

ACADEMIE DE MONTPELLIER

UNIVERSITE MONTPELLIER II

SCIENCES ET TECHNIQUES DU LANGUEDOC

THESE

présentée à l'Université de Montpellier II Sciences et Techniques du Languedoc
pour obtenir le **DIPLOME DE DOCTORAT**

Spécialité : **Génie Informatique, Automatique et Traitement du Signal**

Formation Doctorale : **Systèmes Automatiques et Microélectroniques**

École Doctorale : **Information, Structures, Systèmes**

REDUCTION DE PUISSANCE DURANT LE TEST PAR SCAN DES CIRCUITS INTEGRES

par

Nabil BADEREDDINE

Soutenue le 15 septembre 2006, devant le Jury composé de :

M. Patrick GIRARD, Directeur de Recherche CNRS, Université Montpellier II	Directeur de Thèse
M. Serge PRAVOSSOUDOVITCH, Professeur, Université Montpellier II	Examinateur
M. Olivier SENTIEYS, Professeur, ENSSAT Lannion, Université Rennes 1	Rapporteur
M. Matteo SONZA REORDA, Professeur, Politecnico di Torino (Italie)	Rapporteur
M. Arnaud VIRAZEL, Maître de conférences, Université Montpellier II	Examinateur
M. Jean-Marc VRIGNAUD, Aerospace Product Engineer, ATMEL	Examinateur

A ma fille, à ma femme et à mes très chers parents

Remerciements

Je tiens à remercier en premier lieu Monsieur Michel ROBERT, professeur à l'université de Montpellier II et directeur du Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier, de m'avoir accueilli au sein de son laboratoire.

Je remercie Monsieur Patrick GIRARD, Directeur de recherche au CNRS, sous la direction duquel ces travaux ont été effectués, pour son aide, sa sympathie, son expérience dont il m'a fait profiter et la confiance qu'il m'a accordée tout au long de ces travaux.

Je tiens à exprimer ma gratitude à Messieurs Arnaud VIRAZEL, maître de conférences à l'université de Montpellier II, Serge PRAVOSSOUDOVITCH, professeur à l'université de Montpellier II et Christian LANDRAULT, Directeur de recherche au CNRS, pour leurs nombreux conseils, leur aide et leur disponibilité tout au long de ces trois années de thèse.

J'adresse également mes remerciements à Monsieur Hans-Joachim WUNDERLICH, professeur à l'université de Stuttgart en Allemagne et Monsieur Krishnendu CHAKRABARTY, professeur à l'université de Duke au Etats-Unis, pour leur participation à cette étude.

Je remercie aussi Monsieur Jean-Marc VRIGNAUD, ingénieur en production aérospatiale et expert en testabilité chez ATMEL, d'avoir accepté d'être membre de mon jury de thèse. Je voudrais également remercier Monsieur Olivier SENTIEYS, Professeur à l'université de Rennes I et Monsieur Matteo SONZA REORDA, Professeur à Politecnico di Torino en Italie, qui m'ont fait l'honneur d'accepter d'être rapporteurs et membres de mon jury de thèse.

Remerciements

Une pensée amicale pour tous les thésards croisés durant ces quatre années passées au LIRMM. En particulier pour Robin, Alex³, Razaf, Nico, Fab, Abou, Olive, Vincent, Luigi et Reou.

J'adresse mes remerciements à tous mes amis, les montpelliérains : Samer, Hassan, Rafik, Abdellah, Meziane, Lounis, Alilou, Safir, Nizar et Naoufel, les parisiens : Salim, Mouloud, Redouane et Riad, les grenoblois : Malek, Maher et Riad, Kader le strasbourgeois, Taoui le nantais, Amine l'américain ainsi que Mehdi l'albigeois et Mounir le lannionai, pour leur soutien durant ces quatre années passées en France. Je souhaiterais plus particulièrement remercier Akram pour tout le temps qu'on a passé ensemble à Montpellier et surtout pour son côté flou !

Je voudrais aussi remercier mes chers parents, ainsi que mes frères et sœurs, qui m'ont toujours laissé le choix de mes études et qui m'ont soutenu durant toute ma vie. Je tiens à remercier ma belle famille, et plus particulièrement mes beaux parents, pour leur soutien et leur encouragement durant ces quatre dernières années. Je remercie affectueusement Mina, ma femme, pour son soutien réconfortant, sa présence attentionnée, sa gentillesse et sa compréhension.

Enfin, je remercie le tout puissant sans qui je ne serai pas là.

Shoukrane à tous.

Sommaire

Sommaire

Introduction Générale	15
Chapitre I : Problématique et état de l'art	21
1.1 Introduction	21
1.2 Contexte de l'étude : le Test par SCAN	22
1.3 Modélisation de la puissance dans les circuits CMOS	25
1.4 Consommation de puissance lors du test	28
1.4.1 Origines de la consommation de puissance lors du test	28
1.4.2 Conséquences d'une consommation de puissance excessive	29
1.4.3 Etat de l'art sur la réduction de puissance durant le test	30
1.4.3.1 Modification de la procédure de test	30
1.4.3.2 Modification du circuit sous test	31
1.4.3.3 Modification de la séquence de test	33
1.5 Analyse de la consommation de puissance de pic pendant le cycle de test	35
1.5.1 Estimation de la puissance de pic pendant le test par scan	38
1.5.2 Optimisation de la puissance de pic	41
1.6 Conclusion	42
Chapitre II : Réduction de puissance par réordonnement des cellules scan	45
2.1 Introduction	45
2.2 Impact de l'ordre des cellules scan sur la consommation de puissance	46
2.3 Optimisation globale sans contrainte	48
2.3.1 Modélisation du problème d'optimisation du chaînage des cellules scan	49
2.3.2 Algorithme glouton	51
2.3.3 Résultats expérimentaux	52
2.4 Optimisation globale sous contrainte	54

2.4.1	Algorithme de Recuit Simulé _____	55
2.4.2	Résultats expérimentaux _____	57
2.4.2.1	Réduction de puissance de pic pendant le cycle de test _____	58
2.4.2.2	Réduction de puissance de pic pendant les cycles de décalage _____	59
2.4.2.3	Couverture de fautes de transition _____	60
2.5	Conclusion _____	63
Chapitre III : Réduction de puissance par assignation de bits non spécifiés _____		67
3.1	Introduction _____	67
3.2	Impact de l'assignation de bits non spécifiés sur la consommation de puissance lors du test _____	68
3.3	Techniques d'assignation de bits non spécifiés _____	70
3.3.1	Techniques classiques d'assignation _____	70
3.3.2	Résultats expérimentaux _____	72
3.3.3	Assignation basée sur des informations structurelles _____	75
3.3.4	Résultats expérimentaux _____	79
3.3.4.1	Réduction de puissance de pic pendant le cycle de test _____	80
3.3.4.2	Réduction de puissance de pic pendant les cycles de décalage _____	82
3.3.4.3	Impact sur le volume de données de test _____	82
3.3.4.4	Impact sur le taux de couverture _____	83
3.4	Conclusion _____	86
Chapitre IV : Réduction de puissance et compression de données de test _____		91
4.1	Introduction _____	91
4.2	Compression de données de test _____	92
4.3	Réduction de puissance et compression de données de test _____	94
4.3.1	Codage de Golomb et réordonnancement des vecteurs de test [CHAN01] ____	95
4.3.2	Illinois Scan Architecture [HAMZ99] _____	96
4.3.3	Random Access Scan [ANDO80] _____	97
4.3.4	Bilan et objectifs _____	98
4.4	Solution proposée pour la compression de données et la réduction de puissance _	99
4.4.1	Compression de données par codage sélectif des "Scan Slices" _____	99
4.4.2	Analyse de la consommation de puissance après la décompression des scan slices _____	103
4.4.3	Modification proposée _____	106

4.4.4	Résultats expérimentaux _____	107
4.4.4.1	Réduction du volume de données de test _____	107
4.4.4.2	Réduction de consommation de puissance _____	108
4.4.4.3	Résultats sur circuits industriels _____	110
4.5	Conclusion _____	113
	Conclusion Générale _____	117
	Références Bibliographiques _____	121
	Listes des figures, tableaux et organigrammes _____	129
	Annexe A : Estimation de la consommation de puissance par la métrique de transitions pondérées [SANK00] _____	133

Introduction Générale

Introduction Générale

A l'issue du processus de conception, la fonctionnalité d'un circuit intégré est vérifiée par des outils logiciels de vérification et de simulation ou à l'aide de prototypes. Après l'élimination (supposée complète) de toutes les erreurs de conception, le bon fonctionnement du circuit dépend alors essentiellement de la qualité du procédé technologique de fabrication. Pour s'assurer de son bon fonctionnement après fabrication, il est alors nécessaire de tester le circuit à l'aide de techniques et d'outils de test adaptés.

Compte tenu de la complexité des circuits intégrés actuels, des techniques de conception en vue du test sont couramment utilisées pour améliorer la testabilité d'un circuit intégré. Le test par scan est la technique de conception en vue du test la plus utilisée à l'heure actuelle dans l'industrie [BUSH00]. Cette technique consiste à modifier les bascules du circuit afin de les rendre directement contrôlables et observables durant la phase de test. Cette technique, bien qu'offrant de nombreux avantages, notamment en terme d'efficacité, présente quelques inconvénients tels que son coût en surface ou le temps de test important qu'elle induit.

