de celles qui, à notre avis, compléteront le plus avantageusement HCDE. Notre objectif est de minimiser les coûts par fonction. Celui-ci est directement affecté par la probabilité de défaut non-reconfigurable (causant le rejet d'un circuit), que nous chercherons à minimiser.

Moore et al [25] a proposé de relaxer les règles de design; il indique notamment que la densité de défauts est réduite d'un facteur de 4 lorsque les règles sont relaxées d'un facteur de deux. Nous pouvons pratiquement éliminer la probabilité de perdre deux modules par un seul défaut (court-circuit entre deux lignes) en éloignant quelque peu (ex: de 1 à 2 carrés technologiques) des modules adjacents. Ceci peut permettre de réduire la valeur de r d'un niveau hiérarchique ou augmenter le rendement de l'ensemble. On peut aussi distancer légèrement les lignes distribuées aux modules. On peut grouper en îlots les transistors de même type. Il peut être avantageux de relaxer les règles de design pour certaines lignes de commande cruciales (horloges, etc). Certaines couches sont moins sensibles à certains types de défauts que d'autres. Par exemple, un second métal est moins sujet à des court-circuits au substrat qu'un premier métal. Il sera avantageux de disposer préférentiellement les lignes intolérantes aux défauts du niveau hiérarchique supérieur sur ces couches moins sensibles.

La probabilité de court-circuit entre les alimentations augmente avec la superficie d'un circuit, que ce circuit soit tolérant aux défauts ou pas. Un module pour lequel tous les signaux logiques sont tolérants ne l'est pas nécessairement en ce qui a trait à une défectuosité

(court-circuit) aux alimentations; il en est de même des signaux distribués à tous les modules. Ce circuit fixe réduit l'amélioration du rendement. Il sera donc avantageux de chercher à minimiser cette "charge intolérante" des circuits remplaçables. Il est possible de débrancher les circuits non-utilisés au moyen d'interrupteurs ou de disjoncteurs (transistors) de puissance aux alimentations. On réduira ainsi non seulement la probabilité de court-circuit de l'alimentation au substrat mais aussi la consommation des modules inutilisés. Un disjoncteur peut être réalisé au moyen d'un transistor de grande surface commandé par un circuit de détection de court-circuit entre les alimentations tel que proposé par Agarwal et al [26]. On peut aussi réduire la sensibilité des lignes distribuées à tous les modules en insérant aux entrées de chaque module des tampons de distribution de signaux de commande locaux.

Lorsqu'un "bus" de distribution de signaux occupe une grande superficie, le rendement, même après application du facteur de correction proposé à la section 3.3b (formule 9), peut être bas à cause des superficies impliquées. On peut réduire la probabilité de court-circuit au substrat en intercalant une couche écran (exemple: en poly) séparant les barres de signaux et le substrat. Nous protégeons ainsi les signaux contre une certaine fraction des défauts possibles.

En combinant cette technique avec un ajustement des espacements, une réduction significative de la probabilité de défaut peut être obtenue. Un facteur de correction additionnel de $1/2$ à $1/4$ peut alors être utilisé pour estimer la superficie dans le calcul du rendement. L'écran est une séquence de boîtes isolées entre-elles suivant les lignes à protéger sur toute leur longueur [26].

Une autre type de technique consiste à ajouter de la redondance par recodage de signaux. En recodant un signal selon un code prédéfini, il est possible d'inclure suffisamment d'information redondante pour pouvoir détecter la présence de fautes (code de parité) ou encore mieux, identifier leur localisation (code de hamming). Les codes convolutionnels visent aussi ce type de tolérance aux défauts. Nous envisageons que ce type de technique sera avantageux pour améliorer la fiabilité des lignes de communication séparant les boîtiers d'un système. Nous ne disposons malheureusement pas de l'espace et du temps nécessaires pour élaborer ici sur ce sujet fascinant.

2.9-CONCLUSION

Un mécanisme d'auto vérification et d'auto-reconfiguration a été proposé ici. Les mécanismes

peuvent être étendus à des circuits de complexité arbitrairement grande, la capacité d'expansion étant de nature exponentielle (hiérarchique). Cette propriété de HCDE comme technique d'amélioration du rendement s'accorde bien à la nature aussi exponentielle de l'évolution de la complexité des circuits intégrés. La technique s'adapte à des réseaux multidimensionnels. Bien que la technique s'attaque d'abord aux circuits réguliers, elle peut participer à l'amélioration du rendement de circuits monolithiques complexes tels: processeurs RISC, multiplieurs, FPLAs, LCAs, RAMs, ALUs parallèles, SIMDs, MIMDs, etc. Les critères d'applicabilité de la technique sont:

- 1- le circuit doit être, au moins partiellement, régulier,
- 2- le module du premier niveau hiérarchique CDE doit être vérifiable,
- 3- le module du 1^{ier} niveau hiérarchique CDE doit avoir un rendement suffisant.

Un des avantages principaux de la méthode HCDE est que la complexité de la tâche de vérification des circuits est réduite à celle de vérifier le module élémentaire du niveau hiérarchique CDE le plus bas, plutôt que de vérifier à l'échelle du circuit complet (généralement impossible).

Ceci est valable quel que soit le nombre de modules dans le système, tous les modules étant vérifiés et reconfigurés simultanément. Les économies de vérification et reconfiguration sont proportionnelles à la quantité totale de modules identiques par système. Le diagnostic et la reconfiguration étant effectués localement, ces fonctions sont réalisées sans l'intervention de systèmes hôtes alors informés séquentiellement. En conséquence, la réalisation de circuits intégrés de très grande complexité (à l'échelle d'une tranche) est économiquement envisageable si les systèmes rencontrent les critères précités.

L'utilisation de mémoire vive pour le diagnostic et la reconfiguration permet la vérification des systèmes sur le site, par l'utilisateur. Ceci augmente la disponibilité des systèmes et en réduit les coûts d'entretien. Dans le cas de fautes se produisant dans le circuit reconfigurable, le temps de réparation peut être aussi court que quelques fractions de seconde, pour des systèmes basés sur des modules de faible complexité (exemple: d'IMAGE2 =800 transistors/module). La technique permet la vérification et la reconfiguration sans l'intervention d'équipements externes complexes ou spécialisés (ex: micro-chirurgie par laser). Enfin HCDE s'harmonise avec d'autres techniques complémentaires d'amélioration du rendement.

CHAPITRE 3

ESTIMATION DU RENDEMENT

3.1 - INTRODUCTION

Le but de ce chapitre est de présenter une formulation efficace du rendement de circuits de grande superficie auxquels on a appliqué HCDE. Nous cherchons une formulation relativement simple et pouvant s'adapter à un éventuel algorithme automatique de sélection des paramètres HCDE optimaux d'un circuit.

Selon nous, l'estimation du rendement est autant un art qu'une science. Le second est certain, le premier est utilisé ici pour indiquer qu'il n'existe à ce jour aucune méthode exacte d'estimation du rendement d'un procédé de fabrication. Par conséquent, plusieurs approches fondées sur l'expérience sont valables. Nous discutons donc d'estimation plutôt que de détermination du rendement. Certains auteurs ont déjà présenté des résultats sur l'estimation du rendement en présence de redondance [6]. Les formulations proposées varient d'un auteur à l'autre suivant la nature des modèles statistiques utilisés. Chacune des approches cherche à tenir compte de la nature stochastique (statistique) de l'occurrence des défauts.

La densité de défauts (D) est la quantité moyenne de défauts par unité de superficie de circuit (en défauts/cm²). Certaines formulations tiennent compte d'un facteur additionnel d'entassement ("clustering": α) des défauts; les défauts ont en effet tendance, à l'échelle macroscopique (tranche: "WSI"), à se présenter en grappes. Cette instabilité de la densité de distribution des défauts affecte localement la valeur de D .

Certaines formulations comme Schvan et al [7] estiment séparément les rendements de plusieurs type/mode de défauts, utilisant des paramètres D_i et α_i différents pour chacun. Certains types de défauts peuvent affecter plus qu'une couche (ex: court-circuit entre deux couches, etc) ou plus qu'un noeud (ex: court-circuit entre deux lignes, lignes parallèles ouvertes par une poussière commune). Par conséquent la modélisation des modes de défauts est beaucoup plus complexe qu'il apparaît à première vue à cause d'un phénomène d'interdépendance entre l'occurrence des modes.

Les mesures statistiques se font, en principe, sur des échantillonnages importants. Par conséquent, la distinction des modes de défauts requiert un échantillonnage plus considérable. Le coût d'estimation des paramètres augmente nécessairement avec leur nombre à cause de

l'échantillonnage requis. Il est ainsi plus coûteux de tenir compte de l'entassement et plus coûteux encore de différencier les modes de défauts. Ajoutons: 1-qu'il est fréquent que la densité de défaut varie avec la distance au centre d'une tranche ("wafer"), 2-que les paramètres évoluent par sauts dans le temps, 3-que certains défauts importants (superficie), survenant fortuitement, sont imprévisibles (bri de tranche, erreur dans le découpage en puces, mésalignement d'un masque, manutention inexperte/brutale, etc); et 4-qu'un défaut paramétrique peut affecter une tranche entière. On conçoit donc que l'on puisse considérer l'estimation du rendement comme étant à la fois une science et un art.

En pratique, on ne tient compte que des défauts prévisibles. Un concepteur est ainsi limité par les données dont il dispose sur le procédé (paramètres statistiques) et par son expérience (pouvant lui fournir des données générales complémentaires). Les rendements calculés excluent généralement les défauts paramétriques (une tranche qui est rejetée à-priori) et les défauts fortuits. Le paramètre d'entassement α est souvent difficile à obtenir des manufacturiers.

Une formulation simple ne tenant pas compte de

l'entassement des défauts se base sur une statistique de type Poisson. Un reproche fait à cette technique est que lorsque le rendement calculé est faible en l'absence de tolérance aux défauts, les résultats sont pessimistes présentant des rendements plus faibles que la réalité. Conséquemment, un facteur de mérite calculé sur la base de statistiques de Poisson peut devenir largement optimiste pour des circuits de grande superficie.

Thibeault et al [8] a présenté une formulation du rendement tenant compte du facteur d'entassement ("clustering") des défauts. Sa formulation est donc plus exacte. Il montre comment calculer le rendement d'un segment homogène comportant r modules de rechange. La formulation présentée est valide pour les cas d'une hiérarchie à un ou deux niveaux reconfigurables. Celle-ci devient de plus en plus complexe lorsque le nombre de niveaux hiérarchiques augmente. Puisque cette formulation fait partie d'un autre projet, nous avons évité d'élaborer sur ce sujet.

Bien que le type de formulation proposée par Thibeault [8] soit plus précis, nous considérerons comme acceptable, dans un premier temps, une formulation sur la base de statistiques de Poisson (section 3.2). On constate en effet, trois avantages importants à l'approche

Poissonienne: 1-la formulation est relativement simple (l'opération élémentaire est transitive) et les paramètres sont plus facilement mesurables (moins de paramètres: requiert moins d'échantillons), 2-elle est aisément compréhensible (se conçoit bien intuitivement) et applicable (complexité des calculs limitée), 3-elle est indépendante du nombre de niveaux hiérarchiques et donc complète en elle même dès le premier niveau reconfigurable.

Il est certain qu'un facteur d'entassement ("clustering") des défauts produit des rendements calculés plus exacts. Nous croyons cependant que les hauts rendements obtenus par l'application de HCDE font que la statistique de Poisson n'est pas aussi inexacte dans l'estimation des rendements améliorés que pour l'estimation des rendements de circuits de superficie équivalente sans HCDE. Ceci, selon nous, parce que l'entassement des défauts aura tendance à diminuer le nombre de modules défectueux plutôt que de l'augmenter. Parce que l'approche hiérarchique utilisée avec HCDE garantit que le niveau hiérarchique à partir duquel le facteur d'entassement se fait sentir est tolérant aux défauts, l'entassement présente un avantage pour HCDE. Il y aura donc plus de modules de rechange, que requis, pour profiter de cet avantage si les calculs sont faits en utilisant la statistique de Poisson. L'utilisation de cette

statistique peut alors ne pas être aussi imprécise qu'il semblerait au premier abord et produire un estimé conservateur du rendement.

Les résultats préliminaires de Thibeault et al [8] semblent confirmer notre approche. Selon cet article, le type de statistique utilisé affecte légèrement le facteur de mérite, mais très peu le choix optimal de r pour un n donné lorsque ce choix se base sur le facteur de mérite calculé. Conséquemment, bien que les facteurs de mérites calculés soient supérieurs avec une statistique de Poisson, la détermination des paramètres optimaux ne sera pas nécessairement très différente de celle qui serait obtenue avec une formulation plus exacte du rendement. Nous sommes néanmoins d'accord avec Thibeault qui précise qu'une analyse plus exhaustive est requise pour conclure sur les critères d'applicabilité de chaque formulation.

3.2-ESTIMATION DU RENDEMENT

(STATISTIQUE POISSONNIENNE)

Cette section présente une formulation du rendement basée sur une statistique de Poisson. Trois équations élémentaires suffisent à couvrir tous les cas: 1-calcul du rendement d'un circuit sans redondance (équation 1) extraite

de Savaria [5], 2-calcul du rendement d'un groupe hétérogène (équation 2), 3-calcul du rendement d'un segment de modules avec tolérance aux défauts de type CDE (équation 3). Nous remarquons l'avantage principal du calcul sur la base de statistique de Poisson, apparent à la seconde équation, où le rendement d'un groupe hétérogène est égal au produit des rendements des modules individuels. Cette propriété d'indépendance simplifie considérablement les calculs, particulièrement lorsque le nombre de niveaux hiérarchiques est élevé et lorsqu'un circuit comporte plusieurs groupes homogènes propres.

$$R = \exp (-AD) \quad (1)$$

où R est le rendement du circuit

A en est la superficie

D est le nombre de défauts par cm^2 du procédé

$$R_g = \prod_{j=1}^J R_j \quad (2)$$

où R_g est le rendement du groupe hétérogène

R_j est le rendement d'un module du groupe

J est le nombre de modules

$$R_s = R_i * \sum_{k=0}^{k=r} C(n+r,k) * D_m^k * R_m^{n+r-k} \quad (3)$$

où R_s est le rendement du segment

R_i est le rendement du circuit intolérant aux défauts du segment

$R_m = (1-D_m)$ est le rendement d'un module

D_m est la probabilité de défaut d'un module

$C(n+r,k)$ est le nombre de combinaisons de k dans $n+r$

r est le nombre de modules de rechange

n est le nombre de modules actifs

La troisième équation est un cas particulier de la seconde: pour un groupe homogène, les modules étant identiques, le produit des rendements des modules individuels est égal au rendement du module (R_m), à la puissance du nombre de modules ($n+r$). L'équation tient aussi compte de la tolérance aux défauts: la probabilité de rencontrer moins que $r+1$ défauts dans un segment comportant r modules de rechange. Pour ceci, nous calculons les probabilités de rencontrer zéro, un, deux, jusqu'à r défauts par segment. Ces cas sont les probabilités de k défauts et de $n+r-k$ PE fonctionnels multiplié par le nombre de combinaisons de k dans $n+r$ objets. Le rendement du segment est ainsi la probabilité d'obtenir r modules défectueux ou moins. Nous concluons que le rendement calculé d'un groupe homogène est conservateur parce que le facteur d'entassement augmente la probabilité d'avoir deux défauts dans un même module.

Pour calculer le rendement d'un groupe homogène, on utilise l'équation 1 pour calculer le rendement du module (simple, composé, ou fractionné) du niveau hiérarchique le plus bas et des parties intolérantes aux défauts de chaque niveau hiérarchique (équation 4). On applique l'équation 3 pour calculer le rendement du second niveau hiérarchique (segment de modules) (équation 5), ce rendement est celui du module du second niveau: R_2 . La troisième équation est réappliquée à chaque niveau hiérarchique CDE successif jusqu'au niveau hétérogène (équation 6). On obtient ainsi la formulation (4 à 6) pour le rendement d'un groupe homogène de un à "p" niveaux hiérarchiques CDE (équation 7). Les niveaux hétérogènes (puce, etc) sont calculés en appliquant la troisième équation (équation 8).

$$R_1 = \exp(-AD) \quad (4)$$

où R_1 est le rendement du module du premier niveau
 A en est la superficie du module
 D est le nombre de défauts par cm^2 du procédé

78.

$$R_2 = R_{i2} * \sum_{k=0}^{k=r2} C(n2+r2, n2) D_1^k R_1^{n2+r2-k} \quad (5)$$

où R_2 est le rendement du module du second niveau hiérarchique

R_{i2} est le rendement du circuit intolérant aux défauts du second niveau

$R_1 = (1-D_1)$ est le rendement d'un module du premier niveau

D_1 est la probabilité de défaut d'un module du premier niveau

$n2$ est le nombre de modules fonctionnels au second niveau

$r2$ est le nombre de modules de remplacement au second niveau

$$R_p = R_{ip} * \sum_{k=0}^{k=rp} C(np+rp, np) D_{p-1}^k R_{p-1}^{np+rp-k} \quad (6)$$

où R_p est le rendement du pième niveau hiérarchique

R_{ip} est le rendement du circuit intolérant aux défauts du pième niveau

$D_{p-1} = 1-R_{p-1}$

C est le nombre de combinaisons

rp est le nombre de modules de rechange

np est le nombre de modules actifs

p est l'indice du niveau hiérarchique

$$R_{hoq} = R_p \quad (7)$$

où R_{hoq} est le rendement du groupe homogène q

p est le niveau hiérarchique supérieur du groupe homogène q

$$R_{héu} = \prod_{t=1}^T R_{hét} \prod_{s=1}^S R_{hos} \quad (8)$$

où T est le nombre de modules du groupe hétérogène

S est le nombre de groupes homogènes du niveau

$R_{héu}$ est le rendement du groupe hétérogène du niveau u

R_{hos} est le rendement du groupe homogène s

Cette formulation permet de calculer le rendement de circuits, auxquels on a appliqué HCDE, sur la base

de statistiques Poissonniennes (sans tenir compte de l'entassement des défauts). La section 5.5 présente un exemple de calcul de rendement suivant cette formulation pour un système MIMD comportant 5 systèmes SIMDs, avec 1024 PEs fonctionnels par système SIMD.

3.3-ESTIMATION DE LA SUPERFICIE

3.3a-SUPERFICIE INTOLERANTE AUX DEFAUTS

La superficie d'un circuit a un impact prépondérant sur la valeur du rendement calculé. Lorsqu'un circuit est tolérant aux défauts, la superficie intolérante aux défauts prend plus d'importance dans le calcul du rendement que sa contrepartie tolérante aux défauts. Cette section sépare plus précisément les portions, intolérantes et tolérantes aux défauts, du circuit. La section 3.3b discute, plus particulièrement de l'influence du circuit intolérant aux défauts ne couvrant qu'une seule couche sur le rendement d'un circuit tolérant aux défauts.

Les régions d'un niveau hiérarchique se subdivisent essentiellement en trois groupes pour chaque niveau hiérarchique: 1-tolérante aux défauts, 2-intolérante à plusieurs couches et 3-intolérante à une seule couche (section 3.3b).

Chaque segment de modules est formé de 2 parties distinctes: 1-un segment régulier de modules identiques, 2-un circuit complémentaire commun irrégulier. La seconde n'est pas tolérante aux défauts. La première (module régulier) se subdivise comme suit: 1-le circuit complémentaire intolérant aux défauts, 2-le circuit fonctionnel intolérant aux défauts, 3-le circuit complémentaire tolérant aux défauts et 4-le circuit fonctionnel tolérant aux défauts. C'est dans ces quatre parties que se joue l'amélioration du rendement. Cette subdivision supplémentaire est importante car chaque élément de surface du module répétable revient $n+r$ fois, multipliant d'autant la superficie intolérante aux défauts.

Le second chapitre délimite de manière générale les parties tolérantes et intolérantes aux défauts d'un circuit complémentaire typique. Le chapitre 5 délimite quelles parties de circuit sont tolérantes et intolérantes aux défauts dans les cas de la puce IMAGE2 et d'un système MIMD comportant 5120 processeurs fonctionnels considéré comme exemple. Il serait redondant de reprendre ici cette description.

Dans le calcul du rendement, les superficies équivalentes intolérantes aux défauts seront combinées pour calculer le rendement du module intolérant. La superficie tolérante aux défauts sera elle directement utilisée comme facteur de calcul du rendement (section 3.2).

3.3b-CIRCUITS INTOLERANTS MONOCOUCHE

Lorsque nous avons appliqué HCDE à la puce IMAGE2, nous avons constaté qu'une grande proportion du circuit intolérant aux défauts ne couvrait qu'une seule couche. L'importance relative de la superficie de ce circuit monocouche versus le circuit multicouche du circuit intolérant aux défauts peut affecter significativement l'exactitude des rendements calculés. Nous proposons d'atténuer cet effet en appliquant un facteur de correction dans le calcul de la superficie du circuit intolérant aux défauts ne couvrant qu'une seule couche.

Si nous posons l'hypothèse qu'une proportion importante des défauts provient de poussières affectant le transfert normal de l'information géométrique entre les masques et le substrat lors de la fabrication, la probabilité de rencontrer une poussière en un point est alors, en première approximation, proportionnelle au nombre

de couches. Nous considérerons trois bases de calcul possibles: 1-la moyenne du nombre de couches simultanées dont les dimensions sont de taille minimale (cas pessimiste), 2-le nombre maximal de couches en un point (cas optimiste) et 3-le nombre maximal de couches de taille minimale en un point (cas réaliste). Les couches de taille minimale incluent: 1-poly, 2-diffusion, 3-métal1, 4-métal2, 5-contact, 6-via (n+ et p+ ne sont pas de dimension minimales). Posant l'hypothèse d'équiprobabilité des défauts aux couches différentes, nous estimons la moyenne d'environ 2.5 couches simultanées (estimation visuelle sur la base du dessin des masques pour IMAGE2). Le nombre maximal de couches correspond au cas d'un contact bipolaire de transistor N dans le cas du procédé 3 um de la CMC. Ce cas couvre 7 couches (métal2, métal1, contact, puit du transistor, puit P, P+, N+). La troisième base allège la seconde des gardes et des puits laissant 4 couches.

Nous avons rejeté la seconde base de calcul pour son évident manque de discernement. Une poussière n'affecte pas une grande superficie aussi catastrophiquement que dans le cas d'une boîte de taille minimale. Nous envisageons donc qu'une estimation de la superficie de chaque couche à laquelle nous appliquons un facteur de correction se trouvant entre 1/2.5 et 1/4 permettra d'estimer le rendement

du circuit intolérant aux défauts ne couvrant qu'une seule couche (équation 9). Dans nos calculs, nous avons utilisé un facteur de 1/4. Le reste du circuit ne reçoit pas de facteur de correction de la superficie. Le facteur est utilisé pour l'estimation du rendement et n'entre pas dans le calcul de la superficie lors de l'estimation du facteur de mérite.

Nous avons donc une superficie pour fins de calcul du rendement et une superficie réelle.

$$A_{i\acute{e}} = f_1 \times A_{i1} + A_{im} \quad (9)$$

où $A_{i\acute{e}}$ est la superficie équivalente du circuit intolérante aux défauts pour le calcul du rendement
 A_{i1} est la partie de A_i ne couvrant qu'une seule couche
 A_{im} est la partie de A_i comportant plusieurs couches
 f_1 est le facteur de correction de la superficie monocouche

Dans nos mesures sur la puce IMAGE2 (section 5.4), nous avons soustrait la superficie monocouche de la superficie totale du PE après l'application du facteur de correction f_1 , afin d'assurer que toutes les parties de circuit entraient dans l'estimé du calcul de rendement.

3.4-CRITERE D'OPTIMALITE (FM: facteur de mérite)

Nous avons considéré comme critère d'optimalité le facteur de mérite de Mangir [9], (équation 10). Il s'agit

essentiellement d'une mesure de la valeur d'une solution. L'équation tient compte des superficies de silicium utiles et rejetées (cas sans redondance) plutôt que seulement la superficie d'un circuit fonctionnel. Le coût du cas sans redondance sert de référence et le facteur de mérite est effectivement un estimé du gain de performance par unité de surface de silicium utilisé qui résulte de l'application de HCDE. Les deux coûts (avec et sans tolérance aux défauts) sont simplement factorisés.

$$FMM = \frac{Y_r}{Y_0} \times \frac{A_0}{A_r} \quad (10)$$

où A_0 est la superficie sans redondance,
 A_r est la superficie avec redondance,
 Y_0 est le rendement sans redondance,
 Y_r est le rendement avec redondance,
 FMM est le facteur de mérite de Mangir

Le pessimisme des estimés de rendement sur base Poissonnienne augmente avec la superficie par rapport aux estimés tenant compte du facteur d'entassement α . Puisque HCDE vise à réaliser des circuits de très grande superficie, le rendement de la solution sans tolérance aux défauts d'un circuit, étant de plus faible rendement, sera relativement plus pessimiste. Puisque ce cas (sans tolérance aux défauts) sert de référence dans le calcul du facteur de mérite (section 3.4) des autres solutions, les facteurs de mérite calculés seront optimistes relativement à la réalité.