Un autre problème qui peut apparaître lors du test par scan est lié à la dissipation de puissance, qui peut être supérieure à celle mise en jeu lors du fonctionnement normal du circuit [ZORI93]. Cette consommation excessive lors du test provient du nombre important de commutations qui sont générées lors du chargement et du déchargement des données de test dans la (ou les) chaîne(s) de scan. Des expérimentations réalisées sur des circuits industriels ont montré que cette consommation lors du test pouvait être de 200 à 300% supérieure à celle mise en jeu lors du fonctionnement normal du circuit [SAXE03]. Les conséquences de cette consommation excessive peuvent être de diverses natures [GIRA02] : perte d'informations logiques au cours du test entraînant des erreurs d'interprétation du résultat de test (un circuit

sain pourra être déclaré défectueux, ou vice-versa), diminution de la fiabilité à long terme du circuit, destruction pure et simple, etc.

Plusieurs études ont été présentées dans la littérature concernant des techniques de réduction de la consommation de puissance lors du test par scan. Ces techniques proposent de réduire la puissance moyenne et/ou la puissance de pic consommée pendant le chargement /déchargement des données de test. Cependant, aucune de ces techniques ne cible la réduction de la consommation de puissance qui apparaît entre l'application du vecteur de test et la capture de la réponse dans la chaîne de scan (cette phase est appelée *Cycle de Test*). En effet, une consommation de puissance élevée pendant ce cycle de test peut générer des phénomènes de bruit ou de chute de tension dans le circuit sous test. Associés à une fréquence de test élevée, aujourd'hui nécessaire à la détection de fautes de délai (on parle alors de test at-speed) [BUTL04, CSHI04], ces phénomènes peuvent altérer la capture de la réponse du circuit à chaque vecteur de test. Dans ce cas, un circuit sain pourra être déclaré fautif, réduisant alors le rendement de production. De la même manière, un circuit fautif pourra être déclaré sain, rendant alors le test non robuste. Plusieurs études industrielles récentes [YOSH03, SAXE03, XWEN05] ont démontré l'importance de ces phénomènes de bruit et l'impact sur l'efficacité et la fiabilité du test.

Dans le cadre de cette thèse, nous proposons plusieurs solutions permettant de faire face aux problèmes de consommation de puissance excessive lors du test par scan des circuits intégrés. Cette thèse est divisée en quatre chapitres.

Le premier chapitre décrit le problème de la consommation de puissance durant le test des circuits intégrés et plus particulièrement dans le cas des circuits munis de chaînes de scan. La première partie de ce chapitre présente le contexte de l'étude. La deuxième partie détaille l'origine de la consommation de puissance dans ce type de circuits. La troisième partie détaille les causes et les conséquences d'une consommation de puissance excessive durant le test par scan. La quatrième partie est consacrée à un état de l'art des techniques proposées jusqu'à présent pour résoudre ces problèmes. Enfin, dans une dernière partie, nous précisons le problème auquel nous nous sommes intéressés, et qui concerne la consommation de puissance de pic excessive durant le cycle de test du processus de scan.

Le second chapitre présente une première technique permettant de réduire la puissance dissipée pendant le cycle de test, en jouant sur le réordonnement des cellules scan. Cette technique détermine un ordre optimal des cellules dans une chaîne de scan permettant de réduire l'activité de commutation pendant le cycle de test. La première partie de ce chapitre

montre l'influence de l'ordre des cellules sur l'activité de commutation dans une chaîne de scan, ainsi que le problème d'optimisation à considérer. Les autres parties sont dédiées à la présentation des heuristiques d'ordonnement développées, ainsi qu'à la présentation des résultats expérimentaux obtenus.

Dans le troisième chapitre, nous nous intéressons aux mécanismes d'adaptation des séquences de test permettant de réduire la consommation de puissance lors du test. Pour cela, nous proposons plusieurs techniques d'assignation des bits non spécifiés (bits à 'X') des séquences de test. Ces techniques consistent à spécifier les vecteurs de test en affectant des valeurs logiques aux bits non spécifiés, de manière à optimiser l'impact de ces assignations sur la consommation de puissance lors du test. La première partie de ce chapitre montre l'impact sur la consommation de puissance de l'affectation des bits non spécifiés. Dans la deuxième partie, nous présentons les techniques classiques d'assignation ainsi que les résultats correspondants. Afin d'améliorer les résultats obtenus avec ces techniques, nous présentons dans la troisième partie la technique que nous avons développée pour permettre l'assignation de bits non spécifiés en se basant sur des informations structurelles. Dans la dernière partie, un bilan général des résultats obtenus permet de montrer l'efficacité de la technique proposée en terme de réduction de la consommation de puissance.

Afin de tenir compte de la nature des circuits logiques actuels, qui intègrent souvent des structures de compression/décompression de données de test, nous avons également proposé une technique combinant la compression de données et la réduction de puissance pendant le test. Cette technique est présentée dans le dernier chapitre. Les deux premières parties de ce chapitre sont consacrées à la présentation du principe de la compression de données de test et à un état de l'art sur les techniques combinant la compression de données et la réduction de puissance de lors du test. La troisième partie présente la technique de compression proposée dans [WANG05] à partir de laquelle nous avons travaillé pour combiner réduction de puissance et compression de données. Dans la dernière partie, nous détaillons la technique proposée et nous présentons les résultats expérimentaux obtenus.

Notons que ces travaux de thèse ont été menés dans le cadre du projet européen A503 MEDEA+ ASSOCIATE portant sur la conception en vue du test et le développement d'outils logiciels de test pour les systèmes sur puce. Notons également qu'ils ont donné lieu à deux collaborations internationales, l'une avec le Professeur Hans-Joachim WUNDERLICH, de l'Université de Stuttgart en Allemagne, et l'autre avec le Professeur Krishnendu CHAKRABARTY, de l'Université de Duke aux Etats-Unis.

Chapitre I

Problématique et état de l'art

Chapitre I : Problématique et état de l'art

1.1 Introduction

Dans ce chapitre, nous allons aborder le problème de la consommation de puissance durant le test des circuits intégrés, et plus particulièrement dans le cas des circuits munis de chaînes de scan.

La première partie de ce chapitre présente le principe du test par scan (appelé également test série). Les deuxième et troisième parties sont dédiées à l'analyse de la consommation de puissance dans les circuits CMOS, ainsi qu'aux causes et conséquences d'une consommation de puissance excessive durant le test par scan. La quatrième partie propose un état de l'art des études relatives à ce problème. Enfin, dans une dernière partie, nous précisons le problème auquel nous nous sommes intéressés, et qui concerne la consommation de puissance de pic excessive durant les cycles de test du processus de scan.

1.2 Contexte de l'étude : le Test par SCAN

Grâce à l'amélioration constante des techniques d'intégration, les circuits intégrés actuels peuvent réaliser des fonctions très complexes faisant intervenir plusieurs millions de transistors. Cependant, à cause de cette complexité, des problèmes divers peuvent apparaître lors de la conception ou de la fabrication de tels circuits. Le rôle du test est de s'assurer que les circuits ainsi élaborés ne présentent pas de dysfonctionnement. La validité et l'efficacité du test sont des éléments primordiaux qui permettent de garantir l'obtention de produits de qualité à un coût minimum.

Ce processus de test peut être envisagé de diverses manières qui sont d'ailleurs généralement complémentaires.

Lorsqu'un circuit est conçu et réalisé pour la première fois, il est nécessaire de vérifier la validité de la conception. Ce type de test est appelé *test de caractérisation* (ou test de vérification) et est réalisé avant l'envoi du circuit en production. Des tests fonctionnels sont d'abord appliqués sur un échantillon de circuits, et des mesures paramétriques (mesures de courant, de tension, de temps de propagation, ...) sont ensuite réalisées à l'aide d'outils spécifiques (microscopie électronique ou acoustique, points d'observation, ...).

Le *test de production*, réalisé à l'aide d'Equipements Automatiques de Test (Automatic Test Equipment ATE) sur l'ensemble des circuits issus des chaînes de fonderie, assure quand à lui la détection de défaillances physiques modifiant le comportement du circuit. Ces équipements vérifient que pour une série de stimuli appliquée en entrée du circuit, la réponse correspondante est correcte. Si le comportement du circuit est incorrect, il est éliminé. Cette thèse s'inscrit dans le cadre du test de production.

Le problème principal du test de production est de déterminer une séquence de vecteurs d'entrée applicable dans des temps raisonnables et susceptible de mettre en évidence toutes les défaillances pouvant affecter le circuit. Deux approches principales peuvent être utilisées pour générer la séquence de vecteurs à appliquer aux entrées du circuit : la génération fonctionnelle ou la génération structurelle.

La *génération fonctionnelle* consiste à déterminer une séquence de test en s'appuyant sur la fonctionnalité du circuit. La *génération structurelle* consiste à déterminer une séquence de test en s'appuyant sur une analyse de la structure interne du circuit et sur un modèle de fautes choisi, le plus couramment utilisé étant le modèle de collage. Ce modèle prend en

compte l'hypothèse de collages permanents à '0' ou à '1' des différentes lignes du circuit. La liste de fautes étant élaborée, les vecteurs de test détectant ces fautes sont déterminés par des algorithmes spécifiques [ROTH66, GOEL81, FUJI85 ...]. La métrique utilisée pour évaluer la pertinence d'une séquence de test ainsi générée est le taux de couverture de fautes. Le taux de couverture de fautes est le rapport entre le nombre de fautes détectées par une séquence de test et le nombre total de fautes possibles dans le circuit (dans le modèle considéré).

Déterminer une séquence de test couvrant l'intégralité des fautes pouvant affecter le circuit est, dans le cas général, un problème extrêmement difficile. D'une part, la structure interne des circuits est difficilement accessible compte tenu du faible nombre d'entrées par rapport au nombre de nœuds internes des circuits actuels. D'autre part, la nature des circuits logiques, qui sont généralement des circuits séquentiels, augmente encore la difficulté de contrôle et/ou d'observation de certains nœuds internes du circuit.