Nous proposons de compenser cet effet en calculant le rendement du circuit sans redondance en tenant compte du facteur d'entassement (équation 11). Ainsi, l'inexactitude de l'estimation Poissonnienne dans les cas de faible rendement sera réduite et présentera un facteur de mérite calculé plus conservateur et plus près de la réalité, particulièrement pour des circuits de grande superficie. Les résultats relatifs des diverses solutions possibles pour un groupe homogène donné seraient cohérents entre-eux, même si le critère d'optimalité (facteur de mérite) était inexact. Le facteur amélioré sera plus apte à estimer le gain d'une solution (estimé pessimiste: gain minimal escompté).

$$R = \left[1 + \frac{AD}{\alpha} \right]^{-\alpha} \quad (11)$$

où R est le rendement du circuit sans redondance
 A en est la superficie
 D est le nombre de défauts par cm² du procédé
 α est le facteur d'entassement

Pour être complète, la formulation du facteur de mérite devrait tenir compte des délais encourus par l'application de HCDE (délais additionnels dûs aux interrupteurs de reconfiguration). Nous négligeons ce facteur à cette étape-ci dans le but de minimiser le temps de calcul pour l'optimisation des paramètres HCDE. Nous laissons au concepteur de circuit la responsabilité du contrôle du chemin critique (délai de propagation du canal

le plus lent). Quant-à nous, il nous est impossible de déterminer à priori si les interrupteurs de reconfiguration se situeront sur le chemin critique d'un circuit. Les délais de propagation sont affectés par le "nombre de niveaux hiérarchiques" les largeurs des modules et nombre de modules de rechange de chaque niveau hiérarchique. Le nombre de modules fonctionnels n'affecte pas le délai de propagation du point de vue de l'application de HCDE.

Nous croyons cependant qu'un facteur de mérite basé sur la fonctionnalité par puce à rendement équivalent pourra être un critère plus valable que le facteur de Mangir. Ce facteur de mérite fonctionnel (FMF) peut se définir comme le gain de fonctionnalité obtenu par l'utilisation de HCDE à rendements comparables. Il se calculerait en déterminant le nombre de modules répétables réalisant un rendement identique à celui de la solution HCDE et en effectuant le rapport des fonctionnalités obtenues. Ce rapport a l'avantage de comparer deux solutions réalistes plutôt que la solution HCDE avec une solution irréaliste (rendement médiocre: près de 0) dans le cas de circuits de grande superficie.

3.5- TRAVAUX FUTURS

Nous croyons que pour que HCDE soit utilisé couramment, des outils permettant l'optimisation des paramètres HCDE sont requis. Nous proposons le développement d'un algorithme permettant, sur la base d'une formulation du rendement de déterminer les paramètres optimaux: 1-nombre de niveaux hiérarchiques, 2- n et r pour chaque niveau hiérarchique, 3-orientation de chaque niveau.

Dans cet esprit, une formulation plus exhaustive de la superficie intolérante aux défauts basée sur le nombre d'entrées verticales, horizontales et diagonales peut être développée comme outil de base pour un logiciel d'optimisation des paramètres HCDE. Une formulation du rendement tenant compte du facteur d'entassement sera sûrement utile pour vérifier l'optimalité des solutions obtenues au moyen de notre formulation Poissonnienne.

Une solution manuelle, heuristique, ou optimale, est intéressante si la superficie d'un circuit est restreinte (superficie limitée par l'empaquetage, le nombre de bornes disponibles, etc). Une technique automatique simple consisterait à calculer systématiquement le critère d'optimalité (section 3.4: facteur de mérite) de chaque

solution possible, sélectionnant ensuite par inspection la solution la plus avantageuse (optimisation par programmation non-linéaire).

Il serait intéressant de pouvoir identifier certains paramètres optimaux au moyen d'un abaque. Le nombre de paramètres à optimiser n'aide pas à une telle représentation. Nous avons pensé à une solution consistant à représenter certains paramètres sous forme de rapport plutôt que de valeurs absolues. Cherchant une représentation permettant de restreindre le nombre de variables à une quantité pouvant être représentée graphiquement, nous avons entrevu 3 facteurs possibles: 1-le rapport de la superficie du circuit complémentaire sur celle de la superficie fonctionnelle (S_c/S_t) d'un module, 2-le rapport des rendements des circuits tolérants et intolérants aux défauts du module (R_t/R_i), le rapport du nombre de modules de remplacement (r) sur le nombre de processeurs (n) produisant le plus haut facteur de mérite ($R_{\text{opt}}=(r/n)|_{\text{opt}}$). Un tel graphique ou tableau permettrait d'estimer visuellement une solution relativement optimale. Nous invitons nos successeurs à tenter ce type de représentation.

CHAPITRE 4

EXEMPLE D'IMPLANTATION, PARTIE FONCTIONNELLE

4.1- INTRODUCTION

En complément au chapitre 2 qui décrit HCDE de manière générale, ce quatrième chapitre décrit la partie fonctionnelle d'un exemple d'implantation: IMAGE2. Le chapitre 5 décrit l'aspect HCDE d'IMAGE2.

Pour augmenter l'intérêt, nous avons choisi une application d'actualité, répondant à des besoins réels. Ceci présente un avantage potentiel d'applications commerciales des architectures proposées, en prévision de l'éventuelle disponibilité de procédés pouvant supporter un niveau d'intégration suffisant. Pour conserver une chance de succès commercial, nous avons imposé un critère projetant notre application en avance sur son temps. C'est pour cette raison que nous avons imposé les critères additionnels, contradictoires en 1985, traitement 1-global, 2-en temps réel et 3-économique. La faisabilité devait être démontrable au moyen d'un procédé de densité restreinte: $3\mu\text{m}$, 0.25 cm^2 .

Notre choix s'est finalement arrêté sur le traitement économique d'images en temps réel. Ce choix répondait

naturellement à nos critères de sélection: 1-la puissance de calcul requise, 2-l'actualité du problème, 3-l'ampleur du marché (vision artificielle, robotique, télévision, télématique, imagerie bio-médicale, etc). En effet, même en 1989, il n'existe aucun système de traitement d'images en temps réel qui soit à la fois performant (30 à 60 images par seconde, capable d'effectuer plusieurs opérations par pixel) et économique (de \$5000 à \$10000). Il existe des systèmes soit disant en temps réel, traitant régulièrement une fraction des images reçues; notre définition de traitement en temps réel est plus exigeante: 1-traitement de chaque image à la fréquence des images reçues (30 à 60 images par seconde: standard NTSC, PAL, SECAM, ou HDTV) et 2-un traitement complet, c'est-à-dire jusqu'à reconnaissance d'objet ou de scène.

En démontrant la faisabilité d'un système économique de traitement d'images en temps réel, nous démontrons du même coup un des avantages principaux de HCDE: augmentation de la capacité fonctionnelle des puces et réduction du coût des systèmes. L'architecture de système proposée à la section 4.3 est suffisamment puissante pour répondre aux besoins les plus exigeants en puissance de calcul et suffisamment flexible pour simultanément répondre à une grande variété de besoins. IMAGE2 est une étape dans le

développement d'un tel système.

HCDE permettait d'envisager de projeter d'une ou deux générations vers le futur la densité de transistors par puce, apportant ainsi un élément de solution à l'aspect économique: moins de puces, impliquant moins de cartes, ce qui conduit à des boîtiers plus compact, donc des systèmes plus économiques. L'application présente alors un intérêt commercial sur la seule base de l'ampleur des marchés de la robotique, de la vision artificielle et de la sécurité.

Les chapitres 4 et 5 rencontrent quatre objectifs: 1-démontrer que HCDE peut être pratiquement implantée, 2-décrire un exemple d'implantation de HCDE (IMAGE2), 3-présenter des résultats réalistes sur les gains de rendement/fonctionnalité possibles sans tenter d'optimiser le facteur de mérite et 4-présenter un concept d'architecture de système de traitement d'image suffisamment performante et flexible pour effectuer un traitement en temps réel. Les trois premiers objectifs étayent notre démonstration des avantages qui découlent de HCDE en ce qui a trait au rendement et la fonctionnalité des circuits. Le quatrième est notre contribution au domaine du traitement d'images où nous croyons qu'un traitement économique, en temps réel est possible par l'utilisation de HCDE.

La section 4.2 introduit le domaine d'application: le traitement d'images en temps réel. La section 4.3 décrit le concept d'architecture globale (IMAGE) que nous proposons comme solution aux problèmes identifiés en 4.2 (puissance de calcul, bandes passantes, mémoire). Notre proposition est une combinaison des architectures cousines 1-matrice de pipelines (array of pipelines) et pipeline présentées par Riesenbach [22].

La section 4.4 décrit plus spécifiquement l'architecture du processeur et des réseaux de communication implantés dans IMAGE2. A la section 4.5, nous présentons la partie fonctionnelle d'un système de $32 * 32$ PEs, proposant du même coup certaines améliorations au PE pour augmenter la puissance de calcul, mieux balancer l'interface pipeline et restreindre le nombre de broches au système. Nous proposons à la section 4.6, notre vision de quelques travaux futurs complémentaires à la présentation de ce chapitre. L'aspect HCDE d'IMAGE2 est couvert au cinquième chapitre.

4.2-TRAITEMENT D'IMAGES EN TEMPS REEL

(domaine d'application d'IMAGE2)

Le traitement d'images est un sujet relativement vaste, nous ne présentons qu'une partie de la très volumineuse littérature sur le sujet [10 à 20]. Une image est essentiellement une matrice bidimensionnelle de pixels ("picture elements"). Un signal vidéo est formé d'une séquence d'images consécutives, formant ainsi une matrice tridimensionnelle de pixels. C'est ce facteur qui détermine la qualité temps réel des besoins. La vision artificielle est un processus continu. La complexité des images (nombre de pixels par image, nombre de couleurs ou spectre) dépend, bien sûr, des besoins de chaque application.

4.2a-ESTIMATION DES BANDES PASSANTES

Un signal vidéo comporte une série de signaux de référence complémentaires stabilisant spatialement et temporellement la qualité de l'information de chaque pixel: synchronisation horizontale, niveau de blanc, niveau de noir, synchronisation de phase des signaux de couleur, synchronisation verticale, entrelacement, etc. Des nombreuses normes vidéo existantes, quelques-unes ont atteint un usage international. La référence [21] en définit

3 des plus répandues (NTSC, PAL, SECAM). Certains groupes cherchent à valoriser certaines normes de plus haute qualité (HDTV): plus de pixels par image. Afin de simplifier le texte, nous utiliserons, à titre d'exemple, le standard nord-américain: NTSC (national television standard committee); pour définir les besoins d'un système de traitement en temps réel. Un signal nord-américain typique de type NTSC est passablement complexe et nous ne voulons pas ici en donner une description exhaustive. Nous allons simplement identifier le nombre de pixels par image et par seconde, dans le cas d'images noir et blanc et couleur.

La partie vidéo d'un signal NTSC comporte trois composantes: 1-luminance, 2-I et 3-Q. Le signal de luminance a une bande passante de 4.5 Mhz et correspond à l'image noir et blanc. Les signaux I et Q, correspondant aux signaux de couleur, sont modulés en SSB (single side band) et en quadrature sur une porteuse commune de 3.579545 Mhz. Si nous rétablissons la symétrie des bandes, Le signal I nécessite une largeur de bande de 1.8 Mhz et le signal Q requiert 1.2 Mhz. Un système de vision artificielle, différenciant la couleur, n'a pas à connaître spécifiquement la couleur réelle d'un objet (rouge, bleu, vert). Les signaux I et Q n'ont donc pas à être décodés en signaux bleu, rouge, vert. Deux approches d'échantillonnage peuvent

être utilisées selon que celui-ci est réalisé avant ou après la séparation des signaux de couleur. Un système de vision artificielle basé sur le standard NTSC ne requiert donc que trois échantillonnages respectifs de 9, 3.6 et 2.4 Mhz pour extraire le contenu vidéo d'un signal précédemment séparé.

Dans certains cas, il n'est pas avantageux de séparer les signaux de couleur de l'image noir et blanc, avant l'échantillonnage, (exemple: correcteur de base de temps pour l'élimination du glissement de signal provenant d'un système de transport de ruban: magnétoscope, magnétocassette, etc). Cette opération est nécessaire pour la synchronisation de signaux vidéo contenant du ronflement (variations dans la vitesse de déroulement d'une bande magnétique). Dans ce cas l'échantillonnage précède la séparation des signaux de couleur. Une fréquence d'échantillonnage de quatre fois la fréquence de la porteuse des signaux de couleur (14.31818 Mhz) est souvent utilisée. Ceci simplifie la séparation ultérieure des signaux en quadrature (X, Y: couleur) à une opération arithmétique simple. L'opération est ainsi indépendante de la phase de la porteuse qui demeure constante et de fiabilité comparable pour chacun des signaux en quadrature (I et Q).

Les signaux de synchronisation horizontale et de

niveaux (noir, blanc, couleur) occupent 16.5% de la durée d'une ligne et ne font pas, à proprement parler de l'image. La fréquence du signal de synchronisation horizontale est de 15734.264 par seconde. Chaque ligne d'une image de luminance (noir et blanc) NTSC comprend donc $9000000/15734.264 = 572$ échantillons, desquels nous retranchons 16.5% occupés par la synchronisation et les niveaux de référence laissant 478 échantillons par ligne. De façon similaire, chaque ligne des signaux I et Q comporte $3600000/15734.264 - 16.5\% = 191$ échantillons et $2400000/15734.264 - 16.5\% = 127$ échantillons. Dans le cas d'un échantillonnage pré séparation des signaux de couleur (14.31818 Mhz), chaque ligne contient alors $14318418/15734.264 - 16.5\% = 760$ échantillons.

Chaque image NTSC est formée de deux champs de 262.5 lignes chacun s'entrelaçant. Les signaux de synchronisation verticale occupent 9 lignes, laissant 253.5 lignes par champ (environ 60 par seconde). Une image NTSC (environ 30 par seconde) dépourvue de ses signaux de synchronisation verticale comporte donc $525 - 18 = 507$ lignes de pixels. Une image noir et blanc contient donc 507 lignes de 478 colonnes de pixels (éléments d'image) pour un total de 242346 pixels par image (30/sec) ou 121173 pixels par champ (60/sec) ou, à 30 images par seconde, 7.27 Mpixels/sec. Une image couleur

comporte, après séparation, trois images: 1-luminance de $507 \times 478 = 242346$ pixels/image, 2-I de $507 \times 191 = 96837$ pixels/image et 3-Q de $507 \times 127 = 64389$ pixels/image. Un système de traitement d'images couleur NTSC devra pouvoir traiter $507 \times (478 + 191 + 127) = 507 \times 796 = 403572$ "pixels" par image où, à 30 images par seconde, 12.11 mégapixels/sec.

Les bandes passantes pour les standards PAL et SECAM, bien que légèrement supérieures, sont du même ordre de grandeur. Un système de traitement d'images NTSC en temps réel doit pouvoir traiter jusqu'à 7.27, 12.11, ou 14.32 mégapixels/sec suivant que l'analyse est respectivement 1-noir et blanc, 2-couleur (échantillonnage post séparation), ou 3-couleur (échantillonnage pré séparation de I et Q) lorsque l'on veut traiter toute l'information disponible.

Les bandes passantes inférieures des signaux I et Q (couleur) du standard NTSC restreint l'usage de cette norme pour des terminaux graphiques ou vidéo de haute définition. Pour répondre à ces besoins, des signaux couleur (rouge, bleu, vert de bandes passantes identiques sont utilisés (autres normes). Cependant, les terminaux graphiques ne requièrent généralement pas de traitement à la vitesse de

rafraîchissement des images (30 à 60 fois par seconde) mais plutôt à la vitesse d'entrée des commandes manuelles par l'utilisateur, les images demeurant fixes entre les commandes. L'architecture que nous recommandons (section 4.3), est adaptable et suffisamment performante pour répondre aux applications de traitement haute définition, en temps réel et à la fréquence de rafraîchissement des images (simulateurs de vol, etc). A titre d'exemple, un écran couleur présentant 60 images par seconde, d'une définition de 1200 colonnes par 1024 rangées requiert un système capable de traiter $3 \times 1200 \times 1024 = 3.69$ mégapixels/image et 221 mégapixels/sec.

4.2b-PRECISION DES OPERATIONS

La précision des échantillons de sortie est un facteur qualitatif. Il est couramment reconnu qu'un échantillonnage à 8 bits/pixel reproduit une image avec une qualité non-différentiable d'une image originale, même pour un oeil expert et averti. Un échantillonnage à 6 bits/pixel est souvent considéré de bonne qualité pour un récepteur commercial. D'autres valeurs de précision peuvent être utilisées, suivant les critères de chaque application: 4 bits/pixel peut suffire à plusieurs applications industrielles, 1 bit/pixel suffit pour la reconnaissance de

caractères. La nécessité de séquences d'opérations consécutives sur un même pixel nécessite une précision de traitement suffisante pour que les erreurs d'arrondis demeurent en deça d'un minimum acceptable, ne dépassant généralement pas 2^{-8} (à la sortie).

La précision du traitement dépend aussi des opérateurs et des algorithmes; variant entre 1 bit (détection d'arête, morphologie, etc) à 16 bits (transformée de Fourier, convolutions, etc). La complexité des opérations élémentaires est généralement limitée aux deux opérations arithmétiques de base: addition et soustraction et aux opérations logiques élémentaires (et, ou, ou-exclusif).

4.2c-COMPLEXITE DES OPERATIONS

La complexité des opérations varie suivant deux aspects distincts: 1-les opérations elle-mêmes et 2-les relations spatiales entre les entrées.

Pour le premier, certaines opérations utilisent intensivement des opérations arithmétiques relativement complexes. Une transformée bidimensionnelle de Fourier sur une image de 1024×1024 , par exemple, requiert 10 multiplications et 20 additions par pixel. D'autres

comportent des opérations dont la précision varie de 1 à plusieurs bits: un groupeur d'arêtes a une image d'entrée qui est de 1 bit par pixel et une image de sortie qui est asynchrone et dont le nombre de bits par objet est variable.

Pour le second, les relations spatiales sont de trois ordres: 1-homogènes, 2-hétérogènes synchrones, 3-hétérogènes asynchrones. Suivant le même ordre, nous subdivisons en trois groupes spécifiques les types d'opérations sur une image: 1-traitement primaire, 2-traitement secondaire et 3-traitement tertiaire.

Nous définissons le traitement primaire comme étant la partie du traitement d'une image traitant identiquement chaque pixel. Ce type d'opérations peut s'exécuter au moyen d'un système SIMD avec réseau de communication simple. Une grande quantité d'opérations de filtrage, de gradient, de laplacien, utilisent intensivement une opération de convolution d'une matrice de dimension restreinte (très souvent 3×3) sur l'ensemble ou une région d'une image (convolutions, opérations morphologiques, etc). La quantité d'opérations possibles à ce niveau est très grande et se fait sur tous les pixels de l'image, justifiant la spécification de ce type de traitement. C'est l'homogénéité spatiale du traitement et la simplicité des relations

interpixels qui caractérise ce type de traitement.

Les pixels spatialement (droite, gauche, haut, bas, diagonale, etc) et temporellement (images consécutives) voisins d'un pixel donné, comportent un haut degré de redondance relativement au pixel local. Les objets sont souvent de dimension plus grande qu'un pixel et aussi relativement fixes (ou peu mobiles) d'une image à la suivante. Un système de traitement primaire d'images devrait permettre de profiter de cette redondance en facilitant l'accès aux données des pixels voisins (dans l'espace et le temps) du pixel traité.

Nous définissons le traitement secondaire comme étant l'ensemble des opérations pouvant être réalisées au moyen d'un système dont les processeurs fonctionnent en mode SIMD (opérateurs identiques et synchrones) mais associant les pixels de façon hétérogène. Plusieurs transformées (Fourier, cosinus, etc) relient les pixels entre-eux par des liens réalisant un mélange parfait, tout en réalisant des opérations identiques relativement complexes sur chaque pixel (ou paire de pixels). Les pixels n'étant pas accédés de manière identique relativement à leurs voisins, les accès sont hétérogènes. Ce type de traitement peut inclure plusieurs transformées relativement exotiques (Hough,

Fourier, Hadamard, Cosinus, etc). L'homogénéité du traitement et l'hétérogénéité des accès caractérisent le traitement de type secondaire.

Le traitement tertiaire, enfin, se caractérise par l'absence d'homogénéité, autant du point de vue opérations sur les pixels, que du point de vue des accès. Un système MIMD est requis pour ce type d'opérations ou de données. Comme exemples, nous pouvons citer: groupement d'arêtes, reconnaissance de forme, reconnaissance de scène, etc. Le caractère asynchrone des opérations et des accès caractérise ce type de traitement.

La quantité de mémoire requise varie elle aussi considérablement selon les applications. Une opération de convolution $n * m$ requiert au moins $n - 1$ lignes de mémoire par opération (traitement en mode ligne). Une transformée bidimensionnelle de Fourier requiert au moins 1.5 images (traitement en mode ligne ou segment de ligne) ou trois images (traitement en mode région creuse). Un filtrage moyen sur n images nécessite $n + 1$ images (16 images d'entrée, une image moyenne) et 1 ligne de mémoire (traitement en mode ligne).

4.2d-QUANTITE D'OPERATIONS PAR PIXEL

La quantité d'opérations requises par pixel dépend des besoins des applications ou du champ d'application de chaque système. Nous estimons que pour être utile, la majorité des applications en temps réel, requerront l'application consécutive de plusieurs macro-opérations. La détermination des types, nombre, séquence et précision de chacune doit être effectuée pour chaque champ d'application. Il est, bien sur, hors du contexte de ce mémoire de fixer ce choix.

Nous pouvons toutefois estimer l'ordre de grandeur de ces besoins. Etant donné la variation de précision des données et des opérations, nous utilisons une référence binaire de détermination de la quantité d'opérations: une addition/soustraction de n bits requiert n oés (opérations élémentaires), une multiplication de $n * m$ bits comporte $n * m$ oés, une convolution par une matrice $n * m$ de précision de j bits requiert $n * m * j$ oés.

Du côté des opérations complexes, une transformée rapide bidimensionnelle de Fourier sur une image de $1024 * 1024$ pixels requiert à elle seule 10 multiplications et 20 additions de 16 bits ou $10 * 16 * 16 + 20 * 16 = 2880$ oés/pixel.

Du côté des opérations peu complexes, l'opération filtrage moyen sur deux images consécutives ne requiert qu'une addition et un cycle de décalage ou, à 9 bits de précision, 18 cycles/pixel, mais deux canaux de données d'entrée vidéo.

Les opérations asynchrones ne sont pas réalisées sur des pixels mais plutôt sur des objets dont la dimension minimale est de l'ordre d'une arête, d'un contour, etc. Le nombre d'objets impliqué est donc considérablement restreint relativement aux images des niveaux primaires et secondaires. Nous n'avons pas voulu identifier le nombre d'opérations élémentaires au niveau tertiaire, vu la très grande variété de besoins à ce niveau de traitement: fréquence et dimension des objets (arêtes, contours, trajectoires d'arêtes, etc), raffinement de l'analyse (identification d'objets, analyse de scène, analyse de séquence, analyse de relation interpersonnelle, réseaux neuroniques, etc).

Le tableau 4.1 indique un estimé du nombre de cycles de traitement requis pour compléter quelques algorithmes de traitement primaire d'images, suivant le type de processeur utilisé. Dans ce contexte, IMAGE2 (IM2) est le processeur implanté au laboratoire (section 4.4); IMAGE2+ (IM2+) est un processeur amélioré pour accélérer l'exécution de

105.

convolutions s rieelles (r alisable); IMAGE2p (IM2p) est un processeur parall le de 16 bits (possible); IMAGE2c+ (IM2c+) est un processeur parall le effectuant une convolution 3*3; une comparaison et une s lection en un seul cycle (r alisable); IMAGE2a (IM2a) est un processeur d di    l'amincissement (r alisable); IMAGE2ear (IM2ear) est un processeur d di    l'emphase d'ar te par relaxation (r alisable).