La conception en vue du test, et notamment l'insertion de chaînes de scan [WILL73, EICH91], est une solution permettant de simplifier le problème de la génération de vecteurs de test pour les circuits séquentiels. Cette technique est largement utilisée dans l'industrie, puisque près de 90% des circuits intégrés vendus dans le monde possèdent une ou plusieurs chaînes de scan [SIAS05].

Le test par scan permet en effet d'isoler les blocs combinatoires du circuit, de façon à les tester en tant que tels. En utilisant cette technique, le test d'un circuit séquentiel se ramène au test de sa partie combinatoire, ce qui réduit considérablement l'effort de génération des vecteurs de test. Pour cela, la solution consiste à modifier les bascules du circuit en bascule scan et à les relier entre elles de manière à établir un accès série (registre à décalage) appelé chaîne de scan. Un multiplexeur, placé avant l'entrée de chaque bascule permet d'obtenir deux modes de fonctionnement pour la chaîne de scan. Le premier mode de fonctionnement est le mode chargement parallèle, ou mode normal, où les données chargées dans les bascules proviennent du circuit proprement dit. Dans ce cas, la modification des bascules est transparente et le circuit conserve sa fonctionnalité initiale. Le second mode est le mode décalage, ou mode test, où les données chargées dans les bascules proviennent des bascules précédentes. La chaîne de scan est alors comparable à un registre à décalage permettant d'acheminer les vecteurs de test en série à travers les bascules. Les réponses du circuit sont capturées dans le mode de fonctionnement normal et déchargées en série en utilisant le mode test.

La Figure I.1 décrit la modification apportée à une bascule et la façon dont celle-ci s'insère dans le circuit. Les bascules scan sont connectées entre elles pour former une ou plusieurs chaînes de scan. Cette chaîne de scan possède une entrée *scan_in* et une sortie *scan_out*. Le signal "mode de fonctionnement" permet de choisir le mode de fonctionnement de la chaîne de scan.

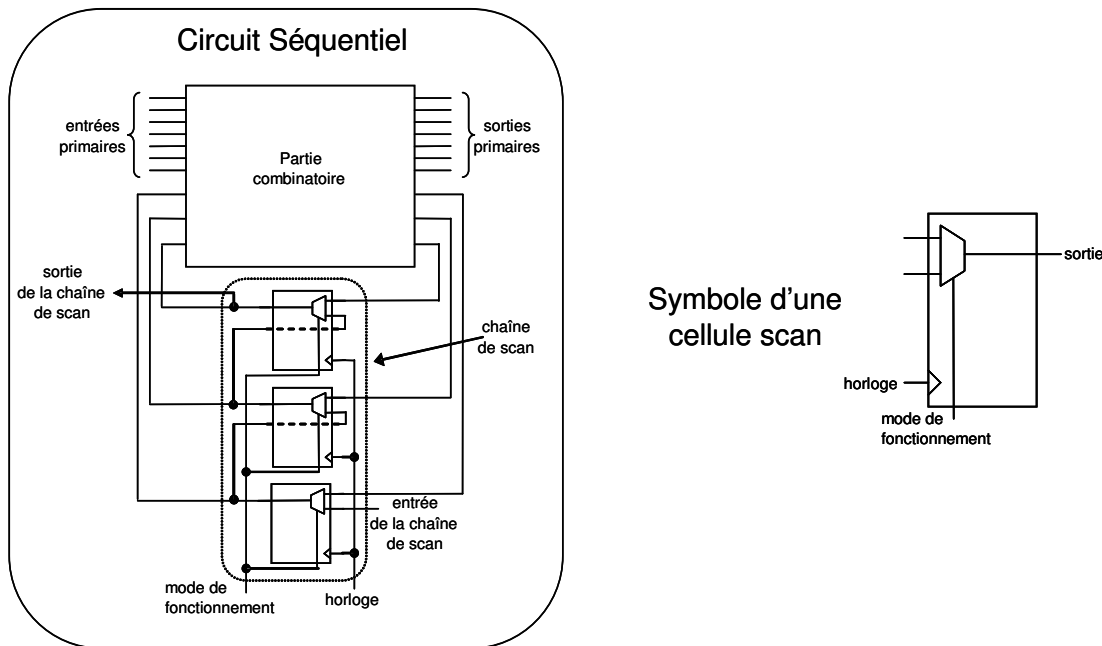


Figure I.1 Description d'un circuit séquentiel muni d'une chaîne de scan

Le test par scan permet ainsi de réduire l'effort de test en rendant directement accessible la partie combinatoire du circuit. Cependant, il présente les inconvénients suivants :

- ✓ Une dégradation des performances temporelles du circuit due à l'ajout du multiplexeur de sélection,
- ✓ Une surface additionnelle toujours due à l'ajout du multiplexeur de sélection et au routage de la chaîne de scan.
- ✓ Une augmentation du temps de test proportionnelle à la taille de la chaîne de scan, car les vecteurs de test sont chargés en série,
- ✓ L'ajout de broches d'entrée et de sortie dédiées au test (signaux de contrôle, entrée et sortie scan).

Pour réduire certains de ces inconvénients, des techniques de scan partiel ou incomplet peuvent être utilisées. Ces techniques consistent à sélectionner les bascules qui vont être transformées en cellules scan, de façon à optimiser les différents critères en établissant un

compromis. Quoiqu'il en soit et malgré tous ces inconvénients, le test par scan reste la technique de conception en vue du test la plus utilisée, car permettant d'obtenir des taux de couverture proches de 100%. C'est aussi une technique fiable, éprouvée, standardisée [POEH03] et qui s'insère facilement dans un flot de conception.

1.3 Modélisation de la puissance dans les circuits CMOS

Les travaux développés dans cette thèse ayant pour objectif de réduire la puissance dissipée dans le circuit durant le test par scan, nous allons nous attacher ici à définir les origines de la puissance dissipée dans les circuits CMOS. Trois composantes de puissance sont généralement considérées dans les circuits CMOS : la puissance statique, la puissance dynamique interne et la puissance dynamique externe.

La *puissance statique* correspond à la puissance consommée quand le circuit est au repos (sans aucune activité de commutation). Dans un circuit idéal, elle devrait être nulle. En réalité, ce n'est pas le cas, à cause des courants de fuite tels que le courant inverse des diodes au niveau du drain des transistors ou le courant de fuite dans le canal des transistors faiblement bloqués. La puissance statique ne représente qu'une faible partie de la puissance totale dissipée, même si dans les technologies submicroniques, elle devient de plus en plus importante [KLEI05].

La *puissance dynamique interne* ou puissance de court-circuit découle du fait que les fronts montant ou descendant des signaux ne sont pas idéaux dans le sens où les commutations ne sont pas instantanées. Ces commutations entraînent par conséquent, et ce pendant un court instant (ce temps dépend de la pente des signaux), la conduction simultanée du plan P et du plan N des éléments de base CMOS, entraînant un chemin de conduction entre la masse et l'alimentation (et par conséquent une consommation de puissance). Cette puissance dissipée dans le circuit représente une faible partie de la puissance dynamique dissipée [NGUY03].

La *puissance dynamique externe* est due à la charge et à la décharge des différentes capacités dans le circuit. Elle intervient également lors de la commutation des portes composant le circuit. La puissance dynamique externe représente la part la plus importante de la puissance dynamique consommée dans un circuit [NGUY03].

En résumé, la puissance statique dépend du procédé technologique d'intégration. Elle est pratiquement invariante lors du test. La puissance dynamique dépend des commutations

des différentes portes du circuit et de la forme des signaux appliqués aux entrées du circuit. Lors du test, cette source de puissance varie en fonction de l'activité de commutation dans le circuit. La puissance dynamique externe jouant un rôle prépondérant dans la dissipation de puissance lors du test, nous allons par la suite nous intéresser plus en détail à l'expression de cette puissance.

Considérons l'inverseur représenté sur la Figure I.2. La capacité de sortie C_i du nœud i est composée des capacités d'entrée des transistors de la (ou des) porte(s) attachée(s) par ce nœud, des capacités d'interconnexion et des capacités de diffusion de drain des transistors. L'énergie consommée dans ce cas est due à un chargement (Figure I.2.a) ou à un déchargement (Figure I.2.b) de la capacité équivalente de sortie C_i . L'expression de cette énergie est [CIRI87] :

$$E_i = \frac{1}{2} V_{dd}^2 \times C_i$$

Pour avoir une bonne approximation de l'énergie consommée au nœud i pendant un intervalle de temps t , l'expression de cette énergie devient :

$$E_i = \frac{1}{2} \times V_{dd}^2 \times N_i \times C_i$$

où N_i représente le facteur d'activité au nœud i (nombre moyen de commutations pendant l'intervalle de temps t).

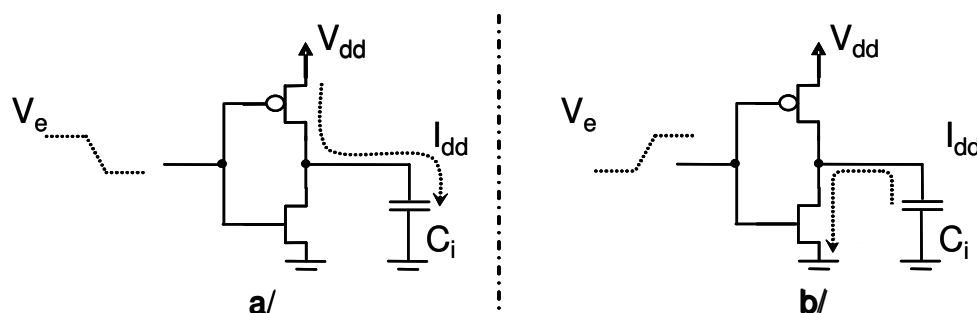


Figure I.2 Charge et décharge de la capacité de sortie d'un inverseur

D'après la formulation précédente, l'énergie consommée sur l'ensemble du circuit après l'application d'une paire de vecteurs d'entrée successifs (V_{k-1} , V_k) s'exprime selon la formule suivante :

$$E_{V_k} = \frac{1}{2} \times V_{dd}^2 \times \sum_i N_i(k) \cdot C_i$$

où $\sum_i N_i(k)$ est le nombre de commutations provoquées dans le circuit au nœud i lors de l'application du vecteur V_k après celle du vecteur V_{k-1} .