TABLEAU 4.1: NOMBRE DE CYCLES PAR OPERATION
(pixels de 8 bits)

	IM2 (oés)	IM2+ (os)	IM2p (os)	spécial (os)	
DECALAGE N BITS (DEn):	n	n	1		
SELECTION n bits (SEn):	n+2	n+2	1		
ADDITION (ADn):	n+1	n+1	1		
COMPARAISON n bits (CO _n):	n+1	n+1	1		
n IF THEN consécutifs (IT _n):	n+1	n+1	n+1		
MULTIPLICATION m*n (Mmn):	m*n + 1	m*n + 1	n		
CONVOLUTION ARITHMETIQUE 3*3 (CA3):	88	12	8	1	IM2c+
CONVOLUTION LOGIQUE 3*3 (CL3):	4	1	1	1	IM2M
FILTRAGE MOYEN 3*3: 1 CA3, 1 DE4:	92	16	9	1	IM2c+
1 AD8, 1 AD9, 2 AD11, 1 DE4:	47	n.a.	5		
FILTRAGE SELECTIF:					
4 CA3, 3 CO12, 3 SE12, DE4:	391	87	38	1	IM2c+
4AD8, 4AD9, 4AD10, 3CO12, 3SE12:	189	189	18	4	IM2c+
GRADIENT INTERPIXEL: 1 AD8:	9	9	1	1	IM2c+
GRADIENT DE SOBEL:					
4 CA3, 3 CO9, 3 SE9:	382	78	38	4	IM2c+
4 AD8, 4 AD9, 4 AD10, 3 CO9, 3 SE9:	164	164	18	4	IM2c+
AMINCISSEMENT: environ 18 IT1, 4 IT2, 14 AD10, 56 AD9, 4 DE1:	766	766	122	122	IM2c+
EMPHASE D'ARETES PAR RELAXATION: 14 CO9, 14 oés, 2 IT1, 2 AD9:	176	176	17	17	IM2c+
DILATATION: 1 CL:	4	1		1	IM2c+
SQUELETTISATION: 8 CL:	4	1		1	IM2c+

LEGENDE:

oés: opérations élémentaires binaires de type IMAGE2

os: cycles du processeur indiqué

XXn: opération XX de n bits

Un système dont les processeurs sont de type IMAGE2 requerra 1625 opérations élémentaires pour compléter un traitement primaire typique (ventilation au tableau 4.2) comportant: 1-filtrage sélectif, 2-gradient de Sobel, 3-amincissement, 4-deux itérations d'emphase d'arêtes par relaxation, 5-binarisation de l'image, 6-quatre dilatations, 7-quatre squelettisations. Dans la pratique, le nombre et les types d'opérations peuvent affecter considérablement le

nombre de cycles de traitement par pixel requis. Une squelettisation complète de la luminance d'une image NTSC (255 cycles de squelettisation) requerrait, à elle seule, 8160 cycles de type IMAGE2. Pour cette raison, nous estimons un ordre de grandeur de la complexité de 1000 à 10000 ops par pixel plutôt qu'une valeur déterminée.

TABLEAU 4.2: EXEMPLE DE SEQUENCE D'OPERATIONS PRIMAIRES
(pixels de 8 bits)

	IM2	IM2+	IM2p	IM2p+
filtrage sélectif:	189	87	18	18
gradient (matrices de Sobel):	164	78	18	18
amincissement:	766	766	122	122
2 emphase d'arêtes par relaxation:	352	176	17	17
binarisation:	10	10	1	1
4 dilatations:	16	4	16	4
4 squelettisations:	128	32	128	32
	----	----	----	----
total exemple:	1625	1153	320	212
50 convolutions arithmétiques	4600	800	450	450
255 squelettisations	8160	2040	8160	2040

L'exemple cité permet l'extraction d'arêtes en présence de bruit, traitement primaire essentiel à un système de vision artificielle. Un système de vision artificielle nécessiterait un traitement secondaire et tertiaire comprenant, par exemple, les opérations additionnelles suivantes: 1-groupement d'arêtes, 2-reconnaissance de forme, 3-analyse de scènes; ces fonctions sont réalisables au moyen d'un système de traitement tertiaire flexible ou, encore, au moyen d'un système de traitement secondaire (exemple: réseau

neuronique).

4.3: ARCHITECTURE GENERALE DU SYSTEME ENVISAGE

Nous croyons que le concept d'une puce de traitement d'images en temps réel, doit s'inscrire dans un contexte de solution globale. Cette section présente un concept d'architecture de système dont la puissance et la flexibilité peuvent répondre à l'ensemble des besoins pour un vaste champ d'applications de traitement d'images en temps réel. L'architecture retenue profite du parallélisme spatial inhérent aux images, particulièrement au traitement primaire.

Nous résolvons le problème de puissance de calcul d'abord, en partageant le traitement entre plusieurs processeurs, favorisant les structures homogènes pour faciliter l'application de HCDE (processeurs et réseau d'interconnexion réguliers), tout en tenant compte des objectifs économiques que nous nous sommes fixés.

S'il fallait cependant réaliser un seul système avec réseau d'interconnexion homogène, le degré de flexibilité requis et les bandes passantes impliquées nous obligeraient à concevoir un système MIMD (instruction multiple et donnée

multiple) avec un réseau d'interconnexion relativement flexible pour répondre aux besoins des niveaux secondaires et tertiaires de traitement. Un tel système, bien que très flexible serait, selon nous, inutilement complexe et coûteux (quantité de processeurs, complexité du réseau de communication, multiplication des circuits de commande).

Nous proposons plutôt de profiter de la subdivision en traitements primaires, secondaires et tertiaires proposée à la section précédente et d'utiliser trois systèmes autonomes pour traiter chaque type: 1-les opérations homogènes (traitement primaire) ne requièrent qu'un système SIMD avec réseau de communication local et homogène simple, 2-les macro-opérateurs nécessitant des réseaux de communication plus flexibles mais synchrones et dont les processeurs peuvent opérer en mode SIMD du traitement secondaire et 3-les macro-opérateurs opérant sur des données asynchrones, requérant alors un système MIMD.

Cette séparation en trois blocs de complexité complémentaires simplifie significativement le circuit de chaque niveau. Le premier est avantage par la simplicité du réseau de communication (local et homogène) et l'absence de multiplication du circuit de commande complexe au niveau processeur (commande partagé). Le(s) système(s) de

traitement secondaire est(sont) avantage(s) par l'absence de multiplication de circuit de commande complexe (commande partagé) et par une réduction du nombre de liens inter-processeurs à réaliser. Ces liens inter-processeurs étant plus complexes au niveau secondaire, plusieurs avantages sont obtenus en partageant les traitements primaire et secondaire: 1-réduction du nombre de liens, 2-réduction des distances à couvrir et 3-évitemment de la complexité au niveau primaire. Le système de traitement tertiaire bénéficie des mêmes avantages que le traitement secondaire sauf pour la simplicité de l'unité de commande qui doit être présente dans chaque processeur (à chaque site).

L'économie réalisée est appréciable puisque le traitement primaire se fait sur une quantité d'objets (pixels) considérablement plus grande que le traitement tertiaire. L'économie en circuit de commande spécifique à chaque processeur obtenue par un fonctionnement en mode SIMD pour les traitements primaire et secondaire constitue un avantage significatif à ces niveaux. L'économie consécutive à l'utilisation d'un réseau de communication simple (maille locale bidimensionnelle), requérant peu de superficie et facile à commander simplifie l'implantation et maximise donc l'avantage de rendement de HCDE au premier niveau.

La superficie allouée au traitement primaire sera ainsi significativement inférieure à ce qu'elle aurait été si un système intégralement homogène, requérant un réseau de communication complexe et une répétition inutile du circuit de commande (un circuit de commande par processeur) avait été utilisé. La nature hautement homogène des algorithmes associés au traitement primaire conduit à des systèmes performants et relativement compacts. Un seul, ou une séquence de systèmes SIMD ne comportant qu'un réseau d'interconnexion local courte distance, n'associant à chaque processeur que ses voisins immédiats, suffit à ce niveau à faible granularité (niveau pixel ou pyramidal). Ces réseaux "en maille" sont peu encombrants et réguliers.

Le traitement secondaire (2) peut être réalisé avec un seul ou une série de systèmes plus ou moins dédiés mais munis des réseaux de communication plus flexibles et plus complexes (communications à grande distance requises). Reprenant l'exemple d'une transformée rapide de Fourier pour un système de niveau secondaire. Des processeurs vectoriels de type "papillon, taille 2" effectueraient la multiplication et les deux additions requises, par paire de pixel, en un seul cycle, réduisant considérablement le nombre de processeurs, simplifiant ainsi significativement

le réseau de communication "mélange parfait" requis pour ce type de traitement. Le réseau de communication pourrait être dédié (mélange parfait: "perfect shuffle") ou plus flexible (ex: entièrement connecté, avec adressage local indirect) pour élargir le champ d'application. Les possibilités sont vastes à ce niveau.

Le traitement tertiaire ne requiert pas nécessairement un réseau de communication aussi complexe que celui envisagé au niveau secondaire. A ce troisième niveau, nous envisageons, un système homogène MIMD avec interconnexions asynchrones. Tel que mentionné à la section précédente, nous supposons qu'à ce niveau, les données sont dégrossies et que bien que le nombre d'opérations par objet puisse être élevé, la restriction sur la quantité d'objets à traiter restreindra les besoins à ce niveau. Bien que plus flexible, ce système ne serait donc pas nécessairement aussi complexe que les deux précédents à cause de la réduction du nombre d'objets.

Chaque niveau peut être subdivisé plus finement pour répondre à des applications plus ou moins spécifiques. Le nombre et la complexité de chaque système SIMD/MIMD dépend, bien sûr, des besoins de chaque application.

Nous proposons de relier ces systèmes SIMD et MIMD par des liens synchrones pour minimiser la superficie des interfaces de communication, particulièrement aux deux premiers niveaux. Une interface tampon est donc nécessaire pour extraire les données hétérogènes à l'entrée du système du niveau tertiaire. Nous considérons qu'une communication inter-système de type pipeline est bien adaptée au traitement d'images en temps réel. Le nombre de liens par système dépend, lui aussi, des approches choisies. La figure 4.1 montre le diagramme bloc d'un système hypothétique dont le réseau de communication est dédié et de nature essentiellement en arbre. Nous croyons que ce type de système est bien adapté à la vision artificielle où l'ampleur du marché et des besoins justifient le développement d'un système, au moins partiellement dédié.

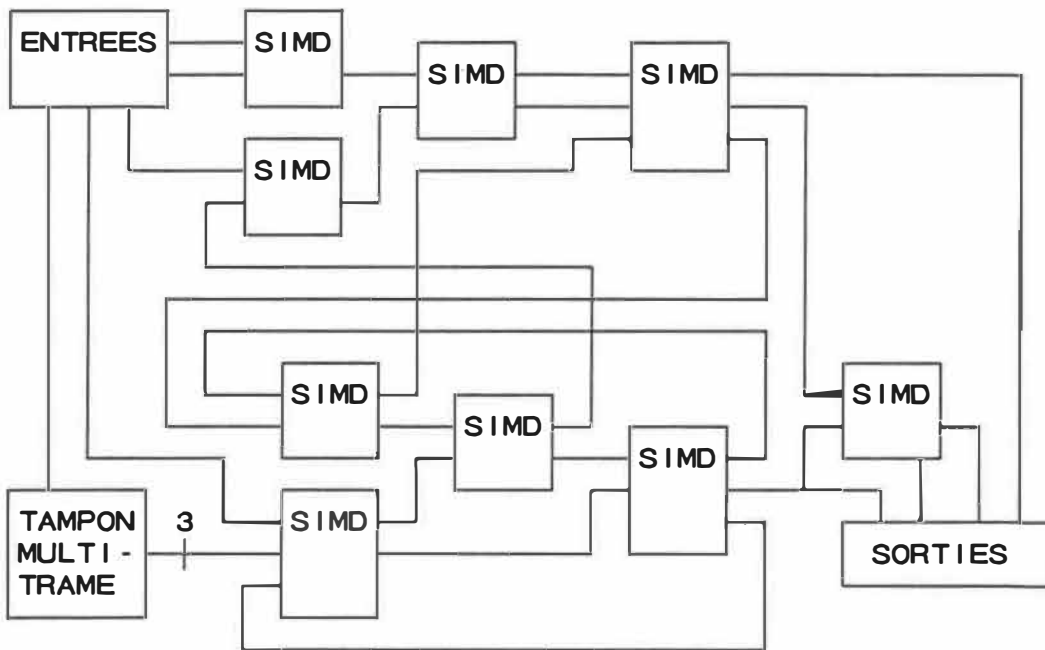


FIGURE 4.1: EXEMPLE DE SYSTEME POUR LE TRAITEMENT D'IMAGES EN TEMPS REEL (réseau d'interconnexion dédié).

La figure 4.2 montre un exemple de système de traitement d'images en temps réel mais dont le réseau d'interconnexion inter-système est programmable (plus flexible), mieux adapté à un champ d'application plus vaste.

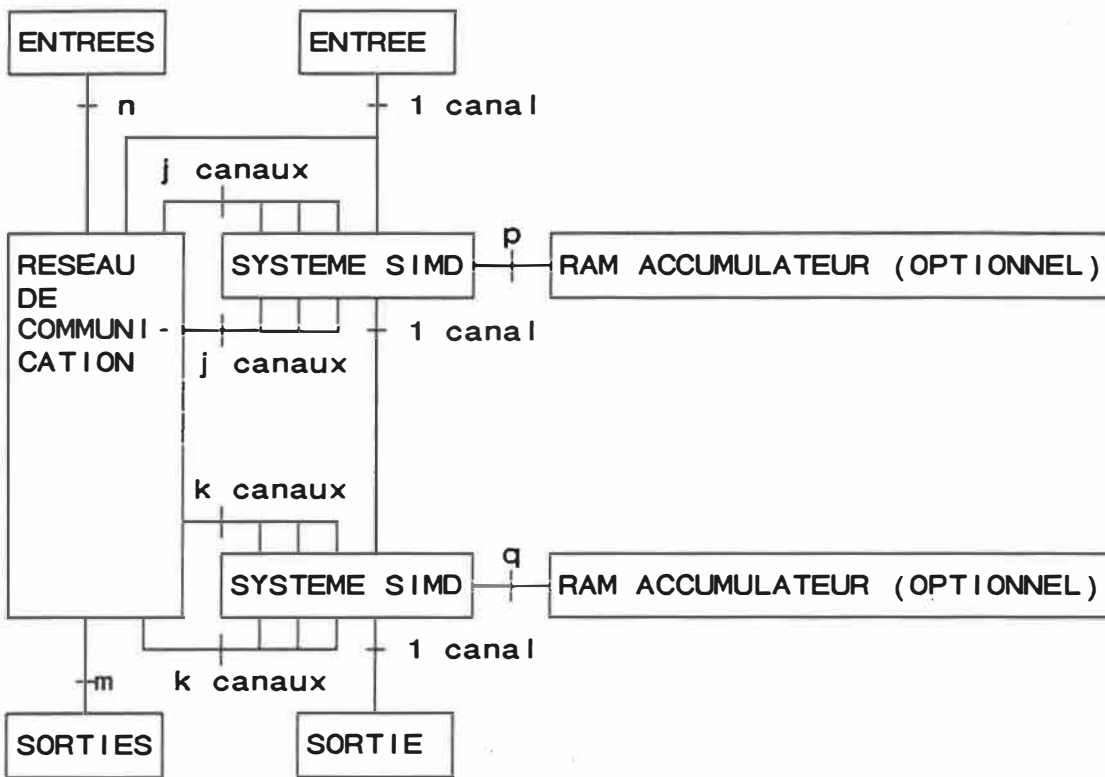


FIGURE 4.2: SYSTEME POUR LE TRAITEMENT D'IMAGES EN TEMPS REEL
(réseau d'interconnexion programmable)

Le nombre et le type des systèmes SIMD par système de traitement d'images dépend de chaque champ d'application. Nous sommes convaincus que le concept d'architecture proposé est suffisamment flexible et performant pour répondre à une vaste gamme d'applications en temps réel. A titre estimatif, reprenons notre exemple de la section 4.2d, pour 2000 (10000) oés IM2 de traitement primaire par pixel. Considérant une technologie de 1 micron (cycles de 30 nsec), un système pouvant traiter 2000 (10000) oés par pixels ne requerra que $2000 \cdot 30 / 68 = 884(4425)$ PEs. Si nous envisageons

des systèmes SIMDS de 1024 PEs, nous requerrons 1(5) systèmes suivant les besoins. Considérant toujours une technologie de 1 micron et les mesures obtenues au laboratoire (tableau 5.3), un PE occuperait alors 86400 microns²; un(cinq) système(s) de 1024(5120) PEs occuperaient environ 88(440)mm². Considérant trois niveaux hiérarchiques CDE reconfigurables, nous ajoutons respectivement 6%, 6% et 40% à la superficie de chaque niveau. Nous obtenons une superficie de $440 \times 1.06 \times 1.06 \times 1.4 = 692$ =environ 7cm² par système de traitement primaire de 10000 oés par pixel. Il est donc envisageable de couvrir 10000 oés par pixel sur une seule puce IMAGE2 d'environ 7cm².

Un système de traitement primaire en temps réel de 10000 oés par pixel est donc réalisable sur une seule puce de 7cm² comprenant $800 \times 1024 \times 1.06 \times 1.06 \times 1.4 = 1288000$ transistors. Si nous considérons un coût de \$5.00 par cm², un rendement de 78% (tableau 5.6), un système de traitement primaire pour 1000 oés par pixel occupe une seule puce et coûte \$45.00 de silicium auquel nous ajoutons \$30.00 d'emballage (88 broches) pour un total de \$75.00 par système. A ce système processeur, nous ajoutons une mémoire accumulateur de 8 mégaoctets pour 18 trames (opération moyenne multitrame sur 16 images couleur consécutives) et constatons que le coût du système n'est pas dominé par le

processeur mais par la mémoire: 64 boîtiers de 1 Mbits \$4000.00 (janvier 89). Même si nous considérons un traitement secondaire et tertiaire 10 fois plus couteux que le traitement primaire, le coût total d'un système demeure $\$4000 + \$75 + \$750 = 4825$ (environ \$5000); ce coût est dominé à 80% par la mémoire principale.

La figure 4.3 montre un diagramme bloc d'un système SIMD typique. Chaque système SIMD et MIMD comprend six éléments principaux: 1-une matrice de processeurs, 2-un réseau d'interconnexion reliant ces processeurs, 3-un circuit de commande commun, 4-une mémoire tampon/accumulateur (selon les besoins), 5-une interface aux autres systèmes et 6-une interface lente au système hôte pour la programmation du circuit de commande. Les trois premiers éléments sont spécifiques à chaque système, les interfaces entre les systèmes seraient idéalement au moins partiellement normalisées pour faciliter l'interconnexion entre les systèmes et la spécification de systèmes.

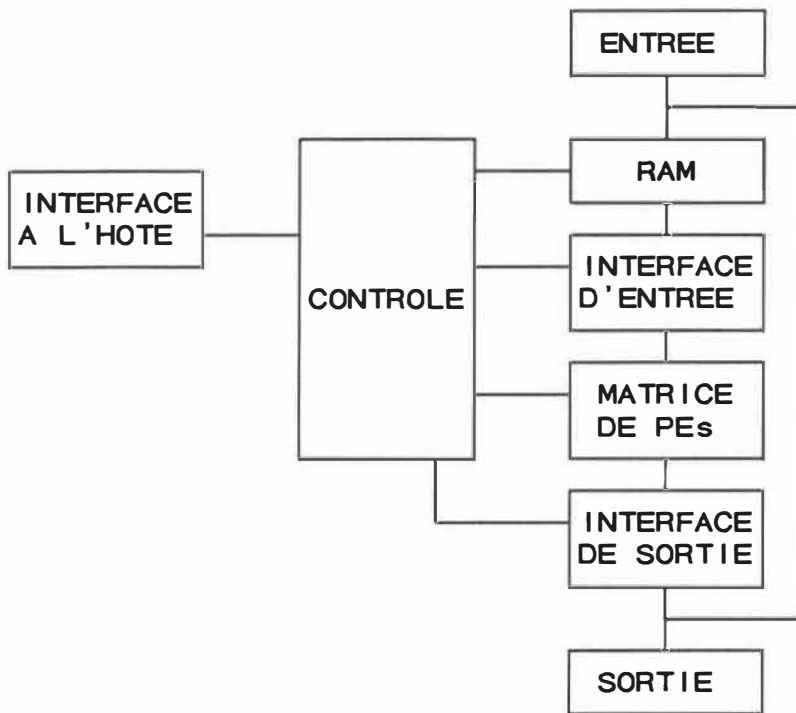


FIGURE 4.3: EXEMPLE DE SYSTEME SIMD AVEC RAM
(canal de données seulement)

La figure 4.4 montre un exemple typique de matrice de processeurs. Deux éléments principaux caractériseront les systèmes de chaque niveau: le réseau d'interconnexion local (reliant les processeurs) et les processeurs. La section 4.4 et d'une façon complémentaire le rapport du projet IMAGE2 [26] couvrent plus en détail ces deux aspects pour la puce IMAGE2.

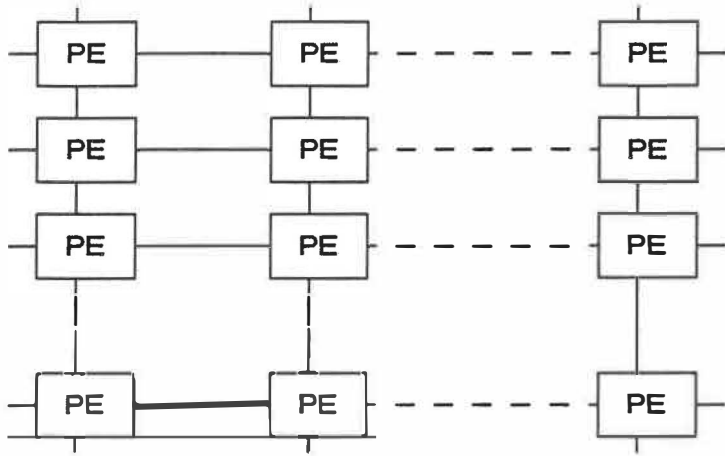


FIGURE 4.4: EXEMPLE DE MATRICE DE PEs EN REGION

Les bandes passantes des signaux vidéo sont compatibles avec les délais de propagation au travers des broches d'un circuit intégré fabriqué à l'aide d'une technologie représentative de l'état de l'art. Si nous envisageons une précision de 8 bits par échantillon entre les systèmes, chaque système ne requiert que 8 broches par interface inter-système ou par accès externe (autre système ou accès à la mémoire principale du système).

Nous considérons qu'il est important que le nombre d'accès à la mémoire principale soit restreint pour minimiser le nombre de ports d'entrée/sortie des systèmes et le coût de la mémoire (dominante économique). Chaque accès à la mémoire principale augmente la bande passante des éléments impliqués, compliquant inutilement le réseau d'interconnexion inter-système (la mémoire principale étant

elle-même considérée comme un système). Conséquemment, chaque système doit être muni des tampons d'entrée/sortie nécessaires pour restreindre la bande passante du réseau pipeline inter-SIMD.

Nous avons envisagé trois modes de traitement possibles: 1-région pleine, 2-région creuse, 3-ligne. Dans un mode de traitement en régions pleines le système SIMD est une matrice de processeurs dont chaque processeur est assigné à un pixel et ou des processeurs voisins sont associés à des pixels voisins immédiats. Une image est traitée en effectuant le traitement d'une région, réassignant le système de région en région jusqu'à ce que toutes les régions de l'image soient traitées. Dans un mode de traitement en régions creuses, chaque processeur est assigné à une région, un processeur voisin est associé à une région voisine. Le traitement d'une image est obtenu en traitant simultanément toutes les régions, un pixel de chacune à la fois. Dans un mode de traitement ligne, les processeurs voisins sont associés à des pixels voisins mais sur une même ligne. Dans ce cas-ci, le traitement d'une image consiste à traiter successivement chaque ligne.

Un débordement de frontière est un ensemble des données d'entrée correspondant à des pixels auxquels un

processeur n'est pas immédiatement assigné; et néanmoins nécessaires au traitement des pixels auxquels un processeur est assigné. Prenons l'exemple d'une fonction appliquant un opérateur de type convolution 3×3 (courant) aux données d'entrée. Dans ce cas chaque processeur assigné à un pixel doit accéder des données se trouvant dans des pixels voisins. Dans le cas d'un système fonctionnant en mode région, le débordement correspond au périmètre extérieur du système ($2m + 2n + 4$ pour un système de $m \times n$ processeurs). Dans le cas d'un système en région creuse, le débordement correspond au périmètre (huit pixels) encerclant chaque pixel/processeur ($8 \times m \times n$ dans un système de $m \times n$ processeurs). Dans le cas d'un système ligne, le débordement correspond aux lignes supérieures et inférieures à la ligne traitée ($2n$ pour un système de n processeurs).