Si maintenant on considère une séquence complète de vecteurs consécutifs, l'énergie dynamique totale consommée dans le circuit s'exprime de la manière suivante :

$$E_{Total} = \frac{1}{2} \times V_{dd}^2 \times \sum_k \sum_i N_i(k) \cdot C_i$$

où $\sum_k \sum_i N_i(k)$ est le nombre de commutations produites durant l'application d'une séquence complète de k vecteurs.

La puissance dynamique dissipée correspond à l'énergie dissipée divisée par le temps. Ainsi, la puissance dynamique dissipée pendant un temps t (t est un laps de temps très court, généralement celui qui suit le front d'horloge du système), appelée puissance instantanée (P_{Inst}), est donnée par la formule suivante :

$$P_{Inst.} = \frac{E_{V_k}}{t}$$

La puissance de pic (P_{Pic}) est la valeur maximale de la puissance instantanée. Ce paramètre peut également s'évaluer comme étant la plus haute consommation d'énergie observée durant le temps t , divisée par t . La puissance de pic peut s'exprimer de la façon suivante :

$$P_{Pic} = \max_k (P_{Inst.}(V_k)) = \max_k \left(\frac{E_{V_k}}{t} \right)$$

La puissance moyenne (P_{Moy}) consommée durant la session de test, qui correspond à l'énergie totale divisée par le temps total d'application de la séquence complète de vecteurs, s'exprime de la manière suivante :

$$P_{Moy} = \frac{E_{Total}}{T_{Total}} = \frac{E_{Total}}{L_{Test} \cdot T}$$

où T_{Total} représente le temps total d'application de la séquence complète de vecteurs, qui est égal à la longueur totale de la séquence de test L_{Test} multipliée par la période de l'horloge.

A partir de ces relations, nous pouvons noter que l'activité de commutation dans les circuits est un paramètre commun à l'énergie, la puissance moyenne et la puissance de pic. En conséquence, si l'on désire optimiser la puissance dynamique externe dissipée dans le circuit,

une des solutions possibles est de minimiser l'activité de commutation dans le circuit. L'ensemble des techniques décrites dans ce manuscrit a pour objectif de réduire cette activité de commutation durant le test par scan.

1.4 Consommation de puissance lors du test

Un circuit est conçu au départ pour répondre aux différentes contraintes de fonctionnement établies lors de sa conception (fréquence de fonctionnement, puissance maximale, limites thermiques des boîtiers, ...etc.). Toutes ces spécifications ont été établies pour un fonctionnement normal du circuit. Mais lors du test, une consommation de puissance excessive peut conduire à une violation de ces spécifications (dépassement de limites). Dans [ZORI93], l'auteur a mis en évidence les problèmes engendrés par la surconsommation de puissance au moment du test. Il montre que la puissance dissipée est considérablement plus élevée durant le mode test qu'en mode de fonctionnement normal du circuit. D'autres études plus récentes ont montré que cette augmentation pouvait être de l'ordre de 200% à 300% [BUSH00, SAXE03, CSHI04]. Cette surconsommation de puissance peut avoir des conséquences diverses sur le circuit, parmi lesquelles une diminution de sa fiabilité ou un risque de destruction pur et simple. Dans cette partie, nous allons détailler l'origine de cette surconsommation durant le test et ses conséquences sur le circuit.

1.4.1 Origines de la consommation de puissance lors du test

La consommation de puissance est plus importante pendant le mode test que pendant le mode de fonctionnement normal d'un circuit intégré. Deux raisons principales peuvent expliquer cette surconsommation de puissance. D'une part, il y a un conflit d'objectif entre la fonctionnalité, où le taux d'activité doit être le plus bas possible (notamment pour les applications "Low Power") et le test, où le taux d'activité doit être le plus élevé possible afin d'obtenir la meilleure qualité de test. D'autre part, il y a une faible corrélation entre les données de test, alors que les données fonctionnelles sont souvent fortement corrélées.

Ce problème de consommation de puissance excessive est encore plus présent lors du test par scan, car chaque vecteur (réponse) de test nécessite un grand nombre de cycles de décalage pour être chargé (déchargée), ce qui se traduit par une forte activité de commutation dans la chaîne de scan, et donc dans le circuit sous test [BUSH00]. L'étude industrielle présentée dans [SAXE03] a montré que pour un circuit ASIC constitué de plus d'un million de portes, le taux d'activité de ces portes est passé de 20-25% en mode fonctionnel à 35-40%

en mode test par scan.

A titre d'exemple, la Figure I.3 détaille le processus d'application d'un vecteur de test "10101" à un circuit sous test. Dans cet exemple, on représente les commutations observées dans la chaîne de scan à chaque cycle d'horloge. Nous remarquons clairement qu'à chaque cycle d'horloge, toutes les cellules de la chaîne de scan commutent, ce qui entraîne une forte activité de commutation dans le circuit sous test, et par conséquent une forte consommation de puissance. A l'inverse, le même circuit en mode fonctionnel ne verra qu'une faible partie de ses flip-flops commuter, surtout s'il s'agit d'un circuit conçu pour une application "Low Power".

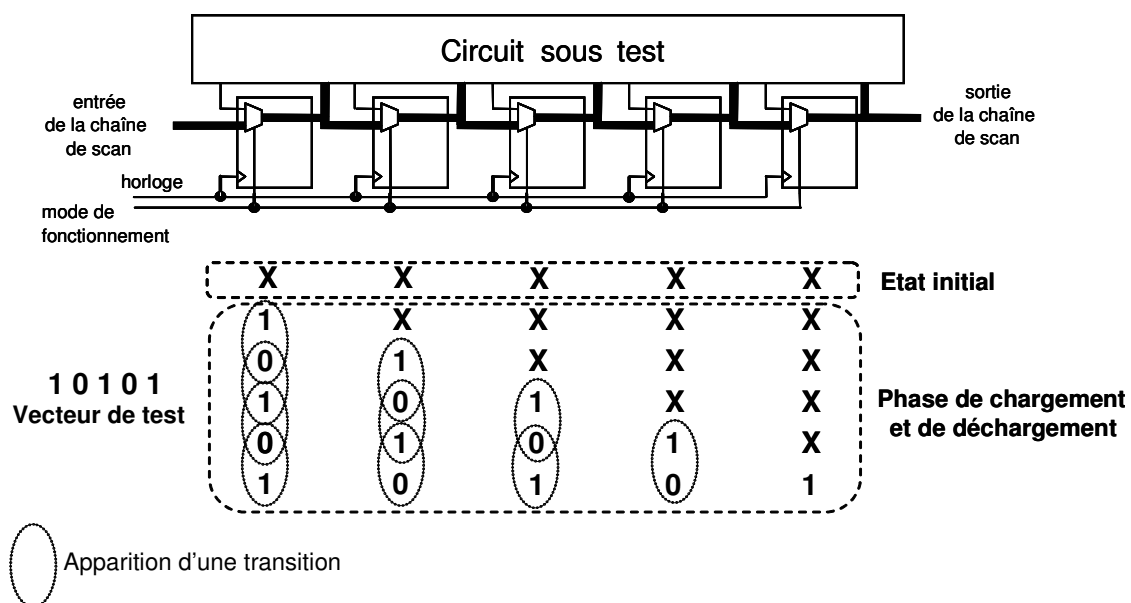


Figure I.3 Mise en évidence de l'activité de commutation lors du test par scan

1.4.2 Conséquences d'une consommation de puissance excessive

Une forte augmentation de la consommation de puissance durant le test peut causer une multitude de problèmes comme cela a été montré dans [GIRA02]. Dans cette partie, un résumé des problèmes induits par une consommation de puissance excessive lors du test est présenté.

Tout d'abord, la chaleur produite par le circuit est directement liée à sa puissance moyenne consommée. Une température élevée dans le circuit peut provoquer des dégradations irréversibles [CROU99]. Certaines de ces dégradations, comme les points chauds, apparaissent directement lors de l'application des données de test et peuvent provoquer la destruction du circuit sous test [GERS99]. D'autres dégradations, apparaissent

souvent de manière graduelle dans le temps, et peuvent affecter les performances du circuit ou encore provoquer des pannes fonctionnelles au delà d'une certaine durée d'utilisation. Dans ce cas, les principaux mécanismes conduisant à ces dégradations structurelles sont la corrosion (oxydation des conducteurs), l'électromigration (causée par le transport d'atomes de métal lorsqu'un flot de courant passe à travers une ligne, et qui provoque des cassures sur cette ligne), etc. [ALTE02].