Les débordements de frontières s'accumulent pour chaque opération récursive et à débordement. L'exemple de séquence de la section 4.2d, comporte 13 débordements de frontière récursifs. Si cet exemple devait être réalisé au moyen d'un système en région pleine, chaque débordement conduirait au sacrifice des résultats des PEs périmétriques. Si un système région pleine est utilisé et qu'une puissance de 32×32 résultats est désirée, le système doit être augmenté d'une bande périmétrique entourant la région

résultante de 13 PEs de largeur. Un système de 58×58 PEs est requis pour produire 32×32 pixels résultants par cycle (surcharge de 328%). Dans ce cas-ci un système global, plutôt que plusieurs petits systèmes, sera plus avantageux. Considérons un système pour 5000 pixels résultants (71×71 PEs). Dans ce cas-ci le système requis sera de 97×97 PEs (surcharge de 87%). Pour l'exemple de la section 4.2d, un système de traitement primaire en régions pleines à mémoire restreinte doit être augmenté de 87% pour compenser les effets du débordement. Si une mémoire suffisante par processeur permet de compléter le traitement de toute une image sans accéder à la mémoire principale, Les opérations peuvent être complétées avant de procéder à une opération récursive subséquente. Dans ce dernier cas le débordement est sans conséquence autre que la nécessité de présenter les données de débordement aux entrées périmétriques du système. Ce mode de traitement requiert une mémoire accumulateur d'une image complète (traitement intermédiaire, +une image d'entrée, +une image de sortie pour un total de trois images par système.

Dans le cas d'un système en régions creuses, il est implicite que chaque processeur doit accéder régulièrement aux données relatives à sa région. Ce système n'est donc possible que si chaque processeur a une mémoire suffisante

pour toutes les données d'entrée/sortie pour la région à laquelle il est assigné. Les besoins en mémoire sont donc d'au moins trois images par système, distribuée à chaque processeur. Le mode de traitement en régions creuses a l'avantage de restreindre la bande passante inter-processeur.

Dans le cas de traitement de type rangée, chaque processeur doit être muni d'un tampon d'entrée/accumulateur de un pixel par rangée de débordement. Dans l'exemple de la section 4.2d, 13 opérations à 2 rangées de débordement nécessite 26 mots de mémoire par PE. Ceci est considérablement moins coûteux que d'augmenter indument le nombre de PEs comme dans le cas de systèmes en régions pleines et certainement moins couteux en mémoire que dans le cas du traitement en régions creuses. Un signal NTSC est transmi/reçu ligne par ligne, l'utilisation d'un mode de traitement de type ligne a l'avantage d'harmoniser le traitement avec le flux de données d'entrée/sortie d'un système. Ce mode nécessite donc moins de mémoire tampon aux entrées.

Les trois modes de traitement sont valides. Certaines présentent des avantages suivant la nature des sigaux, des processeurs et des opérations à réaliser. A notre avis, la

présentation des signaux dans le cas d'images NTSC et le débordement aux frontières de la majorité des opérations encouragent un traitement en mode ligne sur un traitement en mode région. Lorsque l'avancement technologique permettra d'associer de grandes quantités de mémoire aux éléments processeurs le choix du mode de traitement sera moins significatif.

4.4: PARTIE FONCTIONNELLE D'IMAGE2

L'envergure du système décrit à la section précédente était irréalisable avec les ressources humaines, matérielles et technologiques disponibles lors de la réalisation de ce mémoire. Nous avons donc plutôt cherché à réaliser une partie du système. Il était naturel que notre premier objectif vise le traitement primaire et soit donc de type SIMD. Nous croyons en effet que la majorité des applications requerront ce niveau de traitement. Un réseau de communication plus simple, restreignait la complexité du design. L'envergure de l'analyse des besoins pour un champ limité restreignait le travail à un domaine connu et stable. En effet, le champ des algorithmes aux niveaux secondaires et tertiaires étant encore en développement, il devenait inutile pour nous de couvrir immédiatement ces niveaux

autrement qu'au niveau conceptuel comme nous l'avons déjà fait aux sections précédentes. Il n'était ainsi pas nécessaire d'étudier en détail le vaste champ des applications secondaires et tertiaires. La superficie de silicium dont nous disposions ($.25 \text{ cm}^2$) était insuffisante pour réaliser une matrice de processeurs complexes. L'existence de processeurs plus complexes (RISC, DSP, processeurs vectoriels, etc) sur le marché limitait l'intérêt de l'utilisation de ces types de circuits pour notre démonstration.

Attaquant donc d'abord le traitement primaire (puissant, calculs simples, communications homogènes et peu étalées), nous avons envisagé plusieurs types de processeurs élémentaires plus ou moins dédiés. Nous avons retenu le plus flexible (IMAGE2), comme système universel pour le traitement primaire. Le processeur d'IMAGE2 est simple (800 transistors) mais, sa flexibilité étant élevée, il peut traiter une grande variété d'algorithmes primaires.

IMAGE2 est un segment extensible de 8 PEs fonctionnels ($n=8$). La figure 4.5 montre le diagramme bloc général (segment de 8 PEs fonctionnels), comme il se présente aux broches de la puce IMAGE2.

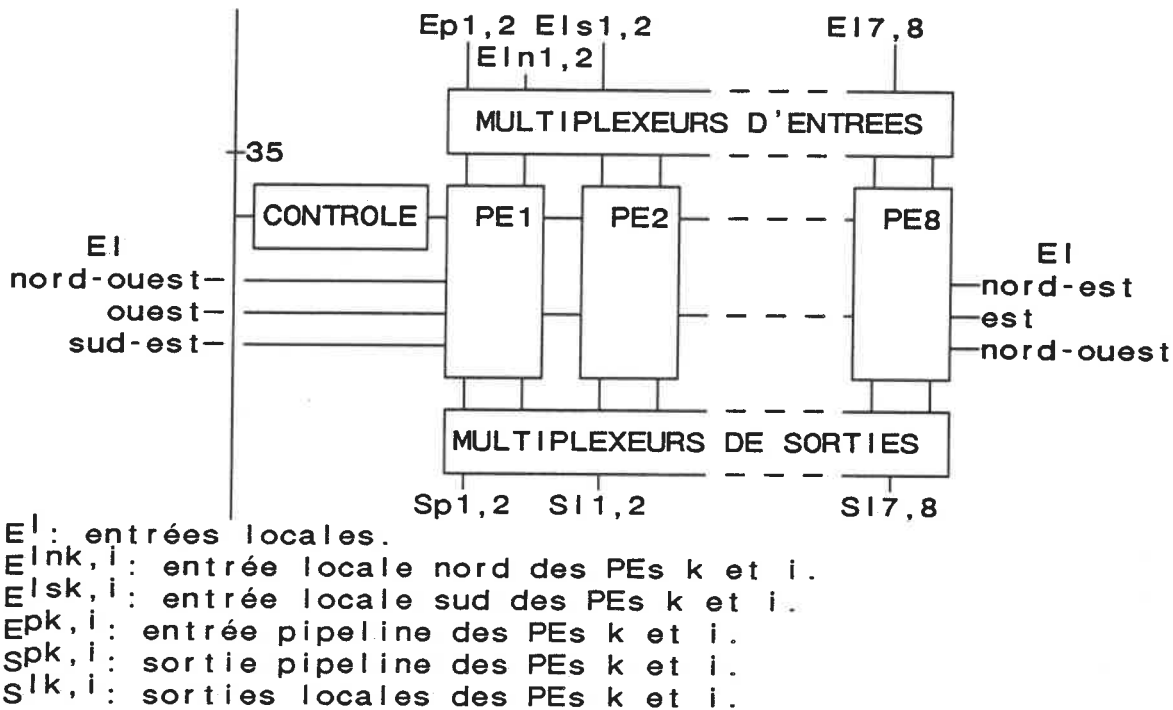


FIGURE 4.5: SEGMENT DE 8 PES (IMAGE2).

Pour compenser la faible densité de processeurs par puce (10), IMAGE2 a été conçu de façon à être extensible. Le PE (processeur élémentaire) fonctionne suivant une arithmétique sérielle mais son unité arithmétique est universellement programmable. Le réseau d'interconnexion entre les PEs est homogène et à courte distance. A l'interface pipeline décrit à la section précédente (figure 4.6), le PE d'IMAGE2 ajoute une interface de communication locale permettant l'accès à chacun de ses huit voisins immédiats: nord, sud, est, ouest, nord-ouest, nord-est, sud-ouest et sud-est (figure 4.7). Bien que le PE n'ait pas d'accès direct aux PEs dont la distance est plus grande que

1, ce réseau de communication est suffisant pour une réalisation efficace d'algorithmes convolutifs; que le système fonctionne en mode région pleine (un PE par pixel d'une région rectangulaire), ou ligne (un PE par colonne ou groupe de colonnes), ou région creuse (une région par PE).

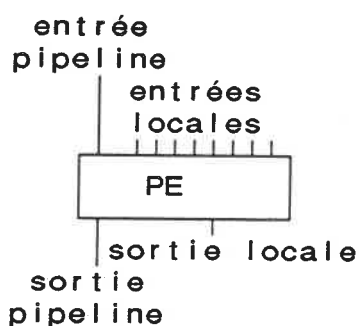
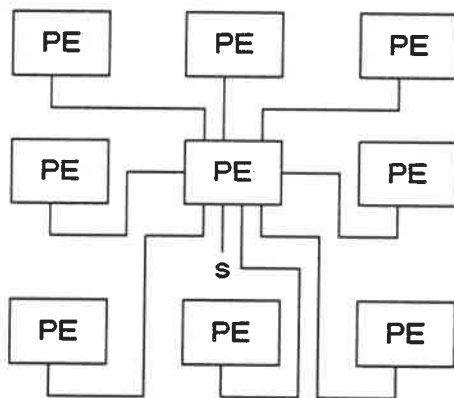


FIGURE 4.6: ENTREES SORTIES D'UN PE



s: sortie locale du PE aux voisins immédiats
PE: processeur élémentaire

FIGURE 4.7: BRANCHEMENT DE RESEAU LOCAL D'UN PE d'IMAGE2

La figure 4.8 montre un diagramme bloc simplifié d'un PE d'IMAGE2 sans circuit HCDE. Pour obtenir un maximum de flexibilité, l'unité de traitement est un ULU (unité logique

universelle) à 4 entrées. La table de vérité de l'ULU est programmable à chaque cycle de commande, via 16 lignes de commande provenant du circuit de commande commun. Il s'agit essentiellement d'un multiplexeur à 16 entrées dont la sortie est commandée par quatre bits provenant du circuit de communication interne du PE (registres internes). Une cellule additionnelle procure simultanément une sortie de retenue pour accélérer les opérations addition et soustraction (un bit par cycle). Ce niveau de flexibilité confère au PE la capacité d'exécuter avec des degrés d'efficacité comparables, les opérations autant logiques (morphologie), qu'arithmétiques. L'approche sérielle permet de varier, en temps réel, la précision des opérations arithmétiques à chaque cycle, via les lignes de commande.

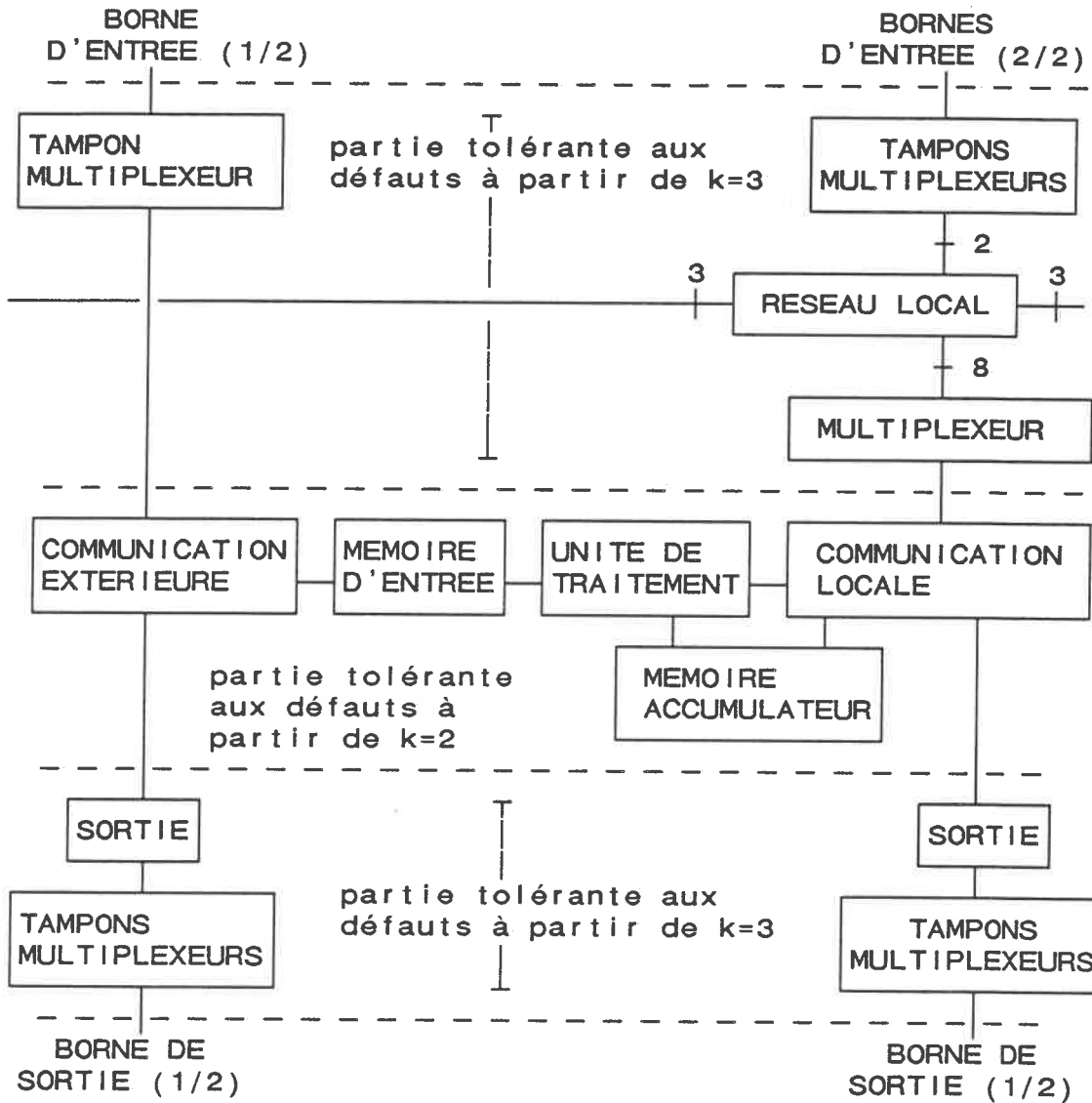


FIGURE 4.8: DIAGRAMME BLOC GENERAL DU PE D'IMAGE2

L'opération est centrée autour de l'ULU. Les accès à la mémoire et aux entrées/sorties ne ralentissent pas l'opération du ULU. Chacune des entrées de l'ULU peut accéder une sortie d'un des blocs de registre interne. Simultanément, la sortie de l'opération précédente peut être stockée pendant l'opération suivante (5 accès simultanés

associés à l'ULU). Simultanément aussi, une donnée provenant d'un segment en amont peut être stockée et une donnée peut être transmise à un segment aval. Le PE d'IMAGE2 peut effectuer simultanément 5 lectures, 2 écritures, une opération logique universelle et une retenue pour un total de 9 activités simultanées.

Suite à une étude plus précise de certains algorithmes, Daniel Audet [27] a proposé un circuit conditionnel permettant l'opération "if then else". Une proportion importante de programmes contenant ce type d'opération justifiait cette fonctionnalité. Chaque PE est muni d'un circuit logique conditionnel simple permettant le traitement de boucles "if then else" éliminant ainsi à peu de frais (ajoute 1 transistor par bit de mémoire et un circuit de 12 transistors) cette limite à la flexibilité du PE.

Les tampons multiplexeurs montrés à la figure 4.8 visaient à réduire d'un facteur de 2 le nombre de broches des signaux séparant les puces. Les PEs fonctionnent par paires, les signaux homologues d'entrées/sorties verticaux (nord, sud et pipeline) de chaque PE d'une paire passent par des broches communes. Un délai de 2 cycles est prévu pour le transfert concurrent des données entre les PEs.

Ainsi, le délai de propagation des broches entre les puces ne ralentissent pas l'horloge du système pour des systèmes multipuces (transferts de données concurrent avec le traitement). De plus amples détails sur le fonctionnement du PE sont fournis dans le rapport du projet IMAGE2 [26].

4.5-EXPANSION EN SYSTEMES PLUS IMPORTANTS

La puce IMAGE2 est extensible, autant au niveau du réseau local que du réseau pipeline. Un système de 32 x 32 PEs, réalisé au moyen d'IMAGE2 comporterait quatre puces par segment de 32 PEs (figure 4.9) et 32 groupes de 4 puces pour former un une matrice de 32 * 32 PEs (figure 4.10).

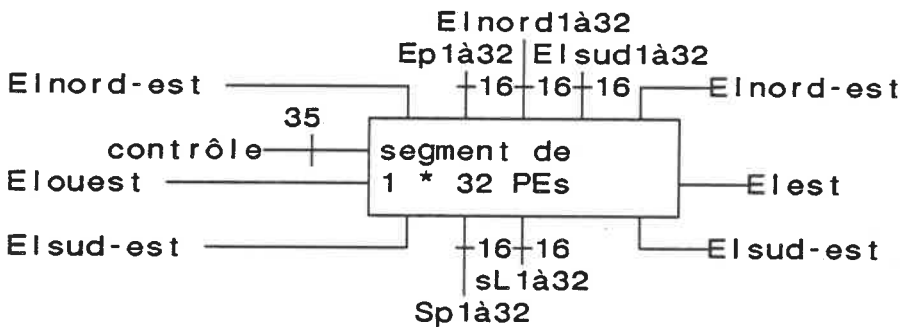


FIGURE 4.9: SEGMENT DE 32 PES (4 PUCES IMAGE2)

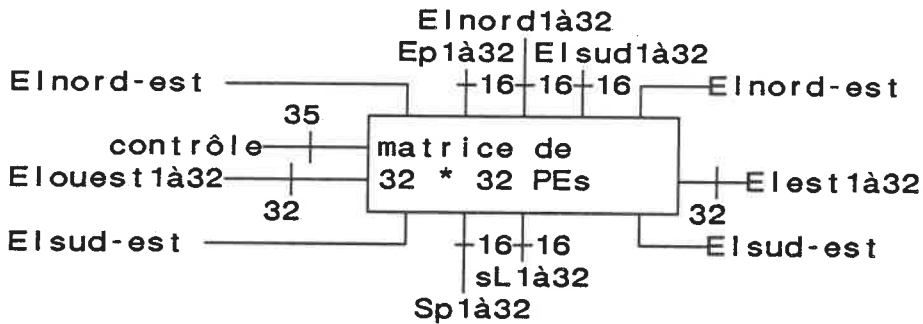


FIGURE 4.10: MATRICE DE 32 * 32 PES ($32*4 = 128$ PUCES IMAGE2)

Pour ne pas avoir à multiplier inutilement le nombre de broches de mémoire, nous proposons de relier la mémoire principale (1 à 16 mégaoctets) du système via le canal pipeline plutôt que le canal local. La bande passante des broches du système étant de 50 nsec, quatre broches de mémoire par paire entrée-sortie sont requise pour des broches d'accès à la mémoire dont le délai de propagation est de 100 nsec. Nous envisageons donc environ 64 broches de mémoire (100nsec). Si nous recherchons une mémoire d'environ 8 mégaoctets, la mémoire sera de 1 mégabit par broche. Notre mémoire se présentera donc en multiples de 1 moctets. Ceci est intéressant car cette granularité de mémoire est au début de sa courbe économique (1 mbits est moins cher que $4 * 256$ kbits).

Pour réduire le coût encore plus, nous proposons d'éliminer l'expansibilité en mode région pour des systèmes

monoblocs de grande importance (exemple: 1024 processeurs). La puissance de calcul peut quand-même être augmentée en cascasant les systèmes via le canal de communication pipeline. Cette décision permet d'éliminer les broches associées au canal de communication local dans le cas de systèmes réalisés sur une seule puce: économie de $31 \cdot 4 + 8 = 132$ entrées et $31 \cdot 4 = 124$ sorties pour un système de 1024 PEs. En maintenant un multiplexage aux bornes d'entrée/sortie, l'économie est de 128 broches par puce.

Pour restreindre le nombre de broches liant les segments au troisième niveau hiérarchique, nous proposons de renverser de 90 degrés les segments de PEs, reliant la sortie pipeline du PE local à l'entrée du PE voisin plutôt qu'à celle du PE du segment voisin. Ceci réduit de $32 + 32 = 64$ à $1 + 1 = 2$ le nombre de noeuds pipeline à relier entre les segments, réduisant ainsi d'un facteur de 32 le nombre d'interrupteurs de reconfiguration des canaux de communication pipeline au troisième niveau hiérarchique CDE (matrice).

Le nombre élevé de PEs dans la chaîne d'entrée/sortie pipeline pose une restriction à la bande passante aux entrées/sorties du système. Pour éviter que l'interface pipeline à la mémoire principale ne devienne un facteur

limitatif à la puissance du système, nous recommandons d'augmenter la mémoire de chaque PE de manière à permettre de concaténer plusieurs opérations consécutives. Le mode de traitement ligne présente un avantage du point de vue de la mémoire interne et des bandes passantes à la mémoire principale. Nous proposons d'utiliser intensivement ce mode de traitement, du moins jusqu'à ce que la mémoire de chaque processeur permette un mode d'opération présentant des avantages complémentaires plus importants. Il serait autrement nécessaire de distribuer la mémoire principale sur un plus grand nombre de broches, ce qui conduirait à une augmentation du coût non seulement du système mais aussi et de manière non négligeable, sur celui de la mémoire. Nous évitons le gaspillage de puissance de calcul, ou de bande passante excessive, obtenus lorsqu'un mode de traitement de type région est utilisé (section 4.3).

Dans IMAGE2, la superficie relative allouée à l'unité de traitement est 15% de celle du PE. L'augmentation de mémoire locale affectera encore plus ce facteur. Nous considérons l'unité de traitement comme le coeur (la raison d'être) d'un processeur, le reste du circuit constituant une charge complémentaire malheureusement nécessaire. Nous proposons d'améliorer la puissance du système en augmentant la superficie relative allouée à l'unité de traitement,

tout en respectant la vocation de traitement primaire, plutôt que de bêtement multiplier le nombre de PEs (IMAGE2). Nous proposons donc d'associer à l'unité arithmétique une(quelques) unité(s) de calcul complémentaire(s).

Nous avons considéré quelques solutions d'unités de traitement complémentaire: 1-un convolveur arithmétique sériel, 2-un convolveur logique, 3-un unité arithmétique parallèle, 4-un unité convolveur, comparateur et sélecteur parallèle. Le convolveur arithmétique sériel permet d'augmenter la vitesse des convolutions sans modifier le fonctionnement de la mémoire actuelle (opération sérielle vectorielle sur 9 pixels simultanément). Une convolution 3×3 en 13 cycles plutôt que $8 + 9 + 10 \times 3 + 11 \times 4 = 91$ dans le cas de l'ULU d'IMAGE2, augmentant alors d'un facteur de 7, à 8 fois la puissance de calcul, sans changer la vitesse des horloges, pour ce type d'opérations. Le convolveur logique permet d'effectuer une convolution logique sur 9 pixels en un seul cycle plutôt que 4 (opérations morphologiques). A titre d'exemple, sur notre recommandation, Claude Thibault [28] a réalisé une puce spécialisée au traitement morphologique. Un unité arithmétique parallèle permet d'accélérer autant la vitesse des convolutions que des comparaisons. Un convolveur-comparateur-sélecteur parallèle permet de réaliser un gradient de matrices de Sobel

en 4 cycles, mais augmente significativement la superficie d'un PE.

Le tableau 4.3 montre une étude comparative de quatre cas de PEs combinant ces unités arithmétiques au PE IMAGE2 (implanté, section 4.3) avec une mémoire d'entrée améliorée comprenant 256 bits plutôt que 48 bits (32 mots). L'augmentation de mémoire est pour restreindre la bande passante à la mémoire principale en mode de traitement ligne. IMAGE2+ est un PE de type IMAGE2 avec mémoire augmentée et 3 unités de traitement: 1-convolutionnel sériel, 2-convolutionnel logique et 3-ULU. IMAGE2p est un PE à mémoire d'entrée augmentée et unité arithmétique parallèle et unité ULU. IMAGE2p+ est un PE avec mémoire d'entrée augmentée et trois unités arithmétique: 1-unité arithmétique parallèle, 2-unité ULU et 3-unité convolutionnelle logique.