Une forte activité de commutation lors du test peut également provoquer une chute de la tension d'alimentation de certaines portes du circuit en raison du fort appel de courant (di/dt) associé à cette forte activité. Ce phénomène peut entraîner une dégradation des performances temporelles, voire la non commutation d'une porte. Une forte activité de commutation peut aussi faire apparaître des phénomènes de bruit tels que Ground Bounce (rebond de masse), V_{dd} Bounce (rebond sur les lignes d'alimentation) ou Crosstalk. Ces phénomènes de bruit peuvent modifier l'état logique de certaines lignes du circuit à un instant donné. Combinés à une fréquence de test élevée, ces phénomènes de bruit ou de chute de tension peuvent faire en sorte que des circuits pourtant sains seront marqués défectueux [WANG97, SAXE03, CSHI04] (à cause d'une perte d'informations logiques), réduisant ainsi le rendement. Nous reviendrons plus en détails sur ces phénomènes de bruit et de chute de tension dans la partie 1.5.

1.4.3 Etat de l'art sur la réduction de puissance durant le test

Dans cette partie, nous allons faire un bilan des solutions qui existent pour limiter ou réduire la consommation de puissance lors du test. Ces solutions peuvent être regroupées en trois catégories : modification de la procédure de test, modification du circuit sous test ou modification de la séquence de test.

1.4.3.1 Modification de la procédure de test

Cette approche, largement utilisée dans l'industrie, consiste en une modification du processus de test. En effet, afin de réduire la consommation de puissance pendant le test, l'ingénieur de test augmente le temps de test soit en réduisant la fréquence de chargement, soit en insérant des pauses, ou soit en planifiant dans le temps les modules à tester.

Une première solution consiste à réduire la fréquence de test durant les cycles de décalage (chargement et déchargement des données dans la chaîne de scan). Cette solution permet de réduire la puissance moyenne dissipée dans le circuit. Cependant, le temps de test

est augmenté et aucune réduction de la puissance de pic n'est obtenue.

Une deuxième solution propose d'insérer des pauses (ou Break) durant les cycles de décalage pour permettre une meilleure dissipation de chaleurs et ainsi empêcher toute surchauffe excessive du circuit. Comme pour la première solution, cette technique ne réduit pas la puissance de pic et augmente le temps de test.

Dans [ZORI93], l'auteur propose de planifier dans le temps l'activation de chacune des parties du circuit (s'il s'agit d'un SOC) pendant le test afin de garder constante la dissipation de puissance. Les puissances moyenne et de pic sont réduites. Cependant, le temps de test est augmenté.

1.4.3.2 Modification du circuit sous test

Contrairement à la première approche, les solutions présentées dans cette partie ne modifient pas la procédure de test mais le circuit sous test par l'utilisation des techniques de Conception en Vue du Test (CVT ou Design-for-Test).

Une première solution possible consiste à augmenter le nombre ou sur-dimensionner les lignes d'alimentation afin de répondre à une forte consommation en courant lors du test. Cette solution permet également d'éviter les chutes de tension. Par contre l'utilisation d'une telle technique nécessite une estimation de la surconsommation de puissance dans le circuit lors du test dès la première étape de conception. Cette estimation ne peut se faire qu'avec la connaissance des données de test afin de les appliqués au circuit. Or, ces données de test sont généralement non disponibles lors de cette étape du processus de conception, et il apparaît donc clairement que cette solution ne peut être envisageable. Quoi qu'il en soit, une telle approche conduit à un surdimensionnement du circuit préjudiciable sur le plan de la surface.

Une autre solution est proposée dans [GERS99], où les auteurs proposent d'inhiber l'activité de commutation dans la chaîne de scan. Il s'agit ici d'une technique permettant de masquer le chargement et le déchargement des données de test dans la chaîne de scan. Ce masquage est effectué par l'intermédiaire d'une porte insérée entre la bascule scan et le circuit comme le montre la Figure I.4. Durant les cycles de décalage, le circuit se retrouve totalement isolé de la chaîne de scan. Il n'est donc pas affecté par l'activité de commutation dans la chaîne de scan. Même si elle réduit de façon significative la puissance moyenne et la puissance de pic pendant les cycles de décalage (de l'ordre de 35%), cette méthode présente plusieurs inconvénients comme la dégradation des performances temporelles ou la surface

additionnelle. Le surcoût en surface et en délai peut toutefois être réduit par un masquage partiel de la chaîne de scan. En effet, il n'est pas nécessaire de transformer toutes les bascules. Le choix des bascules à modifier peut être fait en fonction d'un coût obtenu par une analyse de l'influence des bascules sur la consommation du circuit [WUND85]. Ainsi, les bascules ayant le plus grand coût seront modifiées. Les auteurs ont proposé aussi dans [GERS00] une implémentation de cette technique réduisant le surcoût en surface.

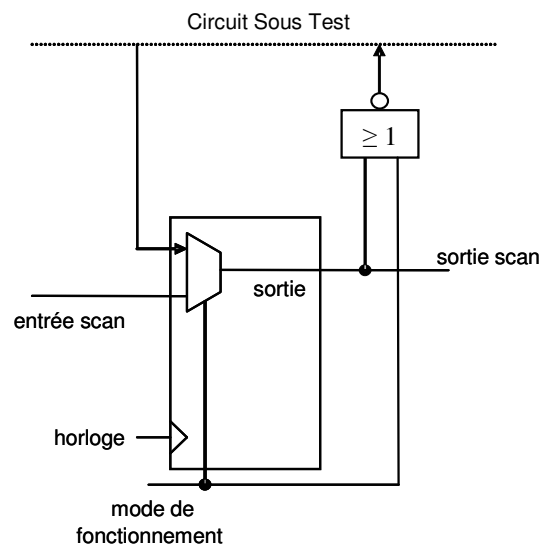


Figure I.4 Modification apportée à la bascule scan

Une autre technique, présentée dans [WHET00], consiste à segmenter la chaîne de scan. Le principe de cette technique est de réduire le nombre de commutations dans le circuit sous test, à l'aide d'une segmentation de la chaîne de scan en plusieurs sous-chaînes, celles-ci n'étant pas activées en même temps grâce à une adaptation du schéma d'horloge. La Figure I.5 montre un exemple d'une segmentation de la chaîne de scan en trois sous-chaînes de scan SC. A, SC. B et SC. C et l'adaptateur d'horloge (Figure I.5.a). Quand l'horloge H_A qui correspond à la sous-chaîne de scan A est activée, les autres sous-chaînes de scan sont inactives. Puis, la sous-chaîne de scan B est activée par l'activation de H_B , et ainsi de suite (Figure I.5.b). La dernière opération consiste à récupérer la réponse du circuit en un cycle d'horloge, d'où l'activation simultanément de H_A , H_B et H_C . Outre l'ajout de l'adaptateur d'horloge, cette technique nécessite l'ajout de portes de type buffer trois états (tristate) commandés par les signaux d'horloge. Ainsi par une division de la chaîne de scan, les réductions de la puissance (puissance moyenne et puissance de pic) obtenue lors des cycles de décalages dans [WHET00] vont jusqu'à 70%. Une approche similaire a également été proposée dans [BONH01].

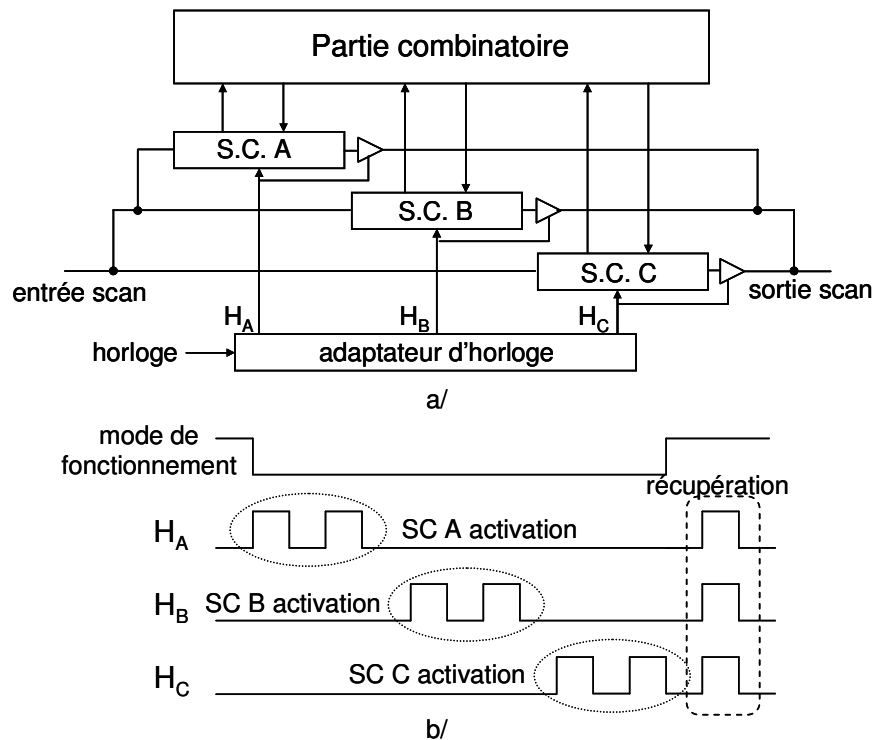


Figure I.5 Segmentation de la chaîne de scan

Dans [BONH02], les auteurs proposent également de réordonnancer les cellules de la chaîne de scan. Cette technique propose de réduire l'activité de commutation dans la chaîne de scan en modifiant l'ordre des cellules scan sur la base des données de test. Elle permet d'obtenir une réduction de la puissance moyenne de l'ordre de 60% pendant les cycles de décalage. Cette technique présente de nombreux avantages. Aucune porte additionnelle n'est nécessaire et le flot de conception reste inchangé. Cependant, la réduction de la puissance de pic est plus variable. Nous reviendrons plus en détail sur cette technique dans le Chapitre II où nous présentons une technique particulière d'ordonnancement des cellules scan pour la réduction de la puissance de pic pendant le test par scan.

1.4.3.3 Modification de la séquence de test

La dernière approche pour réduire la consommation de puissance lors du test consiste à modifier la séquence de test générée par un outil de génération automatique de vecteurs de test (Automatic Test Pattern Generator). Les techniques présentées dans cette partie proposent soit d'assigner les bits non spécifiés des séquences de test, soit de modifier la génération des séquences de test, soit encore d'appliquer des valeurs logiques précalculées sur les entrées primaires du circuit.

Les solutions présentées dans [WANG97, WANG99] consistent à utiliser un générateur de vecteurs de test permettant d'obtenir des séquences de test en vue d'une faible consommation de puissance. Ce générateur permet de minimiser le nombre de transitions durant le chargement et déchargement des données de test dans la chaîne de scan. Cependant, d'une part, le temps nécessaire pour obtenir un bon compromis entre le taux de couverture de fautes et la puissance dissipée est assez important. D'autre part, les générateurs utilisés tendent à augmenter la longueur de la séquence de test ainsi générée [ALYA03] comparée à une séquence de test fournie par un outil ATPG classique. Dans le Chapitre IV nous présentons une solution qui permet de réduire à la fois la consommation de puissance et le volume de données pendant le test par scan.

Dans [HUAN99], les auteurs proposent de contrôler les entrées primaires du circuit afin de réduire l'activité de commutation dans la chaîne de scan lors du test. Cette technique permet de minimiser l'influence du chargement et du déchargement des données de test dans la chaîne de scan en appliquant un vecteur inhibant, appelé vecteur de contrôle, sur les entrées primaires du circuit (Figure I.6). Cette technique permet d'obtenir des réductions de la puissance assez faible, de l'ordre de 15 à 20%. Un autre problème est le temps CPU nécessaire au calcul du vecteur à appliquer sur les entrées primaires.

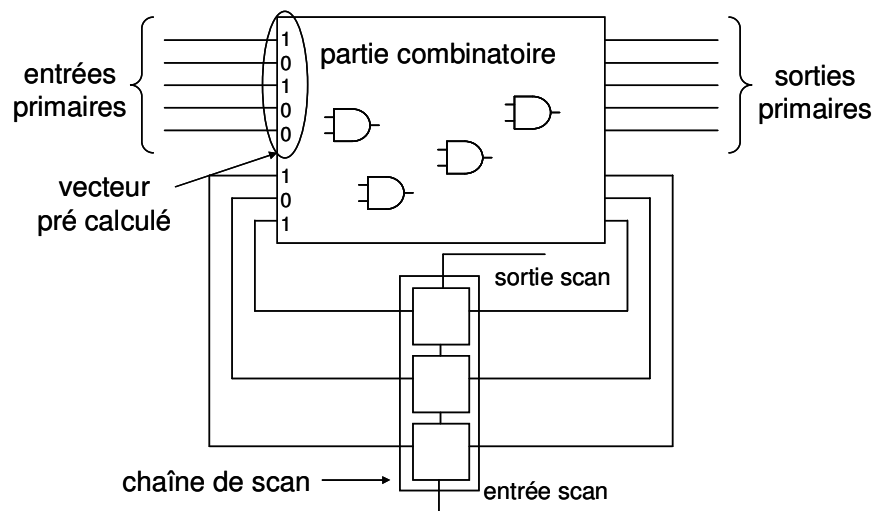


Figure I.6 Illustration de la technique par vecteur d'entrée précalculé

Dans [SANK00], les auteurs proposent une technique qui permet de réduire l'activité de commutation due au chargement des vecteurs de test dans la chaîne de scan. Pour ce faire, cette technique part de la séquence de test fournie par un outil ATPG sans l'assignation des bits non spécifiés. Une opération de compaction est effectuée, qui consiste à fusionner deux

vecteurs de test compatibles en un vecteur de test unique. Par exemple, si nous avons les vecteurs de test "11XX0" et "1X0X0", la fusion de ces deux vecteurs va nous donner le vecteur de test "110X0". Puis, la dernière opération consiste en l'assignation des bits à 'X' des vecteurs de test à des valeurs logiques. Cette technique permet d'atteindre jusqu'à 70% de réduction de la puissance dissipée dans le circuit. Une autre solution reprend le principe de l'assignation des bits non spécifiés des séquences de test mais sans aucune procédure de compaction. Cette solution propose d'assigner les bits à 'X' des séquences de test à des valeurs logiques de façon à réduire l'activité de commutation dans la chaîne de scan lors du test [SANK02, XWEN05]. Nous reviendrons plus en détails sur cette technique dans le Chapitre III où nous présentons différentes techniques d'assignation de bits non spécifiés afin de réduire la consommation de puissance de pic pendant le test par scan.

1.5 Analyse de la consommation de puissance de pic pendant le cycle de test

Dans le paragraphe précédent, nous avons vu que l'activité de commutation était généralement plus importante en mode test qu'en mode fonctionnel, et nous avons vu quel type de solutions pouvaient être mise en place pour répondre à ce problème. Afin de préciser davantage notre contribution dans ce domaine, reprenons dans le détail le déroulement du processus de scan, et voyons ce qui se passe en terme de consommation de puissance.

Pendant le test par scan, chaque vecteur de test est chargé dans la chaîne de scan par décalage. Cette phase utilise donc des "*cycles de décalage*". Après un certain nombre de cycles d'horloge, un dernier cycle de décalage, appelé "*cycle de test*", permet de charger le vecteur de test dans la chaîne de scan afin de l'appliquer au circuit sous test. Ensuite, le signal de contrôle SE est désactivé permettant ainsi la récupération de la réponse du circuit dans la chaîne de scan au prochain front d'horloge. Cette phase est appelée "*phase de capture*". La dernière étape consiste à activer le signal de contrôle SE afin de décharger la réponse par décalage tout en chargeant le vecteur de test suivant [WILL73, EICH91]. Notons au passage que *le courant consommé pendant le cycle de test est dû aux commutations générées dans la chaîne de scan pendant le dernier décalage*. En d'autres termes, le courant consommé pendant le cycle de test est dû aux commutations générées dans la chaîne de scan (avec n cellules) lors du passage du vecteur scan V_{n-1} au vecteur scan V_n (qui est le vecteur de test).

La Figure I.7 illustre la consommation de courant pendant le test par scan. Le premier appel de courant est dû aux commutations générées dans le circuit après le décalage du premier bit du vecteur de test dans la chaîne de scan. Le deuxième appel de courant est dû au deuxième décalage, et ainsi de suite. Le décalage du dernier bit du vecteur de test dans la chaîne de scan provoque une consommation de puissance dans le circuit pendant le cycle de test.

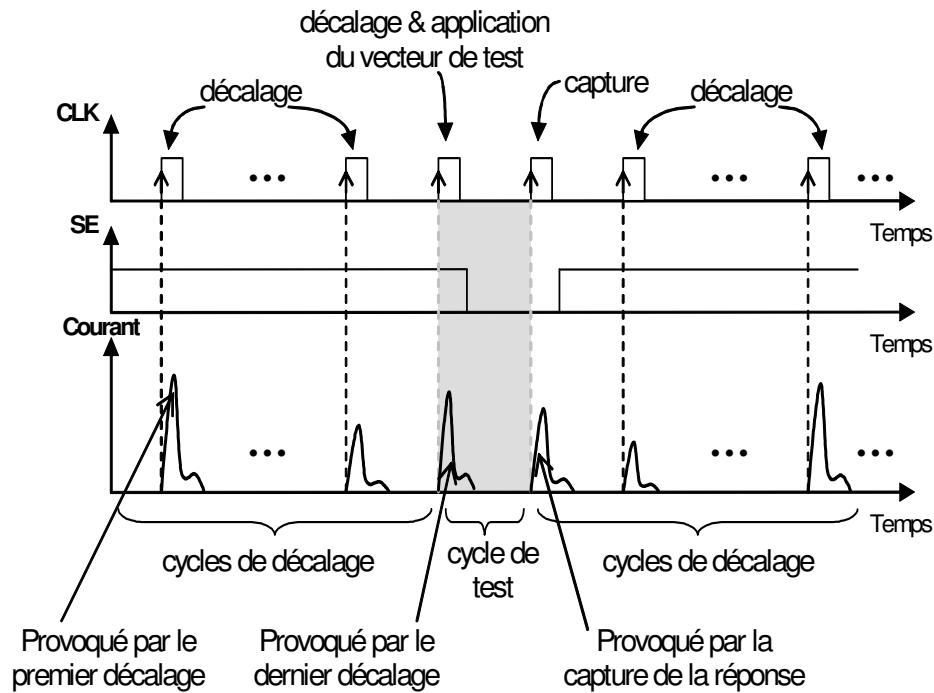


Figure I.7 Courant consommé pendant le test par scan

Comme nous avons pu le constater sur cette figure, des pics de courant très élevés peuvent apparaître pendant les cycles de décalage ou pendant le cycle de test. Dans les deux cas, plusieurs phénomènes de bruit, tels que Ground Bounce, V_{dd} Bounce, Crosstalk ou IR-Drop, peuvent apparaître dans le circuit sous test, alors que ces phénomènes sont normalement maîtrisés pendant le fonctionnement normal du circuit. Etant donné que les processus se déroulant durant ces deux phases sont différents (chargement/déchargement des données de test pendant les cycles de décalage, et capture de la réponse du circuit pendant le cycle de test), nous allons par la suite voir dans le détail les conséquences que peuvent avoir ces phénomènes durant chacune de ces phases.