TABLEAU 4.3: EXEMPLE DE SEQUENCE D'OPERATIONS PRIMAIRES
(pixels de 8 bits)

	IM2	IM2+	IM2p	IM2p+
superficie	1.3	1.6	3.7	4.0
filtrage sélectif:	189	87	18	18
gradient (matrices de Sobel):	164	78	18	18
amincissement:	766	766	122	122
2 emphase d'arêtes par relaxation:	352	352	34	34
binarisation:	10	10	1	1
4 dilatations:	16	4	16	4
4 squelettisations:	128	32	128	32
	----	----	----	----
total exemple:	1625	1329	337	229
densité de puissance de calcul	1	1	1.6	1.9
50 convolutions arithmétiques	4600	800	450	450
densité de puissance de calcul	1	4.7	3.6	3.3
255 squelettisations	8160	2040	8160	2040
densité de puissance de calcul	1	3.2	.35	1.3

La densité de puissance de calcul est une mesure de performance d'un système correspondant au rapport de la puissance de calcul par unité de superficie de silicium occupée. Le tableau 4.3 montre clairement que le PE le plus performant en termes de densité de puissance de calcul dans le cas de l'exemple 4.2d est un PE de type IMAGE2p+ comprenant: une mémoire d'entrée de 32 mots de 8 bits, un unité arithmétique parallèle de 16 bits, un unité convolveur logique (un cycle), une mémoire accumulateur triple accès de 4 mots de 16 bits. Ce système présente une densité de puissance de calcul de 1.9 fois celle de l'unité IMAGE2 avec mémoire augmentée dans le cas de l'exemple pondéré de la section 4.2d. Dans le cas d'applications riches en opérations morphologiques et convolutionnelles arithmétiques, un PE de type IMAGE2+ sera plus performant

(densité de puissance au dessus de 3). Considérant des opérations générales et en préparation au traitement secondaire (PEs de type multiplieur-accumulateur, papillon, etc), le type IMAGE2p+ nous semble le meilleur choix.

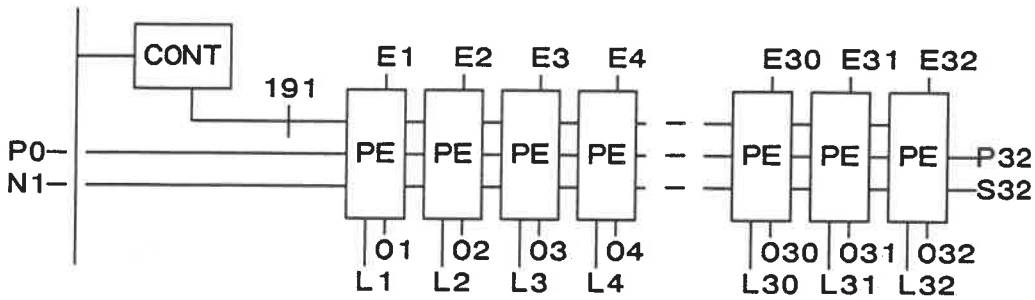
L'augmentation de la puissance de calcul permet aussi de mieux balancer la bande passante relative des opérations hautement convolutives avec celle d'opérations plus simples, permettant ainsi une utilisation plus exhaustive de chaque partie du système et restreignant le coût de la mémoire principale (bande passante restreinte). Considérant un traitement en mode ligne, une mémoire d'entrée de $13(16) * 2 * 8 = 208(256)$ bits suffit à restreindre la bande passante d'un facteur de $1625/4 = 406$.

Considérant deux lignes d'image par opération (2 mots par PE en mode ligne) et une précision de 8 bits, une mémoire d'entrée de $16 * 8 * 2 = 256$ bits garantit que la bande passante du canal ne sera pas un facteur limitatif de la puissance de calcul. Une analyse plus approfondie permettrait d'identifier plus précisément le besoin pour une gamme plus large de séquences. Une mémoire de 256 bits permettrait de concaténer 128 convolutions logiques consécutives desservant ainsi efficacement les besoins dans

le cas de traitement morphologique.

Puisque le bloc de mémoire d'entrée est d'usage structuré (séquentiel) et ne requiert pas de stockage conditionnel, une cellule de 3 transistors par bit suffit. Une analyse sommaire permet d'envisager une cellule de $50 \times 3 = 150$ carrés par bit plutôt que $60 \times 4 = 240$ (actuel). L'augmentation de la mémoire augmenterait d'un facteur de $256/48 \times 150/240 = 3.3$ la superficie de la mémoire d'entrée actuelle, ajoutant 22% à la superficie du PE.

Les deux unités convolutionnels (arithmétique et logique) ajoutent $28 \times 9 + 8 \times 9 + 8 \times 16 + 8 \times 12 + 18 \times 9 + 12 \times 4 = 758$ transistors et 18 lignes de contrôle (partagées) additionnels. En conservant l'ULU existant, la flexibilité est maintenue. Le gain en puissance de calcul n'affecte pas la vitesse de l'horloge. La réduction du nombre de broches de mémoire requises de 1024 à 32 constitue l'économie la plus importante, surtout si on tient compte de l'augmentations de la vitesses des horloges prévisible lorsque la granularité descendra à 1 um ou .5 um par carré. Une puce SIMD comportant 32×32 PEs améliorés, réalisée avec une technologie de 1 um se présenterait comme aux figures 4.11 et 4.12.



Ni/Si/Ei/Oi: entrée nord/sud/est/ouest du PE i.
 Li: sortie locale du PE i.
 P0/P32: entrée/sortie pipeline du segment.

FIGURE 4.11: SEGMENT DE 32 PEs IMAGE2p+

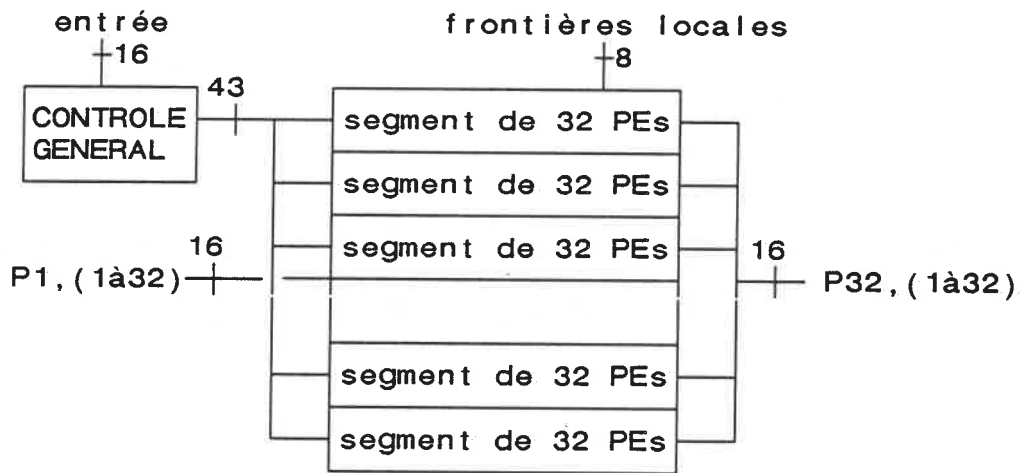


FIGURE 4.12: MATRICE DE 32 * 32 PES IMAGE2p+

A notre connaissance, ce système peut fonctionner suivant les trois modes de traitement identifiés: région pleine, régions creuses, ligne ou colonne.

4.6- TRAVAUX FUTURS

Un système de traitement secondaire performant adapté aux algorithmes utilisant intensivement la multiplication est, selon nous, désirable. Un processeur de type papillon (une multiplication et deux additions par cycle) et un réseau d'interconnexion double (deux sorties par PE par cycle) de type entièrement connecté serait de haute performance et très flexible. Un adressage indirect du réseau d'interconnexion (mémoire locale d'environ 2×16 mots de 5 bits) garantirait une flexibilité suffisante pour résoudre une grande quantité d'algorithmes utilisant intensivement la multiplication et l'addition. Un système de 16 PEs de ce type permettrait de réaliser la transformée rapide et bidimensionnelle de Fourier pour des images de 1024×1024 . La nature homogène du réseau de communication faciliterait l'application de HCDE même au réseau d'interconnexion. Considérant une mémoire de 3 lignes par système et des lignes de jusqu'à 1024 pixels, ce système nécessiterait 192 mots de mémoire par PE. Considérant une mémoire dynamique, de 1 transistor par cellule et un unité arithmétique de 16 bits, le PE comporterait environ:

$$1.1 \times (192 \times 16 + 256 \times 32 + 2 \times 28 \times 16 + 16 \times 16) = 13550 \text{ transistors.}$$

Considérant 30 carrés par transistor, un PE occuperait environ 407000 carrés. Un système de 18 PEs (deux PEs de remplacement) occuperait 7.3 mm^2 (processeurs) $+4\text{mm}^2$ (réseau d'interconnection) $=30 \text{ mm}^2$ avec une technologie de 1 micron. Ce système est réalisable sur une seule puce. Une puce de 750mm^2 permettrait d'incorporer 16 systèmes suffisamment performants chacun pour effectuer une transformée de Fourier sur une image NTSC; et suffisamment flexibles pour opérer comme réseau neuronique (PE multiplieur accumulateur).

Il nous vient également à l'esprit un système SIMD relativement simple mais très performant en mode de calcul réseau neuronique. Chaque PE est un multiplieur par une constante commune (circuit de contrôle). Cette constante correspond à un noeud de sortie du système. Le multiplicande provient de la mémoire locale de chaque PE et correspond au coefficient relatif au noeud évalué. Un transfert séquentiel des sorties des cycles précédents du système amont (un mot de sortie) alimente un nouveau noeud d'entrée à chaque cycle. Un réseau de n noeuds peut être réalisé en n cycles avec n processeurs. Une séquence de systèmes de ce type, réaliserait puissamment un quatrième niveau de traitement hautement flexible (réseau neuronique), mais en utilisant un mode de traitement secondaire (synchrone).

S'il nous était permis de donner notre opinion sur un contexte technologique avantageux pour un système performant de traitement d'image. Nous indiquerions immédiatement l'avantage d'un procédé associant sur un même substrat de la mémoire dynamique et des circuits logiques. Il serait alors possible de distribuer la mémoire principale du système aux PEs, simplifiant considérablement les contraintes de bandes passantes. Dans un contexte où des prototypes de mémoire de 16 mbits par puce deviennent possibles, nous sommes tentés d'inspirer nos confrères japonais à tenter cette fusion. Même s'il s'avérait impossible d'intégrer un système complet sur une seule puce, nous serions ravis de disposer de puces permettant d'augmenter la puissance de calcul proportionnellement à la mémoire, réalisant alors notre objectif au niveau carte: un système de traitement d'images en temps réel économique et performant. Une superficie de 4 microns² par bit est considérablement moins encombrante (37.5 fois) à ce que nous avons réalisé au laboratoire (150 carrés de taille minimale, 3 transistors). Une économie importante de superficie est donc possible en combinant ces deux expertises: mémoires dynamiques denses et notre architecture de système.

CHAPITRE 5

EXEMPLE D'IMPLANTATION, PARTIE HCDE

5.1- INTRODUCTION.

Le chapitre 4 contient une description de l'aspect fonctionnel de la puce IMAGE2. Ce cinquième chapitre contient une description de l'aspect HCDE de la puce IMAGE2.

La section 5.2 décrit HCDE au niveau segment de PEs tel qu'implanté dans la puce IMAGE2. La section 5.3 décrit le concept d'implantation de HCDE pour d'éventuels deuxième et troisième niveaux hiérarchiques. L'exemple pour un système comportant 1024 PEs fonctionnels y est décrit du point de vue de HCDE.

La section 5.4 contient nos mesures (puce IMAGE2) et estimés (exemples proposés) des paramètres de superficie implantés et projetés. La section 5.5 contient une analyse des gains de rendement et de facteur de mérite prévus pour la puce IMAGE2 et le système hypothétique de 5*1024 processeurs IMAGE2p+ de la section 5.4. Nous montrons les avantages de HCDE sur le rendement et la fonctionnalité pour les exemples décrits.

La section 5.6 contient nos conclusions sur: 1-les

facteurs de mérite obtenus (rendement et fonctionnalité) et 2-la faisabilité commerciale d'un système de traitement primaire d'images en temps réel performant et économique. Nous complétons cette section en identifiant certains travaux futurs.

5.2-HCDE DANS IMAGE2

5.2a-CONTEXTE

Cette section décrit l'aspect HCDE de la puce IMAGE2. Le concept de système de traitement d'images en temps réel, présenté au chapitre 4, comporte plusieurs niveaux de parallélisme invitant l'application de HCDE (plusieurs niveaux hiérarchiques CDE reconfigurables possibles, section 5.3). Dans IMAGE2, les restrictions sur la quantité de transistors par puce permettaient la réalisation d'un seul niveau hiérarchique CDE reconfigurable. Tel que décrit dans le chapitre précédent (sections 4.2 à 4.6) la partie fonctionnelle d'IMAGE2 est un segment de 8 PEs ($n=8$).

Dans notre exemple, nous n'avons pas tenté d'optimiser l'application de HCDE et ne prévoyons donc pas d'avantage à l'application de HCDE à la puce IMAGE2. Nous visions plutôt la création d'une famille de cellules en prévision de futures implantations plus denses, lorsque les

superficiés et densités disponibles seraient suffisantes. Envisageant des systèmes de 1024 PEs, r a été choisi plus ou moins arbitrairement pour être avantageux pour des segments d'environ 32 PEs fonctionnels ($n=32$). Afin qu'un défaut affectant deux PEs adjacents n'invalide pas un segment, nous avons choisi de prévoir au moins deux PEs de remplacement ($r=2$). Le dessin des masques d'IMAGE2 comporte donc 10 PEs ($n+r = 10$) dont 8 sont fonctionnels après reconfiguration.

Le circuit complémentaire a été ajouté directement à l'instance PE. Les PEs de remplacement sont identiques aux PEs fonctionnels. La régularité est ainsi maintenue après l'application de HCDE. Au dessin des masques, dans la mesure où r demeure 2, il est possible de changer le nombre de PEs fonctionnels. Ceci se fait en augmentant le nombre de cellules de paires de PEs ($2pe2.kic$), branchant les entrées/sorties supplémentaires, en redimensionnant les tampons de commande et les barres d'alimentation. Les dessins des masques des PEs, du circuit complémentaire d'un PE et du circuit de commande n'ont pas à être modifiés.

Le circuit complémentaire CDE du segment est transparent à l'utilisateur (invisible) après initialisation (mode normal d'opération). L'ensemble du fonctionnement et

du circuit complémentaire correspondent à la description du second chapitre de ce mémoire, pour le cas: reconfiguration verticale symétrique, $r=2$ (section 2.6, figures 2.10 et section 2.6b, figure 2.6c).

La section 5.2b identifie les frontières des circuits tolérants aux défauts des second et troisième niveaux hiérarchiques CDE. Les sections 5.2c à 5.2g décrivent les circuits complémentaires d'un module PE d'IMAGE2, soit respectivement: 5.2c-cellule de vérification, 5.2d-registre de diagnostic, 5.2e-les multiplexeurs de reconfiguration horizontale, 5.2f-le décodeur de reconfiguration verticale et 5.2g-multiplexeurs de reconfiguration verticale. Enfin, la section 5.2h décrit le circuit de commande des circuits complémentaires d'un segment dans IMAGE2.

5.2b-FRONTIERES DES CIRCUITS TOLERANTS DU SECOND NIVEAU HIERARCHIQUE

Cette section montre comment la géométrie des modules a été choisie. Elle décrit également les frontières des circuits tolérants aux défauts de chaque niveau hiérarchique.

Pour minimiser la superficie intolérante aux défauts

(lignes de commande aux PEs au nombre de 135 et lignes du circuit complémentaire au nombre de 15), nous avons dessiné un PE mince et long (longueur/largeur =14). La figure 5.1 montre les limites des circuits tolérantes aux défauts des second (IMAGE2) et troisième (section 4.7) niveaux hiérarchiques. IMAGE2 ne comportant qu'un seul niveau hiérarchique CDE reconfigurable, le troisième niveau de tolérance est celui prévu pour des versions futures plus denses (section 5.3).

Pour minimiser la superficie intolérante au premier niveau hiérarchique CDE, nous voulions englober les multiplexeurs aux bornes de chaque paire de PEs dans la partie tolérante du premier niveau hiérarchique (figure 5.1), considérant une paire de PEs comme module. Ce choix aurait cependant limité à 5 le nombre de modules par segment, réduisant dangereusement la fiabilité de la ligne de comparaison. Nous avons alors décidé de considérer un PE plutôt qu'une paire de PEs (module composé) comme module au premier niveau hiérarchique CDE.

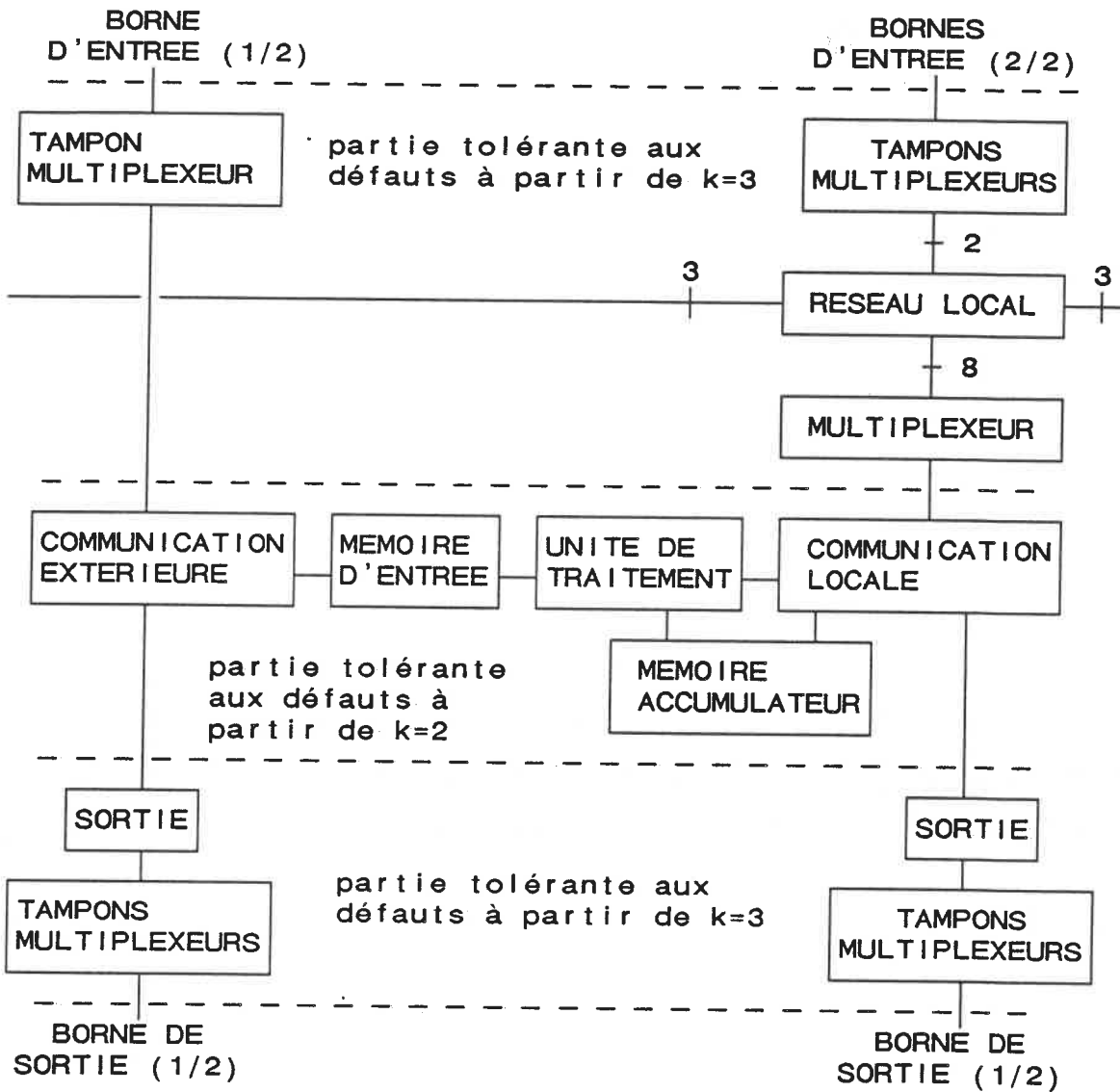


FIGURE 5.1: DIAGRAMME BLOC GENERALISE DU PE D'IMAGE2

La partie régulière du circuit complémentaire (à chaque PE) comprend une cellule de vérification, un registre de diagnostic, une cellule de décodage du diagnostic vertical, deux paires d'interrupteurs pour la reconfiguration horizontale (figure 5.2), 4 triplets d'interrupteurs pour la reconfiguration verticale (figures

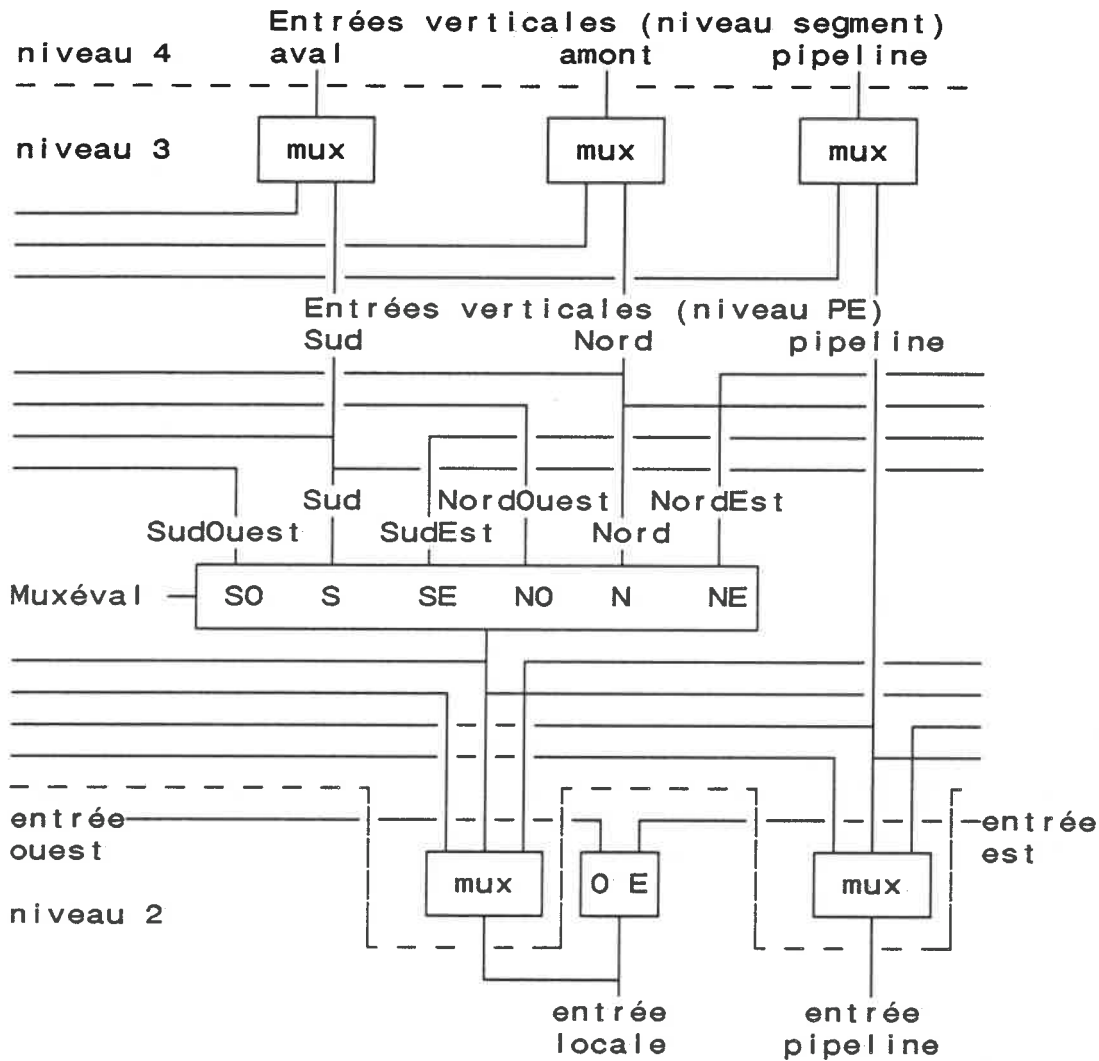
5.1 et 5.2) et 2 paires de transistors pour l'injection de signaux de vérification aux entrées locale et pipeline (sortie des multiplexeurs, figure 5.2). La cellule de vérification (multiplexeur d'accès, cellule de comparaison) est tolérante aux défauts.

La superficie intolérante aux défauts est prépondérante dans la formule d'estimation du rendement. Nous préférons donc inclure le multiplexeur d'entrée locale du PE dans la partie tolérante aux défauts. Une analyse sommaire de la superficie intolérante aux défauts nous a cependant démontré qu'il était préférable, dans notre cas, de reconfigurer au noeud de sortie du multiplexeur d'entrée locale (9 transistors intolérants) plutôt qu'à chacun des six noeuds d'entrée verticale (18 transistors intolérants). Nous avons donc opté pour reconfigurer à la sortie de la section verticale du multiplexeur d'entrée (figure 5.2). Le design d'IMAGE2 gagne ainsi sur deux plans: économie de $3*6 - 2*6 = 6$ transistors intolérants aux défauts (avantage de rendement) et de $3*(6-1) = 15$ transistors de reconfiguration (avantage de superficie). En minimisant la superficie intolérante aux défauts du module PE, nous maximisons le facteur de mérite (chapitre 3) au niveau segment. De plus, le multiplexeur d'entrée étant dynamique, chaque interrupteur de reconfiguration ne requiert alors qu'un seul

transistor plutôt que 2.

Il est donc prévu que le multiplexeur d'entrée locale et les multiplexeurs aux broches ne feront partie du circuit tolérant aux défauts qu'à partir du niveau hiérarchique suivant (troisième niveau).

Aux sorties des PEs, les multiplexeurs aux broches de sortie et les tampons de sortie des sorties locales et pipeline sont intolérants aux défauts au second niveau CDE. Il est prévu que ceux-ci seront tolérants à partir de l'éventuel troisième niveau hiérarchique CDE.



l: interrupteur
 mux: multiplexeur
 niveau k: k est le niveau à partir duquel le circuit est tolérant aux défauts.

FIGURE 5.2: LIMITES DES REGIONS TOLERANTES AU MULTIPLEXEUR D'ENTREE D'UN PE

5.2c-CELLULE DE VERIFICATION

Pour améliorer l'observabilité des noeuds internes, le multiplexeur de sortie locale du PE a été augmenté de quelques noeuds d'entrée additionnels afin d'accéder certains noeuds précédemment inaccessibles dans le PE fonctionnel original: la sortie pipeline, la sortie de la cellule de retenue, etc. Nous avons ajouté 6 interrupteurs supplémentaires pour la lecture de certains noeuds du circuit complémentaire et 2 pour l'injection de signaux de test. La sortie locale permet donc d'observer directement tous les noeuds de sortie (locale et pipeline), une grande quantité de noeuds internes et indirectement (par transparence) la majorité des noeuds internes et les noeuds de sortie du circuit complémentaire. Le coût total de ces transformations a été de 20 transistors tolérants aux défauts et quatre lignes de commande non-tolérantes aux défauts. Ces transistors faisant partie du circuit d'accès nécessaire à la vérification du PE sera considéré partie fonctionnelle plutôt que complémentaire.

La figure 5.3 montre un schéma de la cellule de vérification; tous les interrupteurs et inverseurs sont en logique complémentaire.

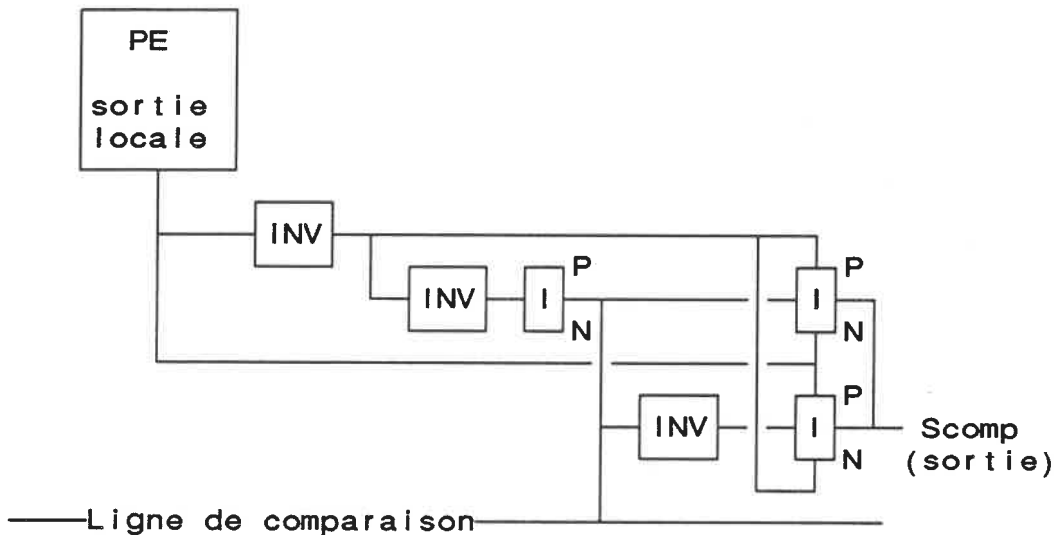


FIGURE 5.3: CELLULE DE VERIFICATION D'UN PE.

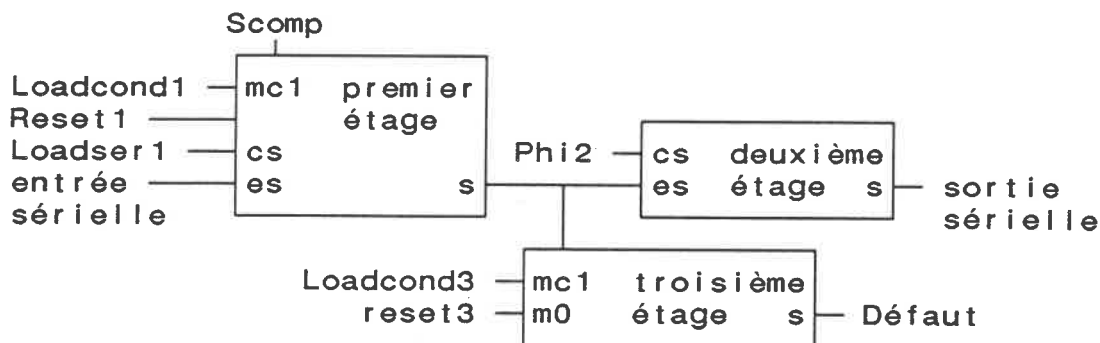
Un tampon trois états formé de deux inverseurs et d'un interrupteur projette le signal de sortie du PE sur la ligne de comparaison analogique. L'interrupteur sert à isoler les sorties des PEs lors de la vérification des cellules de comparaison. Le seuil de ce circuit est près de 2.5 volts (2.45 volts). La vérification des cellules de comparaison se fait par l'injection d'un signal provenant d'un onzième tampon trois états situé dans le circuit de commande au niveau segment de PEs (figure 5.8).

La cellule de comparaison est un XNOR (NON-OU-EXCLUSIF) comportant 8 transistors (figure 5.3). La sortie Scomp de la cellule de vérification locale peut être chargée dans le registre de diagnostic du PE (section 5.2d).

La polarité de la sortie Scomp de la cellule de comparaison et de l'entrée du premier étage du registre de diagnostic sont inversées. Lorsque la sortie d'un PE est identique à celle de la valeur de la ligne de comparaison, la sortie est valide. Par opposition, si la sortie d'un PE est défectueuse, elle est différente de celle de la ligne de comparaison.

5.2d-REGISTRE DE DIAGNOSTIC DES PEs

La figure 5.4 montre le registre de diagnostic complet qui est composé de trois étages distincts. L'entrée Scomp est la sortie de la cellule de comparaison. Les deux premiers étages sont dynamiques, le troisième est statique.



cs: commande de chargement sériel
 es: entrée sérielle
 s: sortie
 m0: commande de mise à 0
 mc1: commande de mise à 1 conditionnelle

FIGURE 5.4: REGISTRE DE DIAGNOSTIC D'UN PE

L'état du premier étage peut être modifié de 3 façons distinctes: mise à 0 (signal de commande Reset1), mise à 1 conditionnelle à la détection d'une erreur (sortie Scomp de la cellule de comparaison et signal de commande Loadcond1) et chargement sériel du contenu du deuxième étage du registre du PE voisin de gauche (signal de commande Loadser1).

Le canal de données sériel (d'un PE au suivant) traverse les premiers et deuxième étages du registre de diagnostic formant ensemble un registre dynamique à deux phases. Ce mode d'opération permet de charger et lire sériellement le registre RD, un PE à la fois. L'entrée du premier étage provenant de la cellule de test est une mise à 1 conditionnelle: seul un 1 peut être chargé. Un diagnostic antérieur (défaut) est ainsi sauvegardé même si des cycles ultérieurs ne détectent pas d'erreurs. Le chargement conditionnel sert à filtrer les transitions imprévues.

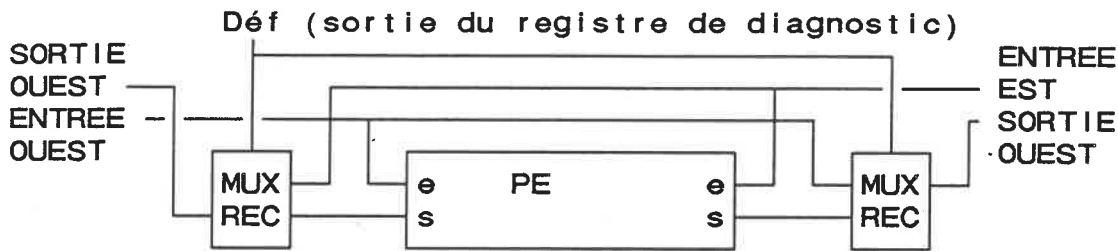
Le second étage est dynamique et propage le signal du premier étage pour le mode sériel d'opération de RD (registre de diagnostic du segment: $n+r = RD = 10$ bits), sur la commande directe de Phi2.

Le troisième étage permet de stocker statiquement la

valeur de sortie du premier étage. La sortie Défaut du troisième étage indique le diagnostic (fonctionnel/défectueux) pour chaque PE. Comme pour le premier étage, le chargement est conditionnel (signal de commande Setcond3): seul un 1 peut modifier l'état du registre; permettant ainsi de conserver les erreurs précédemment stockées. Cet étage peut être remis à 0 par le signal de commande Reset3. Il est possible de lire le contenu du registre de diagnostic en effectuant un chargement du troisième étage dans le premier en passant à travers le PE, via le canal: multiplexeur de diagnostic, RAM, multiplexeur de sortie, cellule de vérification, premier étage, lecture sérielle du registre de diagnostic.

5.2e-MULTIPLEXEURS DE RECONFIGURATION HORIZONTALE

Le segment étant horizontal, la reconfiguration horizontale (figure 5.5) est obtenue en ajoutant une paire d'interrupteurs à chaque sortie horizontale. Dans ce cas-ci, chaque PE comporte deux sorties horizontales (Est et Ouest) et chaque interrupteur comporte un seul transistor. Chaque signal de sortie horizontale est dévié au moyen d'un multiplexeur à deux entrées directement commandé par la sortie Déf du troisième étage du registre de diagnostic.



MUX REC: multiplexeur de reconfiguration horizontale (2 interrupteurs).

FIGURE 5.5: RECONFIGURATION AUX SORTIES HORIZONTALE d'IMAGE2

5.2f-DECODEUR DE RECONFIGURATION VERTICALE

Cette section décrit le circuit de décodage de la reconfiguration verticale pour un PE. Son fonctionnement correspond à celui du cas symétrique (cas $r=2$) dont le fonctionnement est décrit à la section 2.6b. Il serait redondant de répéter ici l'explication complète du fonctionnement de ce circuit. Cette section vise seulement à faciliter la compréhension d'IMAGE2. La figure 5.6 montre le schéma bloc du décodeur de diagnostic vertical d'IMAGE2.

Tous les canaux non-horizontaux sont considérés verticaux: nord, sud, nord-est, nord-ouest, sud-est, sud-ouest, tout canal qui n'est pas est ou ouest. Le circuit de reconfiguration verticale comporte deux sections principales: un décodeur de diagnostic déterminant les branchements et une série de triplets interrupteurs effectuant la déviation des canaux de données. Les chemins

de données verticaux sont reconfigurés en déviant les entrées et sorties verticales du PE, vers la gauche, la droite, ou non déviés; selon la présence ou l'absence de PEs défectueux dans chaque direction, relativement à la localisation du PE local.

Le circuit de décodage vertical de chaque PE garantit que s'il y a 2 PEs identifiés défectueux par segment et que le reste du segment est fonctionnel, la fonctionnalité du circuit est assurée et la reconfiguration est automatique et immédiate.

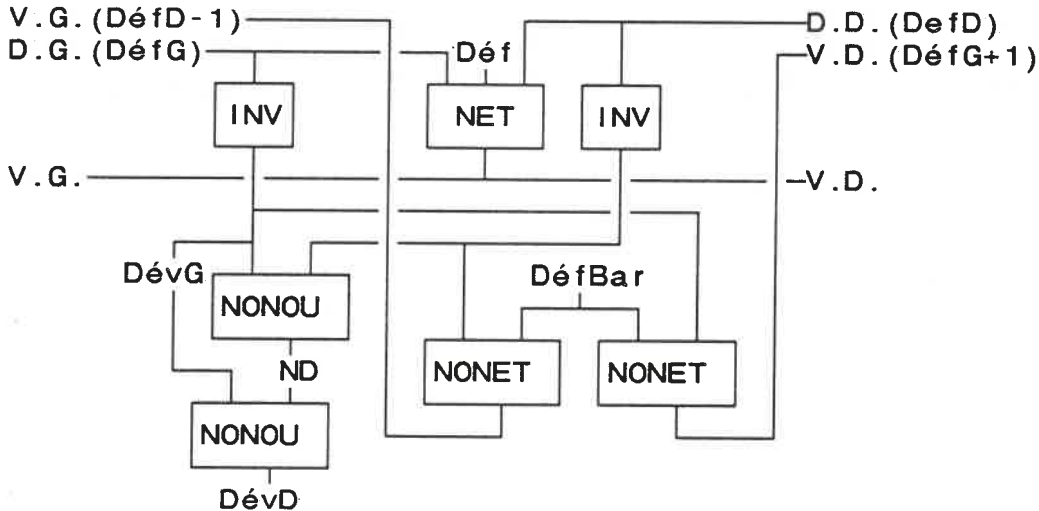


FIGURE 5.6: DECODEUR DE RECONFIGURATION VERTICALE

Dans IMAGE2, toutes les portes du décodeur sont en logique complémentaire, sauf la porte NET (non-ET) à la sortie SD (segment défectueux) qui fait partie d'une porte

ET-OU câblée unipolaire déterminant s'il y a plus que 2 PEs défectueux au niveau segment. La sortie SD indique un segment défectueux sur détection, par un PE, d'au moins trois PEs défectueux. Le tableau 1 montre la table de vérité des signaux du décodeur de diagnostic vertical d'IMAGE2. Notez que l'état "sans importance" a été remplacée par "dévier à gauche" pour simplifier le circuit de décodage (nombre de transistors par décodeur).

TABLEAU 5.1: TABLE DE VERITE DES SIGNAUX DévG, DévD, ND et SD

AU MOINS UN PE DEFECTUEUX A		SORTIE DU REGISTRE DE DIAGNOSTIC LOCAL	
DROITE	GAUCHE	FONCTIONNEL	DEFECTUEUX
non	non	dévier à gauche	dévier à gauche
non	oui	dévier à droite	dévier à droite
oui	oui	ne pas dévier	segment defectueux
oui	non	devier à gauche	dévier à gauche

5.2g-MULTIPLEXEURS DE RECONFIGURATION VERTICALE

Les multiplexeurs à 3 entrées sont des triplets d'interrupteurs directement commandés par les signaux DévG, DévD et ND. Les interrupteurs de reconfiguration ont été adjoints aux multiplexeurs du PE: multiplexeur d'entrée,

multiplexeur de sortie locale et entrée(sortie) pipeline. Ceux-ci étant dynamiques, les interrupteurs de reconfiguration verticale comportent un seul transistor par interrupteur ce qui simplifie la cellule et sa commande. Les schémas au niveau transistor apparaissent au chapitre 9 de la documentation d'IMAGE2 [26] que nous reproduisons ici pour faciliter la compréhension (figures 5.7 et 5.8).

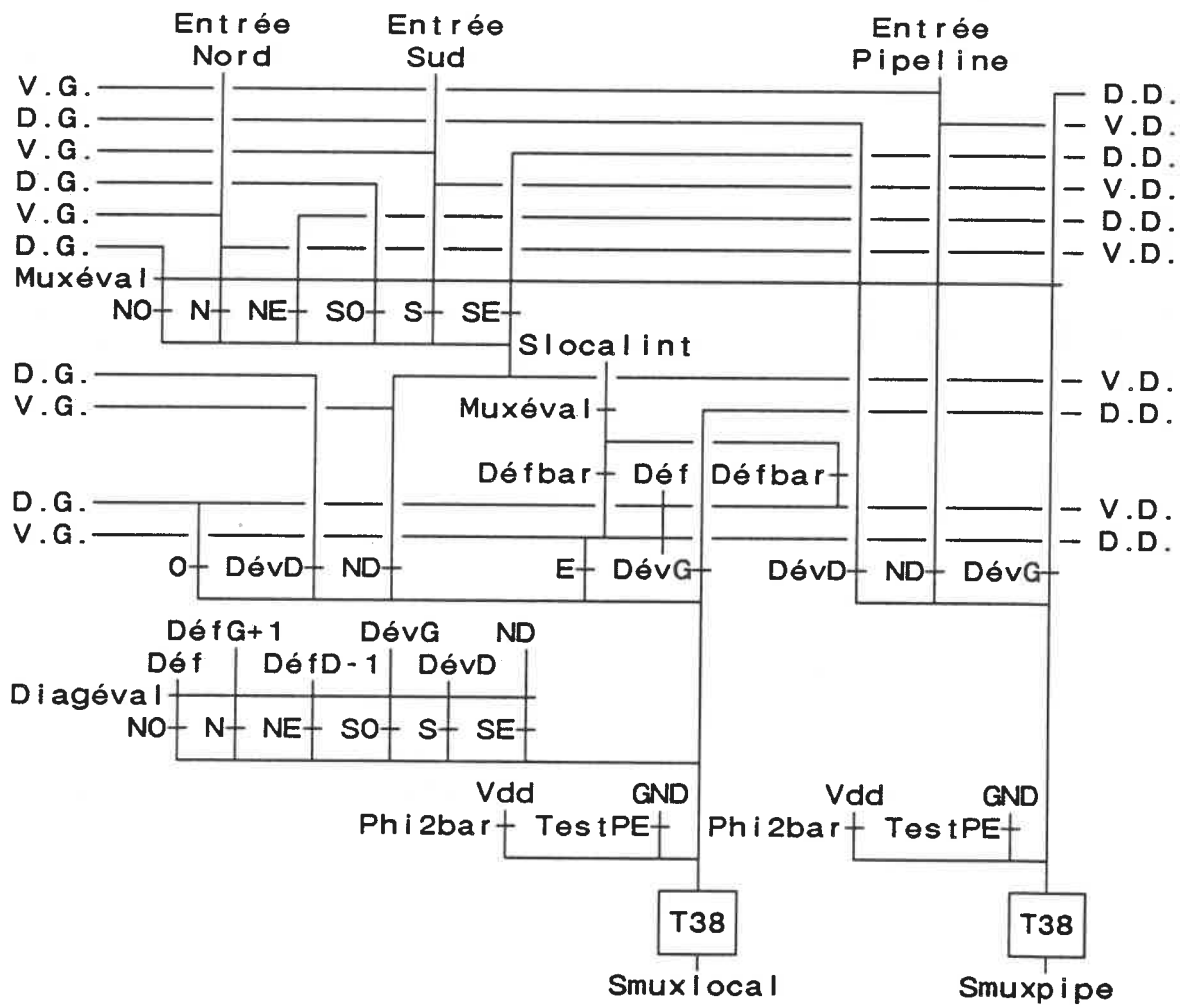


FIGURE 5.7: RECONFIGURATION DU MULTIPLEXEUR D'ENTREE

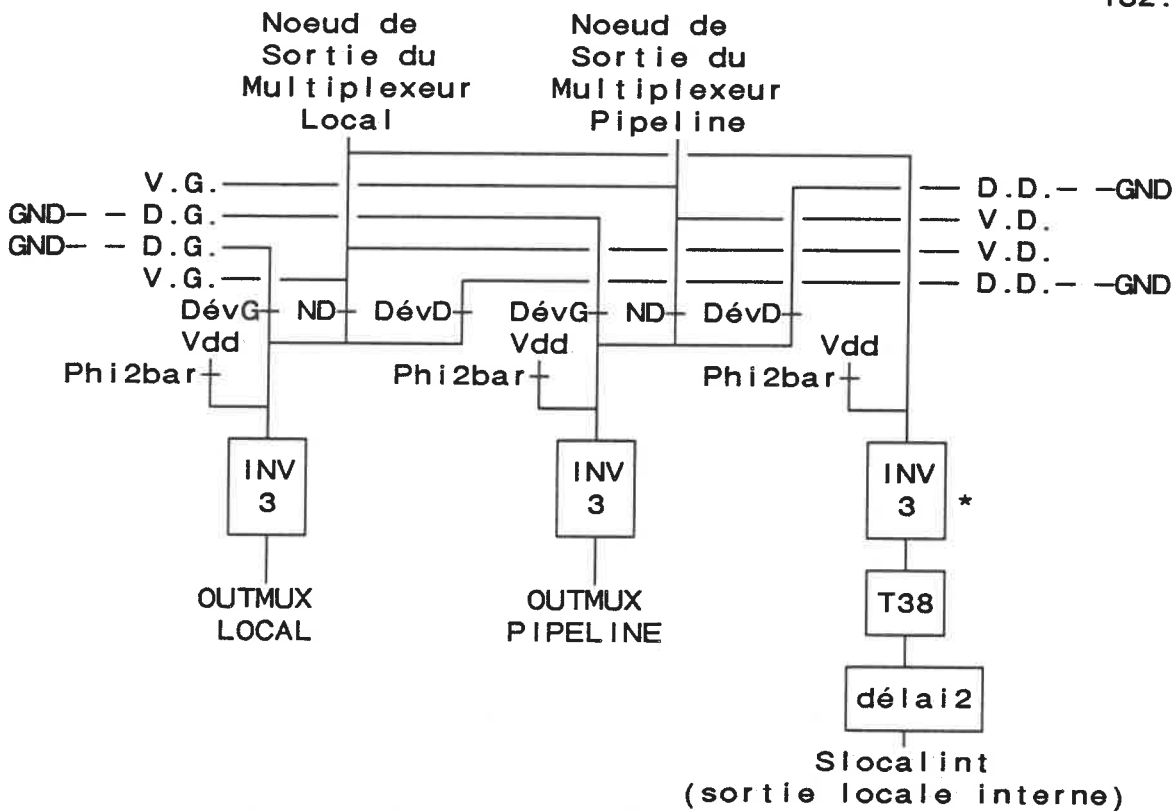


FIGURE 5.8: RECONFIGURATION DES SORTIES LOCALES ET PIPELINE

La reconfiguration de l'entrée pipeline comporte 6 transistors: trois paires interrupteur-évaluation. La reconfiguration de l'entrée locale (figure 5.7) comporte deux parties distinctes: 1-Les entrées verticales et diagonales du multiplexeur d'entrées locales se trouvent en amont d'un multiplexeur de reconfiguration verticale à trois entrées (3 transistors), 2-les entrées horizontales dont le multiplexeur de reconfiguration est un multiplexeur à deux entrées (4 transistors).

Une approche similaire a été utilisée pour la

reconfiguration des signaux de sortie. Les interrupteurs ont été imbriqués aux multiplexeurs de sortie locale et pipeline. Cette paire de multiplexeurs n'ajoute donc que 6 transistors par PE.

5.2h-COMMANDE DU CIRCUIT COMPLEMENTAIRE

Un registre de commande (RCC) détermine l'état de chaque ligne de commande des circuits complémentaires des PEs pour tout le segment de PEs. RCC est statique, transparent et de 9 bits. Quelques portes élémentaires et quelques tampons de distribution complètent le circuit de commande du circuit complémentaire d'image2 et des horloges de la puce (figures 5.9 et 5.10).