L'observation de pics élevés et successifs de puissance (de courant) pendant les cycles de décalage peut engendrer des problèmes liés à l'augmentation de la température dans le

circuit sous test, tel que le phénomène d'électromigration qui peut réduire la fiabilité du circuit. Dans ce cas, des techniques de réduction de la puissance de pic pendant les cycles de décalage, comme par exemple l'inhibition de l'activité de commutation [GERS99] ou la segmentation de la chaîne de scan [WHET00] peuvent être utilisées. Un pic élevé de puissance de pic pendant les cycles de décalage peut aussi faire apparaître des phénomènes de bruit ou de chute de tension qui peuvent affecter à leur tour l'état logique des portes du circuit ou des cellules de la chaîne de scan. Cependant, ces phénomènes n'ont pas impact sur le résultat du test. En effet, d'une part, l'état logique des portes du circuit pendant les cycles de décalage n'a pas d'importance sur le résultat du test car aucune réponse n'est capturée dans la chaîne de scan à cet instant. D'autre part, la faible longueur des interconnexions entre les cellules de la chaîne scan fait en sorte que le délai engendré par les phénomènes de bruit n'a pas effet sur le contenu logique de ces cellules. Notons aussi que les cycles de décalage sont généralement réalisés à une faible fréquence pour des raisons de consommation de puissance, ce qui se traduit par une influence, encore plus faible, de ces phénomènes de bruit.

Contrairement aux cycles de décalage, le cycle de test est généralement réalisé à la fréquence de fonctionnement du circuit (At-Speed) [SAXE03, BUTL04]. Si cette fréquence élevée est associée à un fort appel de courant, alors dans ce cas, les phénomènes de bruit ou de chute de tension peuvent entraîner la capture de valeurs logiques erronées dans les cellules de la chaîne de scan s'ils apparaissent pendant le cycle de test. Dans ce cas, un circuit sain (fautif) pourra être déclaré fautif (sain) réduisant ainsi le rendement. Notons aussi que les circuits à venir seront de plus en plus sensibles aux bruits provenant soit de phénomènes inductifs tel que le ground bounce ou de phénomènes capacitifs tel que le crosstalk [SERV01].

Nous avons vu que des phénomènes de bruit ou de chute de tension peuvent avoir des conséquences graves sur le rendement s'ils apparaissent pendant le cycle de test. Afin de compléter et de préciser les explications précédentes, nous décrivons ci-après ces phénomènes.

Le phénomène d'"IR-drop" est une chute du niveau de la tension d'alimentation lorsqu'un fort appel de courant est mis en jeu. Cet appel de courant élevé intervient lors d'une commutation simultanée de plusieurs portes du circuit. En général, les circuits sont fournis avec une certaine limite de chute de tension. Le dépassement de cette limite peut engendrer une dégradation des performances temporelles du circuit avec d'éventuelles erreurs sur les résultats de test [SAXE03]. En effet, dans le cas d'une chute de tension, la capacité de sortie

d'une porte A peut être chargée à une tension inférieure à la tension de seuil d'une porte B en aval. Ainsi la commutation de la porte B est masquée et le résultat du test en est faussé [CSHI04].

Le "Ground Bounce" et le " V_{dd} Bounce" sont des phénomènes inductifs intervenant dans les circuits lorsqu'une forte variation de courant est mise en jeu. Dans les technologies submicroniques, des inductances apparaissent entre les lignes du circuit ("wire/substrate inductance") ou entre le boîtier et le circuit au niveau des sources d'alimentations ("package lead inductance"). Dans certains cas, un pic de tension peut être observé aux bornes de ces inductances si une forte variation de courant intervient. Ce pic de tension est proportionnel à la valeur de l'inductance et à l'amplitude de la variation du courant la traversant. Ce pic de tension non négligeable peut entraîner le changement d'état d'une ligne ou bien augmenter le délai des portes du circuit [SHIN97]. Le changement d'état d'une ligne entraîne une altération des informations logiques. L'augmentation du délai des portes provoque une dégradation des performances temporelles du circuit.

Le "Crosstalk" est caractérisé par les capacités de couplage qui peuvent exister entre les lignes du réseau d'interconnexion du circuit. Ce phénomène engendre une augmentation du délai des portes ce qui peut réduire le rendement du test.

1.5.1 Estimation de la puissance de pic pendant le test par scan

Le but de cette étude est de déterminer si la puissance de pic consommée pendant le cycle de test est plus importante que celle mise en jeu lors du fonctionnement normal du circuit. Cette comparaison ne peut se faire, à priori, qu'avec la connaissance des données fonctionnelles du circuit. Or, ces données fonctionnelles ne sont pas disponibles (pour les circuits expérimentés). Cependant, des expériences industrielles [SAXE03, CSHI04] ont montré que la puissance de pic consommée pendant le test par scan, et plus particulièrement pendant les cycles de décalage, est largement supérieure à celle consommée pendant le mode normal (3x voir 30x pour certains circuits industriels). A partir de cette conclusion, l'idée est de comparer la puissance de pic consommée pendant le cycle de test à celle consommée pendant les cycles de décalage. Si cette comparaison montre que la puissance de pic consommée pendant le cycle de test est supérieure ou du même ordre de grandeur que celle consommée pendant les cycles de décalage, alors nous pourrions en déduire que la puissance de pic consommée pendant le cycle de test est largement supérieure à celle consommée en mode normal. Nous allons par la suite expliquer la procédure suivie pour réaliser cette

comparaison.

Pour chaque circuit, nous avons considéré qu'une seule chaîne de scan avec n bascules et une séquence de test générée pour le test du circuit considéré. Nous avons mesuré pour chaque circuit expérimenté le courant consommé dans sa partie combinatoire à chaque période d'horloge. Ensuite, nous avons relevé la valeur maximale du courant pendant les $n-1$ cycles de décalage et pendant le cycle de test.

Les circuits expérimentés appartiennent aux benchmarks ISCAS'89 [BRGL89] et ITC'99 [DAVI99]. Le Tableau I.1 présente leurs caractéristiques principales. Pour chaque circuit, il indique le nombre de cellules scan, le nombre de portes, le nombre de vecteurs de la séquence de test et le taux de couverture de fautes de collage correspondant.

Circuit	# Cellules scan	# Portes	# Vecteurs de test	Taux de couverture de fautes (%)
b04s	66	512	55	99,14
b09	28	129	28	100
b10	17	155	43	100
b11s	31	437	60	100
b12	121	904	93	100
b13s	53	266	30	100
b14s	245	4444	409	99,52
b17s	1415	22645	692	98,99
s1196	18	529	128	100
s5378	179	2779	145	100
s9234	228	5597	159	99,76
s13207	669	7951	254	99,99
s38417	1636	22179	144	100

Tableau I.1 Caractéristiques des circuits expérimentés

Toutes les expérimentations sont basées sur un test déterministe avec des séquences de test générées par l'outil ATPG "Tetramax" de Synopsys [SYNO01] pour détecter les fautes de collage. Les entrées primaires et les sorties primaires ont été considérées comme stables durant le chargement des vecteurs et le déchargement des réponses dans la chaîne de scan. Nous avons considéré un état initial aléatoire pour les cellules scan de chaque circuit. Le courant consommé dans la partie combinatoire de chaque circuit a été estimé par l'outil de simulation "PowerMill" de Synopsys [SYNO00]. Les simulations ont été réalisées pour une

fréquence de test de 200MHz et une tension d'alimentation de 2,5 V et les circuits utilisés ont été implantés sur une technologie CMOS 0,25 μm standard. Les expérimentations ont été réalisées sur une station SUN Solaris 9 avec 32 Go de RAM. Le temps nécessaire à ces simulations est inférieur à une minute dans tous les cas.

Les résultats concernant la consommation de puissance de pic sont reportés dans le Tableau I.2. La première colonne présente le nom du circuit. Les deuxième et troisième colonnes indiquent le maximum des puissances de pic (en milliWatts) consommées pendant les cycles de décalage et les cycles de test.

Circuit	Max Puissance de pic (mW)	
	Cycles de décalage	Cycles de test
b04s	77,50	59,60
b09	34,43	30,48
b10	27,88	23,71
b11s	50,42	41,27
b12	113,84	101,46
b13s	61,09	52,92
b14s	395,55	319,83
b17s	1038,35	1118,68
s1196	66,89	10,03
s5378	197,76	179,66
s9234	359,68	339,88
s13207	499,68	483,30
s38417	1121,80	1074,33

Tableau I.2 Consommation de puissance de pic pendant le test par scan

Ces résultats montrent que la puissance de pic consommée pendant les cycles de décalage est toujours supérieure à celle consommée pendant les cycles de test. Ce résultat était prévisible car le nombre $n-1$ de cycles d'horloge pendant la phase (cycles) de décalage est très grand par rapport au seul cycle d'horloge de la phase (cycle) de test par vecteur. Néanmoins, ces résultats montrent que même si la puissance de pic consommée dans le circuit est plus importante pendant les cycles de décalage, la puissance de pic consommée pendant les cycles de test reste dans le même ordre de grandeur. Sachant que la puissance de pic consommée pendant les cycles de décalage est largement supérieure à celle consommée en mode normal [SAXE03, CSHI04], alors nous pouvons en déduire que la puissance de pic consommée pendant les cycles de test est plus importante que celle consommée pendant le fonctionnement

normal du circuit. Dans ces conditions, les phénomènes de bruit peuvent avoir un effet direct sur le délai des portes du circuit pendant le cycle de test. Comme cette phase est généralement réalisée à la fréquence de fonctionnement du circuit sous test (At-Speed), ces phénomènes ont de forte chance d'apparaître pendant le cycle de test et affecter les résultats du test, et par conséquent un circuit sain pourra être déclaré fautif réduisant ainsi le rendement du test. A partir de cette observation, nous pouvons en déduire qu'il faut limiter la consommation de puissance de pic pendant le cycle de test. Une des solutions possible est de réduire l'activité de commutation dans la chaîne de scan pendant cette phase.