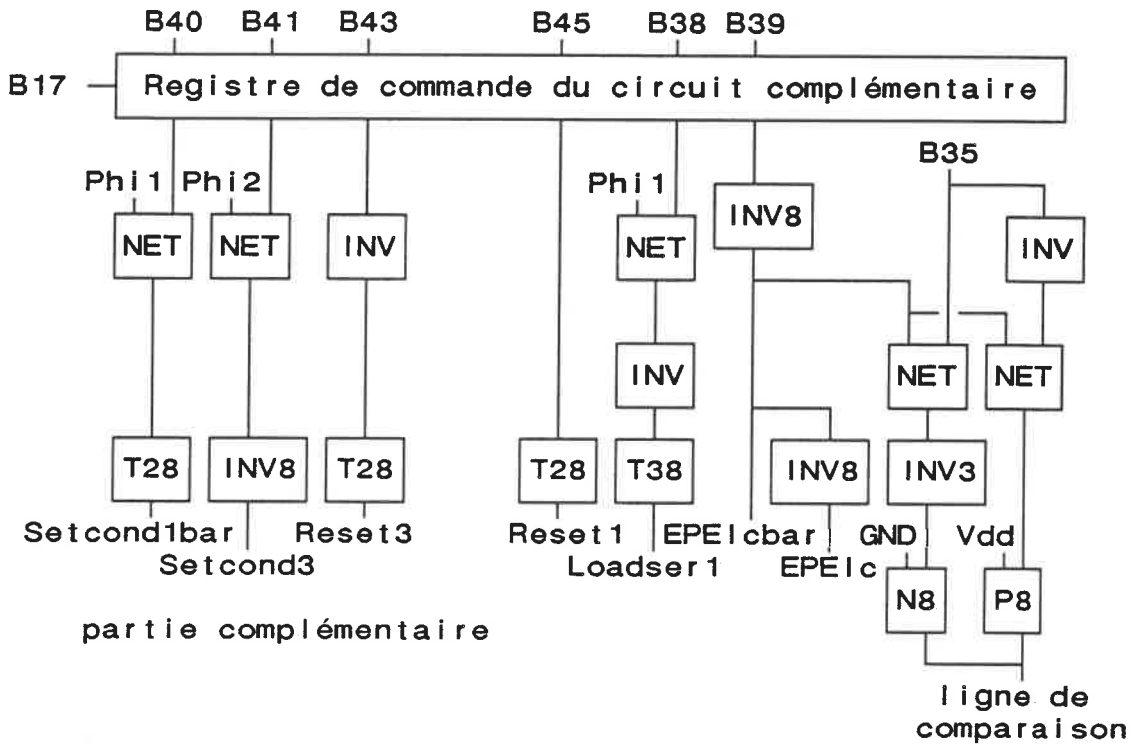


FIGURE 5.9: CIRCUIT DE COMMANDE DU REGISTRE DE DIAGNOSTIC

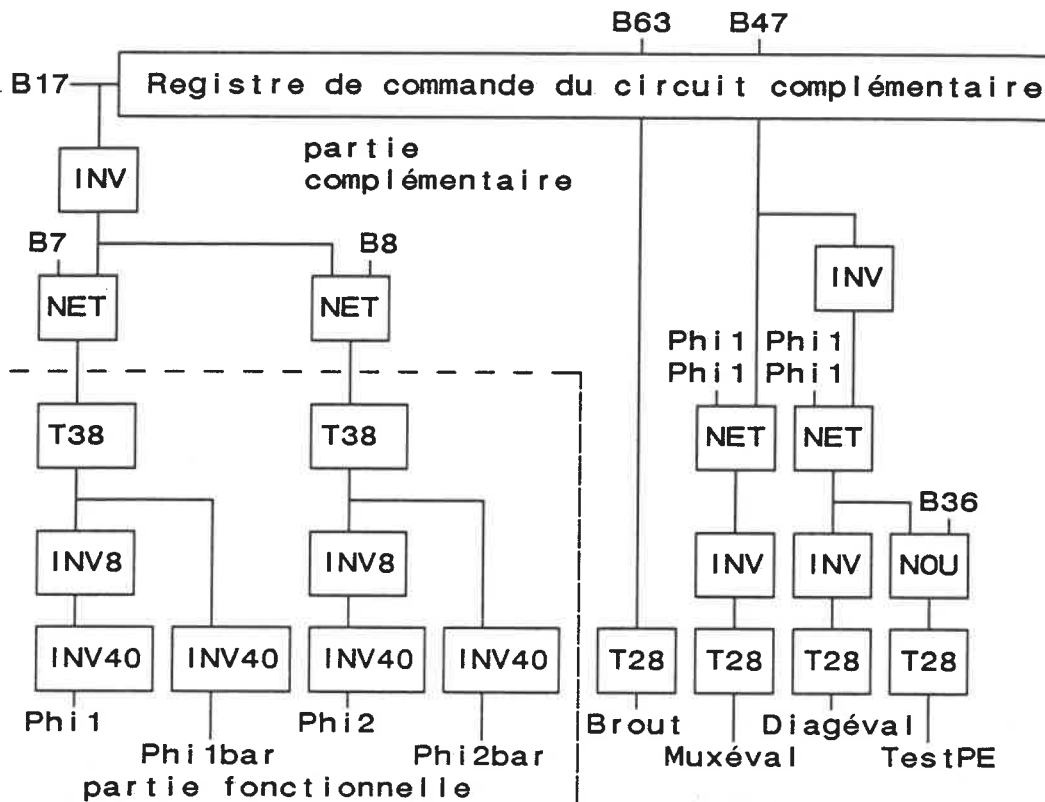


FIGURE 5.10: CIRCUIT DE COMMANDE DES CIRCUITS COMPLEMENTAIRES DES MULTIPLEXEURS D'ENTREE ET DE L'HORLOGE

Une broche d'entrée a été allouée pour commander le chargement de RCC. Tous les autres signaux empruntent des lignes déjà utilisées par les signaux de commande et certaines lignes d'entrée. Le coût des circuits complémentaires en broches se limite à une seule broche additionnelle. Ceci permet d'éviter d'ajouter une quantité massive de broches en appliquant CDE au segment. L'horloge du système est débranchée pendant le chargement de RCC, maintenant les états de la partie fonctionnelle intacts pendant le chargement de RCC.

Une ligne de commande permet d'invalider les deux PEs d'extrémité lorsqu'on ne veut pas profiter de CDE. Cette ligne ne faisant pas normalement partie de HCDE a été exclue des calculs d'estimation du rendement, ainsi que le circuit commande associé. Les signaux de commande du multiplexeur d'entrée ont été réutilisés pour accéder aux noeuds de sortie du circuit complémentaire. L'accès à ces noeuds n'a ainsi nécessité l'addition que de 2 lignes de commande. Deux bornes de sorties des PEs ont été réutilisées pour accéder aux sorties du registre de diagnostic (RD) et à la sortie SD (segment défectueux) provenant du circuit complémentaire des PEs.

5.3-HCDE AUX NIVEAUX HIERARCHIQUES SUPERIEURS

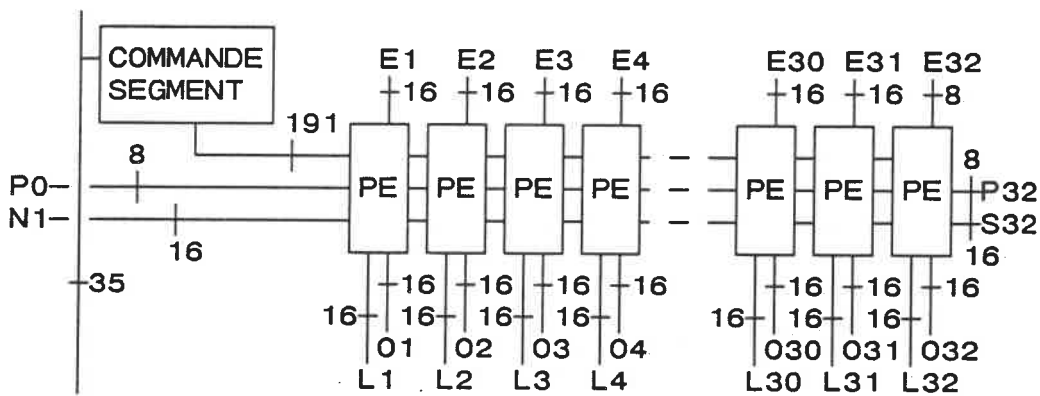
5.3A-PREAMBULE

Cette section décrit comment nous proposons d'appliquer HCDE à d'éventuels troisième et quatrième niveaux hiérarchiques CDE. Le dessin des masques des PEs a été conçu de manière à pouvoir être réutilisés pour réaliser des segments de PEs plus importants dans la mesure où le nombre de PEs de rechange (r) demeure 2. On peut réaliser un circuit avec un niveau hiérarchique CDE reconfigurable supplémentaire si on dispose d'une superficie plus grande.

Par exemple, une puce de 1 cm^2 dans une technologie de 1.5 um permettrait d'implanter une matrice fonctionnelle de $16 * 8 = 128$ PEs de type IMAGE2, sans modifier la disposition géométrique des cellules réalisées. Une puce de cette densité remplacerait 16 puces de type IMAGE2. Voyons maintenant de combien nous pouvons augmenter la fonctionnalité d'IMAGE2 sans sacrifier le rendement. Pour fins de simplicité, nous utilisons l'exemple de système de 1024 processeurs IMAGE2p+ proposé à la section 4.6.

Le processeur étant parallèle, chaque canal comporte 8 bits (IMAGE2 a un bit par canal). Le premier niveau hiérarchique sera similaire à celui du PE d'IMAGE2 sauf que le nombre de multiplexeurs augmente d'un facteur de 16(8) aux entrées locales(pipeline). Nous estimons la complexité d'un PE à 3000 transistors. Plutôt que d'optimiser, nous utilisons $r=2$ pour chaque niveau CDE reconfigurable. Notre premier choix de système consiste à envisager un partage fonctionnellement symétrique des niveaux hiérarchiques du système SIMD: $n_2 = n_3 = 32$. Le quatrième niveau (MIMD: 5 systèmes SIMD en cascade) comporterait 7 systèmes SIMDs: $n_4 = 5$. Une puce de grande superficie d'environ 12 cm^2 (section 5.4) dans une technologie de 1 micron par carré est ainsi envisagée.

Du point de vue de HCDE, les premiers et second niveaux hiérarchiques sont essentiellement les mêmes que dans le cas d'IMAGE2 sauf que le nombre de PEs par segment est de 34 plutôt que 10 et que chaque canal d'entrée comporte 8 bits par canal plutôt qu'un seul (figure 5.11).



Ni/Si/Ei/Oi: entrée nord/sud/est/ouest du PE i.

Li: sortie locale du PE i.

P0/P32: entrée/sortie pipeline du segment.

FIGURE 5.11: SEGMENT DE 32 PEs IMAGE2p+

Aux troisième et quatrième niveaux hiérarchiques, l'application de HCDE est sensiblement la même sauf pour le nombre de lignes d'entrée/sortie par module et l'orientation du troisième niveau qui est perpendiculaire aux deuxième et quatrième niveaux CDE (figures 5.11 et 5.12).

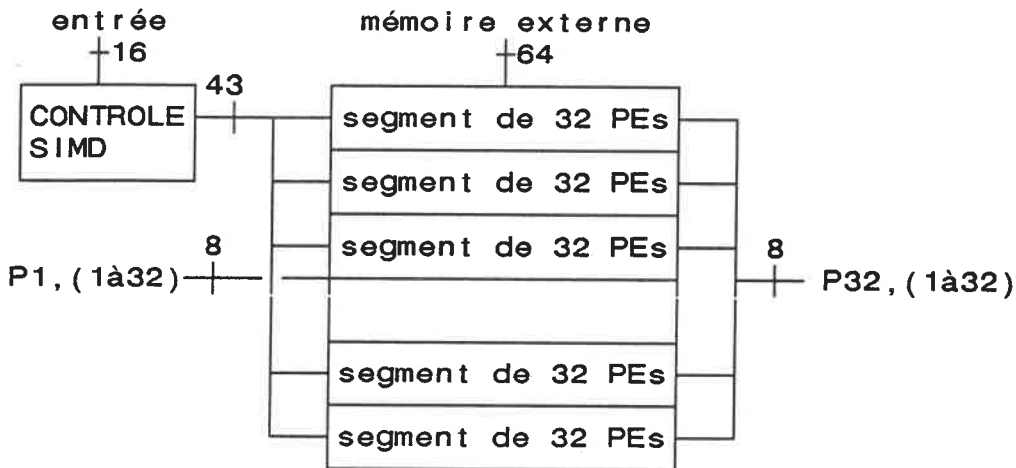
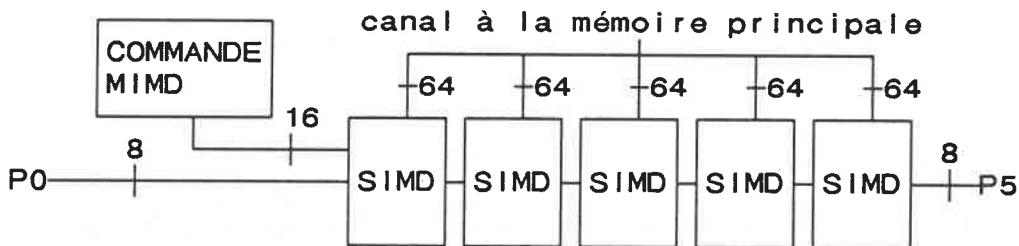


FIGURE 5.12: SYSTEME SIMD DE 32 * 32 PES IMAGE2p+



P0/P32: entrée/sortie pipeline du système

FIGURE 5.13: SYSTEME MIMD DE 5 SIMDS IMAGE2p+

Puisque les systèmes SIMDs n'ont pas à être extensibles, les multiplexeurs de broches aux entrées peuvent être éliminés. Les interrupteurs de reconfiguration du troisième niveau peuvent alors être à un seul transistor incorporé au multiplexeur d'entrée du niveau inférieur. Nous utilisons des interrupteurs à 2 transistors au quatrième niveau pour englober les registres d'entrée/sortie du canal pipeline requis entre les systèmes SIMDs.

Sauf pour la localisation des interrupteurs, les circuits complémentaires des niveaux supérieurs sont les mêmes que pour le second niveau r étant 2 pour chaque niveau CDE. Nous avons prévu un circuit de commande indépendant au niveau SIMD dont l'objectif est de restreindre la bande passante du canal de commande au système hôte (niveau MIMD). Un circuit dont la complexité serait comparable à celle de quelques blocs autonomes de mémoire nano-programme (512 * 70 bits), incluant quelques compteurs de séquençement et un peu de logique suffirait, à notre avis, pour remplir l'objectif visé. Considérant des mémoires statiques (6 transistors), le circuit de commande du niveau SIMD comporterait environ $6 * 512 * 70 = 220000$ transistors (4% de la superficie d'un système SIMD, rendement: environ 90%).

5.3b-GENERATEUR AUTOMATIQUE DE VECTEURS DE TEST

Nous proposons d'utiliser un générateur automatique de vecteurs de vérification pour la vérification des trois premiers niveaux hiérarchiques et la reconfiguration jusqu'au quatrième niveau. La figure 5.14 montre un diagramme bloc d'un concept possible. Ce concept est compatible avec l'approche pseudo-exhaustive de Mc Cluskey et al [22] que nous préconisons [2] pour la puce IMAGE2 (approche pseudo-exhaustive). Le circuit est formé: 1-d'une

cascade de compteurs se succédant, 2-d'un "ROM" ("read only memory"), 3-d'un compteur d'adresse pour le ROM et 4-d'un groupe de multiplexeurs déterminant, pour chaque cycle, l'état de chaque ligne du canal de commande du système SIMD.

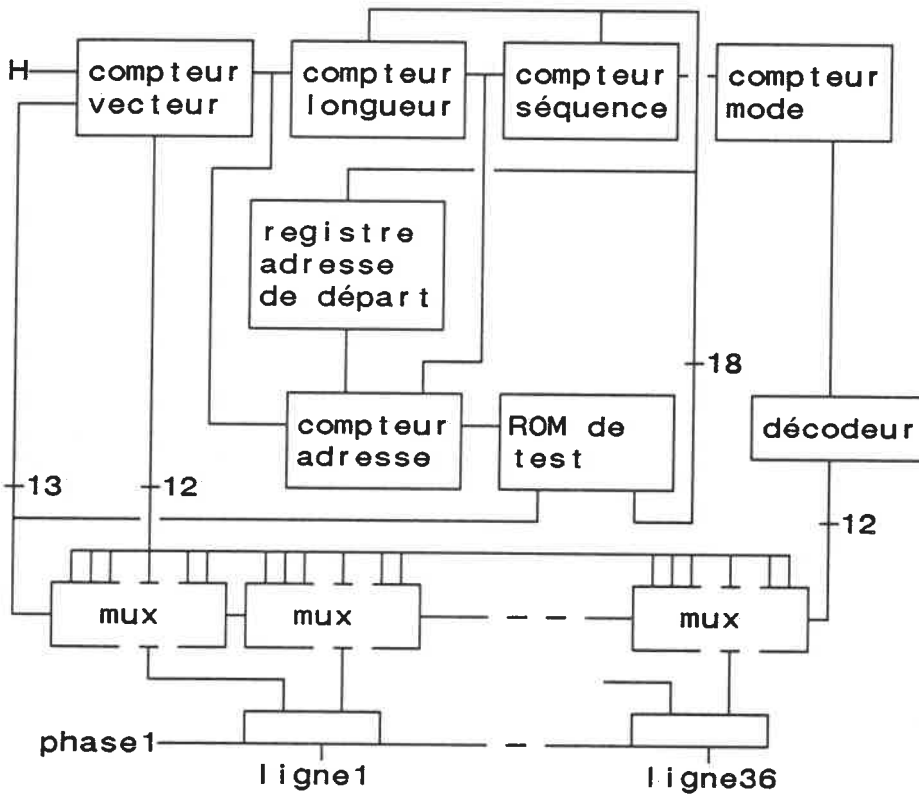


FIGURE 5.14: DIAGRAMME BLOC D'UN GENERATEUR AUTOMATIQUE DE VECTEURS DE TEST

Le compteur de vecteurs est un compteur pseudo aléatoire dynamique de longueur programmable générant les vecteurs de test, sur commande d'un bit du ROM (un cycle par séquence), pour chaque mode de test. Le compteur suivant est une paire: 1-compteur de séquence, 2-compteur d'adresse.

Au début de chaque mode, un cycle est consacré au chargement de l'adresse de départ du ROM (registre de l'adresse de départ du ROM), le nombre d'adresses ROM de la séquence (compteur de longueur) et enfin le nombre de séquences identiques consécutives (jusqu'à $n+r$) avant le mode suivant (compteur de séquence). Ces trois compteurs simplifient le circuit dans la détermination des états du registre RCC dans la vérification du circuit complémentaire de chaque niveau hiérarchique. Le dernier compteur est un compteur de mode établissant la séquence des tests, la longueur du compteur de vecteurs et le choix des signaux (fixes, compteur de vecteur) pour chaque ligne de commande.

82 multiplexeurs déterminent les entrées à chaque entrée de commande. Ces multiplexeurs sont commandés directement par le vecteur de mode de test (compteur de mode). 41 multiplexeurs à deux entrées sélectionnent le signal pour chaque ligne de commande (commandé par la séquence de phases). La mémoire ROM contient: 1-les adresses de départ, 2-la longueur de chaque séquence, 3-le nombre de séquence pour chaque mode et 4-un bit de commande pour bloquer le compteur pseudo aléatoire. Sans effectuer d'analyse exhaustive, nous estimons qu'un ROM d'environ 64 mots de 16 bits suffirait.

Ce circuit permettrait de générer les vecteurs requis pour vérifier la majorité des noeuds internes de la puce aux trois niveaux inférieurs sans passer par les broches. Le quatrième niveau hiérarchique et certains noeuds difficiles à vérifier pourraient être vérifiés en injectant extérieurement les vecteurs de vérification additionnels requis.

5.4-MESURES ET ESTIMES DES SUPERFICIES

Cette section contient nos mesures et estimés de superficie pour les exemples identifiés. Le tableau 5.2 contient les quantités de transistors implantés (IMAGE2) et estimés (IMAGE2p+). Ce tableau comporte 3 sections correspondant chacune à un groupe distinct de cellules. Le circuit complémentaire fixe inclut le circuit complémentaire régulier (au module), excluant les interrupteurs. Ce circuit est ajouté aux modules des trois premiers niveaux CDE. La commande des circuits complémentaires est le circuit de commande des circuits complémentaires des modules. Ce circuit, installé au niveau segment, est le même pour les trois niveaux de groupes homogènes. Il est utilisé aux trois niveaux CDE supérieurs (2, 3, 4). Les interrupteurs correspondent aux interrupteurs de reconfiguration pour

chaque cas (IMAGE2 et IMAGE2p+) et pour chaque niveau hiérarchique CDE. Bien que ces interrupteurs soient partiellement tolérants aux défauts, nous les avons considérés entièrement intolérants pour garantir la validité de notre étude de faisabilité.

TABLEAU 5.2: COMPTES DE TRANSISTORS DES CIRCUITS COMPLEMENTAIRES

CIRCUIT COMPLEMENTAIRE FIXE, MODULE, R=2: (niveaux 1, 2, 3)

cellule:	transistors lignes		
	tol.	int.	int.
comparaison	12		1
registre de diagnostic	2	21	7
décodeur de reconfiguration verticale		23	3
complète	14	44	11

COMMANDE DU CIRCUIT COMPLEMENTAIRE DES NIVEAUX 1, 2, 3
(circuit entièrement intolérant aux défauts)

cellule	transistors	tampons	lignes
registre RCC:	104		1
logique	28	10 (inverseurs tailles 2 et 8)	
complète	132	10 (2,8)	1

INTERRUPTEURS (circuit considéré entièrement intolérant)

IMAGE	niveau	direction	transistors	lignes
2	1	horizontale	4	1
2	1	verticale	12	4
2p+	1	horizontale	32	1
2p+	1	verticale	96	4
2p+	2	horizontale	96	4
2p+	2	verticale	1536	1
2p+	3	horizontale	0	0
2p+	3	verticale	96	4

Les colonnes de transistors indiquent les comptes de transistors tolérants et intolérants aux défauts. La colonne ligne indique le nombre de lignes intolérantes aux défauts (contrôle et données). Il était nécessaire de séparer la superficie des lignes et des transistors afin de permettre d'obtenir des estimés réalistes des superficies dans le cas de l'exemple du système de processeurs IMAGE2p+. Les tampons sont des paires d'inverseurs (un de taille 2 et un de taille 8).

Le tableau 5.3 contient les mesures de superficies au dessin des masques de la puce IMAGE2. La colonne fonctionnelle correspond à la superficie du circuit fonctionnel, sans la superficie de l'alimentation neutre. La colonne neutre indique la superficie correspondant au neutre des éléments considérés. L'importance de cette colonne est expliquée à la section 5.5a. Le facteur de correction pour circuit monocouche est appliqué avant de soustraire la superficie concernée à la superficie hors tout du PE d'IMAGE2 afin de garantir que toutes les surfaces sont incluses dans notre estimé du rendement.

TABLEAU 5.3: MESURES SUPERFICIE, PUCE IMAGE2
(en carrés de dimension minimale: 3microns *3microns)

	fonctionnel	neutre
IMAGE2, processeur		
module tolérant	60234	
module intolérant multicouche	4873	2505
module intolérant une couche	10314	3578
IMAGE2, segment		
commande	125000	
10 PEs	711000	
module intolérant multicouche	131400	80300
module intolérant une couche	53000	13500
IMAGE2 sans HCDE		
segment de 10 PEs et commande	820600	
broches: 68 *380 *380	392000	
hors-tout: .25 cm ²	2800000	

Le tableau 5.4 contient les estimés de superficies (IMAGE2p+) relatives à l'exemple de 5*1024 processeurs de type IMAGE2p+. Nous avons considéré de 30 à 50 carrés par transistor dans nos estimés de superficie en tenant compte des résultats obtenus pour chaque cellule dans le cas de la puce IMAGE2, lorsque disponible. Selon les besoins, nous avons considéré des PEs de même largeur que ceux d'IMAGE2. La superficie des circuits monocouche n'inclut pas l'espacement entre les lignes et correspond à la superficie avant l'application du facteur de correction pour circuit monocouche. La dernière section du tableau 5.4 sert de base au calcul du facteur de mérite fonctionnel (FMF, section 5.5b). L'opération de ce dernier cas a été limité au mode de traitement ligne afin de restreindre le nombre de broches à une quantité réaliste (89 plutôt que 430). De plus, la

superficie des broches n'inclut que la partie multicouche de circuit (plots exclus).

TABLEAU 5.4: ESTIMES DE SUPERFICIE, IMAGE2p+
(en carrés de dimension minimale: 1micron *1micron)

	fonctionnel	neutre
IMAGE2p+, processeur d'environ 3000 transistors		
module tolérant	140000	
module intolérant multicouche	8820	14800
module intolérant une couche	11812	5500
IMAGE2p+, segment de 34 processeurs et commande		
commande	130000	
module tolérant	3500000	315000
module intolérant multicouche	90000	450
module intolérant une couche	122000	68900
IMAGE2p+, SIMD: 34 segments de processeurs et commande		
commande	5000000	
module tolérant	123100000	
module intolérant multicouche	90000	450000
module intolérant une couche	903000	
IMAGE2p+, MIMD: 7 modules SIMD, commande et broches		
module tolérant	899000000	
module intolérant multicouche	7380	450
module intolérant une couche	509000	
IMAGE2p+, MIMD sans HCDE		
processeur		fonctionnel et neutre
segment, commande		140000
segment, 32 processeurs et commande		200000
SIMD, commande		4680000
SIMD, 32 segments et commande		6000000
MIMD, 128 broches		156000000
MIMD, 5 SIMDs et broches		6400000
		786400000
IMAGE2p+, puce pour le calcul du FMF		
processeur		140000
segment de 80 processeurs		11200000
commande		200000
segment, 48 processeurs et commande		11400000
broches (20 sorties, 57 entrées, 16 alimentations)		820000
superficie de circuit pour le calcul du rendement		12220000

5.5-ANALYSE DU RENDEMENT ET DES FACTEURS DE MERITE

5.5a-ORIENTATION ET EFFET D'ALIMENTATIONS PARTAGEES

Cette section contient une analyse des rendements avec l'utilisation de HCDE pour les exemples cités. La section contient aussi une discussion de l'effet des alimentations sur le rendement aux niveaux hiérarchiques supérieurs. La section 5.6b contient les résultats des calculs de rendement et de facteurs de mérite sur la base de la formulation présentée du troisième chapitre.

Au second niveau CDE reconfigurable, le circuit du segment, incluant les PEs, le circuit de commande, le circuit complémentaire des PEs et les multiplexeurs des broches, font partie du module tolérant. Les interrupteurs du niveau segment, les sources des interrupteurs aux entrées/sorties et les lignes de distribution des signaux de commande, les lignes entre les segments et aux broches d'entrée/sortie font partie du circuit intolérant du troisième niveau. Nous excluons volontairement les alimentations du circuit intolérant du troisième niveau parce que nous considérons qu'à partir de ce niveau, le neutre de chaque module (0 volt) doit être indépendant de celui des autres modules.

Nous avons constaté qu'à partir du troisième niveau hiérarchique, l'interdépendance des alimentations des modules devient un facteur prépondérant sur le rendement de notre exemple. Dans IMAGE2, l'alimentation neutre représente environ 9% du circuit multicouche, mesure au laboratoire, excluant toutes les lignes traversant le PE (30 à 50 carrés par transistor). Seul le mode de défaut "court-circuit entre le neutre et l'alimentation Vdd" est fatal en ce qui concerne le neutre. Nous avons négligé le mode de défaut "circuit ouvert", faute de données suffisantes, pour simplifier l'exposé, garantissant alors de couvrir le pire cas.

Le tableau 5.5 montre l'importance relative de l'alimentation neutre sur le rendement de notre exemple de système de $5 \cdot 10^4$ processeurs lorsque les alimentations sont communes et indépendantes. Chaque ligne correspond au rendements respectifs des circuits: 1-neutre seulement, 2-la solution avec neutres communs et 3-une solution avec neutres indépendants. On constate que l'utilisation de CDE restreint l'importance relative du rendement des modules sur le rendement global. Il était donc intuitivement prévisible que l'importance de la partie intolérante aux défauts prenne plus d'importance au rendement global. L'alimentation

devient un facteur prépondérant de chute du rendement (.650 vs..911, .154 vs..780) et, en conséquence, des facteurs de mérite si les alimentations neutre ne sont pas autonomes aux niveaux supérieurs (3 et 4).

TABLEAU 5.5: EFFET DE SEPARER LES ALIMENTATIONS NEUTRE DES SEGMENTS

neutre	rendement		
	34 PEs	34 segments	7 systèmes
1-seul	.993	.802	.198
2-communs	.981	.650	.154
3-séparés	.988	.911	.780

Nous proposons donc d'utiliser un mécanisme de débranchement de l'alimentation neutre des modules à partir d'un niveau hiérarchique justifiant cette opération. Ceci permettrait de sauver des circuits comportant un court-circuit entre le neutre et l'alimentations aux niveaux inférieurs (plus de 96% du circuit). Suivant que le procédé est à puits P (IMAGE2) ou N, le débranchement du neutre (0 volt) ou du vivant (5 volts) permettra cette séparation. Nous envisageons plusieurs mécanismes possibles: 1-fusible de courant, 2-disjoncteur, 3-plots d'alimentation neutre indépendants, 4-fusible au laser.

L'utilisation des deux premières méthodes requiert une analyse de la demande de courant pour éviter le problème de "latch up" dans la technologie CMOS. Dans la première,

une résistance fusible coupe une alimentation fuyante. La seconde solution occupe une superficie importante (transistor de grande superficie) et laisse une superficie importante d'alimentation directe au neutre. La troisième approche consiste à alimenter indépendamment chaque segment en prévoyant une fenêtre indépendante pour l'alimentation neutre de chaque segment. La quatrième approche prévoit un lien métal pouvant être sectionné au moyen d'un faisceau laser lors de la fabrication. Ces deux dernières approches ont l'avantage d'éliminer la résistance additionnelle à l'alimentation neutre, contournant ainsi le problème potentiel de "latch up" et restreignant les pertes. Un robot pourrait être utilisé pour sectionner l'alimentation neutre des segments courts-circuités entre les alimentations.

Nous laissons à d'autres le soin d'identifier, pour chaque procédé une famille de solutions pratiques. Dans le cadre de cet exposé, nous contournons les problèmes potentiels de "latch up" et d'encombrement en retenant l'une des deux dernières solutions. Quelle que soit la méthode de débranchement utilisée, l'alimentation du circuit complémentaire CDE du module d'un niveau hiérarchique doit être la même que celle du segment (niveau hiérarchique supérieur suivant). Ceci afin d'assurer l'intégrité des interrupteurs aux niveaux considérés. Un mécanisme (porte

OU) doit être ajouté au circuit complémentaire pour identifier la condition "alimentation défectueuse", pour que le décodeur de reconfiguration verticale et les interrupteurs de reconfiguration assument les états appropriés.

5.5b-RENDEMENTS ET FACTEURS DE MERITE ESTIMES

Sans tenter d'optimiser, nous avons utilisé un système SIMD symétrique avec $n=32$ aux deux premiers niveaux hiérarchiques CDE reconfigurables. Le tableau 5.6 contient les rendements obtenus en suivant la formulation du troisième chapitre et les superficies des tableaux 5.3 et 5.4. Nous avons utilisé un facteur de correction de .25 sur les valeurs de superficie monocouche lorsque celles-ci étaient intolérantes aux défauts au niveau considéré. L'analyse est pour des alimentations indépendantes à partir du troisième niveau CDE (chaque segment reçoit une alimentation neutre indépendante). Les quatre colonnes de droite indiquent les rendements respectifs: 1-un segment de 10 PEs de type IMAGE2, 2-un segment de 34 processeurs IMAGE2p+, 3-une matrice de 34 segments de 32 processeurs IMAGE2p+ fonctionnels et 4- un système MIMD comprenant 7 systèmes SIMD de 1024 PEs fonctionnels. Les deux premières colonnes indiquent: 1-le module considéré et 2-le nombre de

défauts considérés dans le cas de groupes homogènes (section 3.2, équation 6). Dans le cas de la puce IMAGE2, nous avons utilisé $D=7$ défauts/cm², en accord avec le rendement de 85% obtenu pour une puce de dimension B (.125 cm²) sur le même procédé et des carrés de 9 microns². Dans le cas du système de 5*1024 PEs, nous avons utilisé $D=2$ et considéré des carrés de 1 micron² en correspondance avec notre interprétation de l'état actuel de l'art sur le marché international.

TABLE 5.6: RENDEMENTS

module	nombre de défauts	rendement			
		IMAGE2 10 PEs	IMAGE2p+ 34 PEs	34 segments	7 SIMDs
un module		.962	.997	.988	.811
module tol.	0	.679	.903	.663	.231
	1	.268	.092	.270	.378
	2	.048	.004	.053	.265
	somme	.995	.999	.986	.874
module intolérant		.932	.999	.910	.898
module de commande		.866	.990	.904	(test).996
niveau complet		.803	.988	.811	.781
rendement sans HCDE					.000000148
FMR: facteur de mérite de Mangir:		5200000/1.5 =3500000			
FMF: facteur de mérite fonctionnel:		64			

Le rendement de la puce IMAGE2 a été amélioré de .592 à .803. Nous avons prévu que l'amélioration serait de faible importance, le circuit étant de faible complexité: 10 modules de haut rendement. Dans le cas du système IMAGE2p+, nous constatons que le rendement des modules est suffisant pour obtenir un fonctionnement normal des comparateurs

analogiques à chaque niveau CDE reconfigurable. Malgré le choix d'une solution architecturalement simple (segments de 32 processeurs fonctionnels) et sans optimiser les paramètres HCDE, le rendement estimé nous semble acceptable pour une application commerciale.

Considérons maintenant les facteurs de mérite FMR et FMF (section 3.4). Le facteur de mérite de Mangir (FMR) indique le rapport du gain de rendement sur le rapport des superficies; montrant l'avantage lorsque une puce complexe est réalisée. Le rendement sans HCDE étant démesurément faible (.00000015), nous croyons que cette mesure de comparaison est irréaliste en ce qui concerne la solution sans HCDE (rendement médiocre).

Le facteur de mérite fonctionnel est une mesure du gain de fonctionnalité obtenu avec HCDE, pour deux solutions à rendement équivalents. Utilisant comme base le rendement avec HCDE (.781), nous avons estimé le nombre de processeurs produisant un rendement comparable si HCDE n'était pas utilisée et avons obtenu une puce de 80 processeurs IMAGE2p+ (tableau 5.4). Le facteur de mérite fonctionnel devient alors $5120/80 = 64$. L'application de HCDE permet donc de réduire le nombre de puces requises d'un facteur de 64.

5.6-CONCLUSION ET TRAVAUX FUTURS

Nous avons démontré à la section 5.5 que HCDE permet effectivement d'augmenter le rendement de circuits. Nous avons aussi démontré que des gains de rendement successifs sont possibles dans la mesure où l'alimentation des puits des modules peut être programmée de manière à isoler les courts-circuits entre les alimentations. Nous avons montré (section 5.5a) l'avantage d'isoler l'alimentation (aux puits) lorsqu'on veut utiliser CDE sur plusieurs niveaux hiérarchiques successifs. Dans notre exemple cette isolation était avantageuse dès le troisième niveau hiérarchique.

Nous avons montré aux chapitres 4 et 5, au moyen d'un exemple de 5120 processeurs de type IMAGE2p+, qu'il est possible de réaliser un système de traitement primaire d'IMAGES en temps réel sur une seule puce. L'analyse de rendement de la section 5.5 montre qu'il est possible de concevoir cette puce de manière à obtenir un rendement suffisant (.781) pour permettre une application commerciale. Ceci étayé par le fait que l'exemple de système proposé utilise un processeur (IMAGE2p+) plus performant qu'IMAGE2, en quantité correspondant aux besoins conservateurs estimés

dans le cas de processeurs de type IMAGE2. Nous affirmons donc qu'il est possible de réaliser un système de traitement primaire d'images en temps réel sur une seule puce; entendu que les puces de mémoire principale additionnelles requises sont adjointes à la puce processeur. L'utilisation du mode de traitement en lignes, permet de restreindre la bande passante à la mémoire principale sans augmenter indûment la mémoire interne de chaque processeur.

Nous croyons qu'il est important de développer une méthode efficace d'isolation des courts-circuits entre les alimentations. Ceci afin de permettre de profiter de gains successifs de fonctionnalité possibles par l'application de l'aspect hiérarchique de HCDE.

CHAPITRE VI

CONCLUSION

6.1-CONCLUSIONS GENERALES

Nous avons développé une technique permettant d'augmenter la superficie d'une puce afin d'augmenter la puissance (calcul et capacité de mémoire) possible sans sacrifier le rendement. Nous avons présenté une version généralisée de cette technique (HCDE) au second chapitre. La technique s'applique à des circuits: 1-comportant un degré suffisant de régularité, 2-dont les modules réguliers sont de rendement suffisant et 3-dont les modules sont vérifiables. Nous avons ensuite proposé une formulation simple pour l'estimation du rendement afin de montrer l'avantage de la technique HCDE sur le rendement.

Nous avons ensuite mesuré, au moyen d'une implantation réelle (IMAGE2), la complexité et les superficies des circuits complémentaires ajoutés lors de l'application de HCDE. Extrapolant ces mesures, nous avons présenté un exemple d'application de HCDE pour une puce d'une complexité d'environ 25 millions de transistors. Cette complexité a été choisie pour montrer que HCDE peut avancer de deux et peut-être trois générations la complexité des circuits auxquels elle s'applique.

Dans un procédé hypothétique de 1 micron avec une densité de 2 défauts par cm^2 nous avons obtenu, sans optimisation, un rendement de .78, un facteur de mérite de Mangir démesurément élevé (3500000) et un facteur de mérite fonctionnel de 64 pour un circuit d'environ 25 millions de transistors. Ceci signifie que dans le cas de l'exemple cité, l'application de HCDE avec trois niveaux reconfigurables a permis d'augmenter la complexité du circuit d'un facteur de 64 sans en affecter le rendement. Ceci signifie aussi que sans l'utilisation proposée de HCDE, la puce aurait un rendement insuffisant pour permettre une application commerciale à la densité proposée.

Ayant dévoilé une technique permettant de réaliser des circuits complexes de haut rendement, nous encourageons à utiliser HCDE, tous les concepteurs de systèmes désireux de profiter de l'avantage de gain de fonctionnalité. Si un manufacturier utilise une technique d'amélioration du rendement, les autres seront désavantagés au niveau des performances des circuits produits (encombrement, coût, consommation, etc.). Nous concluons que, dans un contexte concurrentiel, l'utilisation de HCDE ou d'une autre technique d'amélioration du rendement est maintenant essentielle pour produire des systèmes de grande complexité

qui soient commercialement concurrentiels.

6.2-AVANTAGES DE HCDE

Cette section donne un aperçu de certains avantages à l'utilisation de HCDE et un aperçu de certaines limites possibles envisagées.

HCDE peut être utilisé pour améliorer le rendement d'un circuit de grande complexité pour en réduire le coût. Ceci peut être reformulé en une opportunité d'augmenter la puissance d'un système, en compactant plusieurs puces en une seule. Ceci permet 1-d'augmenter le nombre de modules par puce; dans notre exemple, nous avons obtenu un gain en performance de deux ordres de grandeur (10^6) par rapport à une solution sans HCDE. Ceci conduit aussi: 2-à une réduction de la superficie allouée au contrôle et aux broches (économie de silicium et de consommation), 3-à une réduction de la capacité des lignes (gain de bande passante du réseau reliant les modules et réduction de la consommation), 4-à une réduction du coût d'un lien entre des modules (augmentation de la flexibilité des réseaux entre les modules) et 5-à une réduction du volume requis pour réaliser un système. Tous ces avantages conduisent à des réductions importantes de coût pour la réalisation d'un

système.

En augmentant le nombre possible de modules par puce, les entrées/sorties des modules sont immédiatement accessibles à un plus grand nombre de modules voisins, sur une même puce. La multiplication du nombre de modules, possible avec HCDE, permet d'augmenter significativement la fonctionnalité des circuits sans provoquer de baisses de rendement.

Dans le cas de systèmes multiprocesseurs homogènes de type MIMD et SIMD, quel que soit le type de processeur utilisé, l'augmentation de puissance de calcul par puce est proportionnelle au nombre de processeurs. Dans le cas de systèmes SIMD, l'augmentation du nombre de modules conduit aussi à une réduction de la superficie allouée aux circuits de commande, qui peuvent alors être partagés. Ceci permet d'augmenter la flexibilité du circuit de commande.

On peut aussi profiter d'une meilleure accessibilité des entrées/sorties des modules voisins alors plus nombreux sur une même puce. Ceci peut conduire à une augmentation significative de la flexibilité d'un système autrement irréalisable (réseau entièrement connecté à accès indirect versus réseau mélange parfait versus réseau fixe). Dans le

cas d'algorithmes limités par les interconnexions interprocesseur, ceci peut conduire à une augmentation du taux d'utilisation de la puissance de calcul disponible, élargissant alors le champ d'application d'un système.

Les circuits complémentaires sont relativement peu encombrants dans l'exemple des chapitres 4 et 5. La superficie augmente d'une fraction plutôt que d'être doublée ou même triplée lorsque des techniques comme la double ou la triple redondance sont utilisées.

HCDE comporte une approche structurée de vérification d'un circuit de grande complexité. La vérification simultanée de tous les modules restreint d'un facteur égal au nombre de modules testés simultanément la longueur des tests de tous les modules identiques. La complexité des tests se limite à vérifier un module pour chaque niveau hiérarchique successif, ne vérifiant à chaque niveau que les noeuds/lignes n'ayant pas été vérifiées aux niveaux inférieurs. Ceci tend à confiner la complexité des tests des niveaux supérieurs au périmètre des modules, évitant alors d'avoir à pénétrer profondément dans les modules aux niveaux supérieurs où l'accessibilité est restreinte.

Les cellules de vérification sont non seulement

vérifiables mais aussi tolérantes aux défauts. L'utilisation de l'aspect hiérarchique conduit à des circuits dont la partie intolérante est restreinte aux circuits complémentaires du niveau hiérarchique supérieur.

La vérification des modules se fait à leur vitesse maximale d'opération plutôt qu'à vitesse réduite. Un défaut affectant la vitesse d'opération du circuit, non détecté par une analyse strictement fonctionnelle sera donc plus facilement détecté quelle que soit l'ensemble de vecteur utilisé à cet effet.

Le circuit complémentaire d'un module est presque entièrement vérifiable pour les fautes bloqué à 1 et 0 et les transitions. Un interrupteur de reconfiguration inutilisé peut être ouvert sans que le fonctionnement du circuit soit affecté. Un court-circuit avec une ligne fonctionnelle peut cependant affecter le fonctionnement du circuit. Nous n'avons pas trouvé de méthode pour la vérification de ce dernier mode de défaut des interrupteurs de reconfiguration.

La couverture de défauts des modules devient un paramètre critique lorsque la complexité d'un circuit augmente. Puisque le troisième critère d'applicabilité

concerne à la vérifiabilité des modules. Quelles sont les conséquences d'une vérifiabilité partielle des modules? Nous nous contentons ici, faute d'espace et de temps, de poser cette question.

6.3-SYSTEME DE TRAITEMENT D'IMAGES EN TEMPS REEL ET ECONOMIQUE

Nous croyons sincèrement avoir démontré la faisabilité d'un système de traitement primaire d'image performant et occupant une seule puce (mémoire principale exclue). Nous avons constaté que le coût d'un système de traitement primaire n'est pas dominé par le processeur mais plutôt par la mémoire principale lorsqu'un traitement multitrame est envisagé.

Nous concluons alors qu'il est nécessaire de restreindre la bande passante entre la mémoire principale et le processeur afin de pouvoir utiliser des puces de mémoire plus denses et plus économiques. Dans le cas du traitement primaire, l'utilisation massive du mode de traitement ligne nous a permis d'harmoniser le flux de traitement avec le flux de données dans le cas des standards envisagés (NTSC, PAL, SECAM). Ce mode de traitement permet de restreindre efficacement la dimension de la mémoire principale ainsi que

sa bande passante.

A cause du coût des composants mémoire de grande capacité, (64 puces de 1 Moctets), un système de traitement primaire en temps réel coûterait environ \$4000 de composants et comporterait environ 65 puces (64 mémoire à \$4000 et 1 processeur à environ \$75). Le faible coût du processeur de traitement primaire et la bande passante restreinte au niveau tertiaire, nous encourage à supposer que le reste d'un système de vision artificielle ne requerrait pas une dépense supérieure à celle requise pour le traitement primaire. Nous estimons donc qu'un système de vision artificielle, jusqu'à l'analyse de scènes, pourrait n'occuper qu'une seule carte et coûter entre \$4000 et \$10000 de composants.

BIBLIOGRAPHIE

"[1] Adit D. Singh, An Area Efficient Redundancy Scheme for Wafer Scale Processor Arrays, IEEE, ICCD85, Computer Society Press of the IEEE, Massachusetts ave., N.W., Washington, 1985."

"[2] C. Cyr, Y. Savaria, D. Audet et J. L. Houle, A Novel Self-Testing and Reconfiguration Scheme for Two Dimensional Logic Arrays, ICCD87, Computer Society Press of the IEEE, Massachusetts ave., N.W., Washington, 1987."

"[3] Carver Mead et Lynn Conway, Introduction to VLSI Systems (2nd Edition), Addison-Wesley Publishing Company, Reading, Massachusetts, 1980."

"[4] Neil H. E. Weste et Kamran Eshraghian, Principles of CMOS VLSI Design (1st Edition), Addison-Wesley Publishing Company, Reading, Massachusetts, 1985."

"[5] Yvon Savaria, Conception et Vérification des Circuits VLSI (1ère Edition), Editions de l'École Polytechnique de Montréal, Montréal, 1988."

"[6] Israel Koren and Dhiraj K. Pradhan, Introducing Redundancy into VLSI Designs for Yield and Performance Enhancement, IEEE, Fault-Tolerant Computer Symposium, IEEE Computer Society Press, Los Angeles, CA, 1985."

"[7] P. Schvan, R. Hadaway, and M.King, Defectivity and Yield Analysis for VLSI and WSI, IEEE ICCD86, Port Chester, New-York, October 1986."

"[8] Claude Thibeault, Yvon Savaria et Jean-Louis Houle, A New Yield Formula for Fault-Tolerant Large-Area Devices, IEEE International Workshop on Defect and Fault Tolerance in WSI Systems, Springfield, N.Y., 1988."

"[9] Mangir T., Sources of Failures and Yield Improvement for VLSI and Restructurable Interconnects for RVLSI and WSI: Part I - Sources of Failures and Yield Improvement for VLSI, Proc. IEEE, vol.72, no.6, juin 84."

"[10] William K. Pratt, Digital Image Processing (1st Edition), Wiley Interscience, John Wiley Sons, New York, New York, 1978."

"[11] Raphael C. Gonzalez et Paul Wintz, Digital Image Processing (6th Edition), Addison-Wesley Publishing Company, Reading, Massachusetts, 1977."

"[12] Dana H. Ballard et Christopher M. Brown, Computer Vision (1st Edition), Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982."

"[13] Azriel Rosenfeld et Avinash C. Kak, Digital Picture Processing (1st Edition), Academic Press, Orlando, Florida, 1982."

"[14] Ernest L. Hall, Computer Image Processing and Recognition (1st Edition), Academic Press, New York, New York, 1979."

"[15] Avron Barr et Edward A. Feigenbaum, The Handbook of Artificial Intelligence (1st Edition), William Kaufmann, Inc., Los Altos, California, 1981."

"[16] Rama Chellappa et Alexander A. Sawchuk, Digital Image Processing and Analysis (1st Edition), IEEE Computer Society #665, Los Angeles, California, 1985."

"[17] J.K. Agarwal, R.O. Duda, A. Rosenfeld, Computer Methods in Image Analysis (1st Edition), IEEE Press Reprint Series, New York, New York, 1977."

"[18] Théo Pavlidis, Algorithms for Graphics and Image Processing (1st Edition), Computer Science Press, Rockville, Maryland, 1982."

"[19] Ramakant Nevatia, Machine Perception (1st Edition), Prentice Hall, Englewood Cliffs, New Jersey, 1982."

"[20] N. S. Jayant et Peter Noll, Digital Coding of Waveforms (1st Edition), Prentice Hall, Englewood Cliffs, New Jersey, 1984."

"[21] Donald G. Fink et Alexander A. McKenzie, Electronics Engineer Handbook (1st Edition), McGraw-Hill Book Company, New York, New York, 1975."

"[22] Ron Riesenbach, Ran Ginosar et Alfred Bruckstein, A VLSI Architecture for Real Time Image Processing, ICCD86, Computer Society Press of the IEEE, Massachusetts ave., N.W., Washington, 1986."

"[23] Edward J. McCluskey, Verification Testing - A Pseudoexhaustive Test Technique, IEEE Transactions on Computer Science, Vol. C33, No.6, June 1984."

"[24] P. Goel, B.C. Rosales, Podem-X: An Automatic Test Generation System for VLSI Logic Structures, 18th Design Automation Conference, 1981."

"[25] W. R. Moore, Yield-Enhancement of a Large Systolic Array Chip, Microelectronics, Reliability; Vol.24, No.3, Great Britain, 1984."

"[26] V. K. Agarwal, E. Cerny, N. C. Rumin, Y. Savaria, and P. Schvan; Techniques for Implementing Large Area Devices, IEEE 1986 International Conference on Computer Design, Port Chester, New-York, October 1986."

"[27] Daniel Audet, Gilles Chouinard, Claude Cyr, Jean-Louis Houle et Yvon Savaria, IMAGE2, un Circuit Multiprocesseur pour le Traitement Parallèle, Rapport Technique EPM/RT-88-35, Ecole Polytechnique de Montréal, 1988."

"[28] C. Thibeault, Y. Savaria et J. L. Houle, IMAGE2M, a VLSI Implementation of a Multi-Processor Chip for Morphological Image Processing", Proc. 5th Canadian Conference on VLSI, Winnipeg, Canada, october 1987.

ÉCOLE POLYTECHNIQUE DE MONTRÉAL



3 9334 00290811 7