1.5.2 Optimisation de la puissance de pic

A ce jour, aucune technique, à notre connaissance, ne permet de répondre à ce problème de consommation de puissance de pic excessive pendant le cycle de test lors d'un test At-Speed. En effet, d'une part, la seule technique utilisée dans l'industrie pour éviter l'apparition des phénomènes de bruit pendant le cycle de test est de réduire la fréquence pendant ce cycle, ce qui ne permet pas de réaliser un test At-Speed. D'autre part, la majorité des solutions présentées dans la partie état de l'art ciblent plutôt la réduction de la puissance moyenne et/ou de la puissance de pic pendant les cycles de décalage.

Cependant, quelques techniques peuvent être adaptées ou modifiées afin de réduire la puissance de pic pendant le cycle de test. La technique de segmentation de la chaîne de scan [WHET00] peut être utilisée pour réduire la puissance de pic pendant le cycle de test en capturant la réponse du circuit en plusieurs temps dans chaque sous-chaîne de scan. Or, cette modification ne permet pas de réaliser un test At-Speed. Une deuxième solution consiste à adapter la technique d'ordonnement des cellules scan [BONH02] dans le but de réduire la puissance de pic pendant le cycle de test. Nous nous sommes basés sur cette modification lors du développement des techniques d'ordonnement de cellules scan présentées dans le Chapitre II. Une autre solution consiste à modifier la séquence de test par une assignation adéquate des bits non spécifiés des séquences de test afin de réduire l'activité de commutation dans le circuit pendant le cycle de test. Les techniques présentées dans le Chapitre III de ce manuscrit se basent sur ce principe. Une dernière solution consiste à cibler conjointement la réduction de la puissance dissipée (puissance moyenne et puissance de pic) pendant tous le processus de scan et la compression du volume de données de test. Dans le Chapitre IV nous présentons une nouvelle technique qui permet d'obtenir des taux de compression de données très importants tout en réduisant d'une façon significative la consommation de puissance dans

le circuit pendant le test par scan.

1.6 Conclusion

Le test par scan est une technique de conception en vue du test permettant d'augmenter la testabilité des circuits intégrés. En contre partie, cette technique présente quelques inconvénients tels que le temps de test ou la surface additionnelle. Un autre problème lié au test par scan est la surconsommation de puissance.

La puissance dissipée lors du test par scan d'un circuit intégré peut être beaucoup plus importante que celle mise en jeu lors de son fonctionnement normal. Cette consommation excessive entraîne des conséquences de diverses natures sur le circuit. Ces conséquences sont soit liées à une forte consommation de puissance moyenne telle que l'augmentation de la température du circuit avec une possible apparition des phénomènes d'électromigration qui peuvent diminuer la fiabilité du circuit, soit liées à des pics de puissance élevés telles que l'apparition des phénomènes de bruit ou de chute de tension qui peuvent réduire le rendement du test. Dans les deux cas, une surconsommation de puissance peut détruire le circuit sous test.

Diverses solutions plus ou moins intéressantes ont vu le jour pour résoudre ce problème de surconsommation de puissance moyenne et/ou de puissance de pic lors du test par scan. Cependant, aucune de ces solutions ne permet de cibler le problème de la consommation de puissance de pic excessive durant le cycle de test.

Dans le cadre de cette thèse, nous avons mis au point plusieurs techniques afin de limiter la consommation de puissance de pic pendant le test par scan. Elles sont développées dans la suite de ce manuscrit.

Chapitre II

Réduction de puissance par réordonnancement des cellules scan

Chapitre II : Réduction de puissance par réordonnancement des cellules scan

2.1 Introduction

Le nombre de commutations qui intervient dans la chaîne pendant le test par scan influe directement sur les commutations produites dans le circuit, et donc sur la puissance dissipée dans celui-ci. Un moyen de réduire cette puissance est de réduire le nombre de commutations dans la chaîne de scan. Pour ce faire nous proposons dans ce chapitre une approche basée sur le réordonnancement des cellules scan.

Le réordonnancement des cellules scan consiste en la détermination d'un ordre optimal afin de réduire l'activité de commutation dans la chaîne de scan pendant le test. La première partie de ce chapitre montre l'influence de l'ordre des cellules scan sur l'activité de commutation dans la chaîne de scan ainsi que le problème d'optimisation à considérer. Dans la deuxième partie, nous détaillerons le problème du réordonnancement. Les deux dernières parties sont dédiées à la présentation des heuristiques d'ordonnement développées, à savoir l'algorithme glouton et l'algorithme de recuit simulé. Des résultats expérimentaux mettant en évidence l'efficacité de ces techniques et leurs inconvénients sont également présentés.

2.2 Impact de l'ordre des cellules scan sur la consommation de puissance

Le réordonnancement des cellules scan consiste en la détermination d'un ordre optimal qui réduit l'activité de commutation dans la chaîne de scan afin de minimiser la dissipation de puissance dans le circuit. Dans cette partie, nous montrons l'influence de l'ordre des cellules scan sur l'activité de commutation dans la chaîne de scan pendant le cycle de test. Pour cela, nous avons vu dans la partie 1.5 que pour réduire la consommation de puissance de pic pendant le cycle de test il faut minimiser le nombre de commutations dans la chaîne de scan pendant le dernier décalage (le passage du vecteur scan V_{n-1} au vecteur scan V_n qui est le vecteur de test), ce qui revient à réduire le nombre de transitions (différences de bits) sur chaque vecteur de test. Il faut noter que pour un calcul exact du nombre de commutations pendant le cycle de test, nous devons prendre en compte la différence entre le dernier bit de la réponse du circuit et le premier bit du vecteur de test suivant. Cependant, le nombre de cellules scan étant généralement très grand (de 20 jusqu'à plus de 1600 cellules pour les circuits expérimentés), ce qui se traduit par une très faible influence de ce bit. Pour des raisons de simplification, nous ne considérons pas ce bit et donc nous ne considérons pas la réponse du circuit. La Figure II.1 montre un exemple des commutations présentes dans la chaîne scan pendant le cycle de test. Le nombre de commutations à considérer dans cet exemple (nombre de transitions dans le vecteur de test) est alors égal à 4.

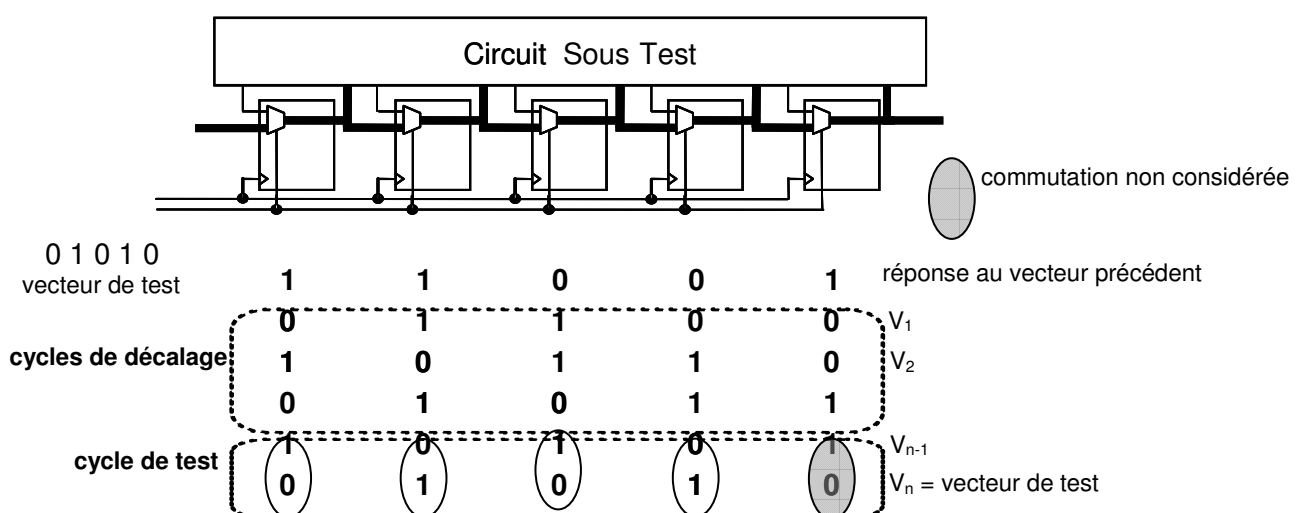


Figure II.1 Exemple des commutations dans la chaîne de scan pendant le cycle de test

Sur la Figure II.2, nous montrons l'influence de l'ordre des cellules scan sur l'activité de commutation dans la chaîne de scan pendant le cycle de test. L'ordre initial (Figure II.2.a : ff₁-ff₂-ff₃-ff₄-ff₅) donne un nombre de commutations des cellules scan pendant le cycle de test égal à 4. Si nous modifions l'ordre de ces cellules (Figure II.2.b : ff₁-ff₂-ff₄-ff₃-ff₅), nous constatons que ce nombre de commutations diminue et devient égal à 2. Dans ce cas, la simple modification de l'ordre des cellules scan entraîne une réduction de 50% du nombre de commutations dans la chaîne de scan pendant le cycle de test. Notons par ailleurs que le changement d'ordre des cellules scan n'entraîne en aucune façon un changement de connectique entre la chaîne de scan et le circuit. Le changement de connectique se passe entre les éléments de la chaîne de scan. Pour que la séquence de test ne soit pas modifiée, l'ordre d'application des données de test sur l'entrée scan de la chaîne de scan doit être adapté. L'ordre des bits des réponses du circuit est modifié de façon identique.

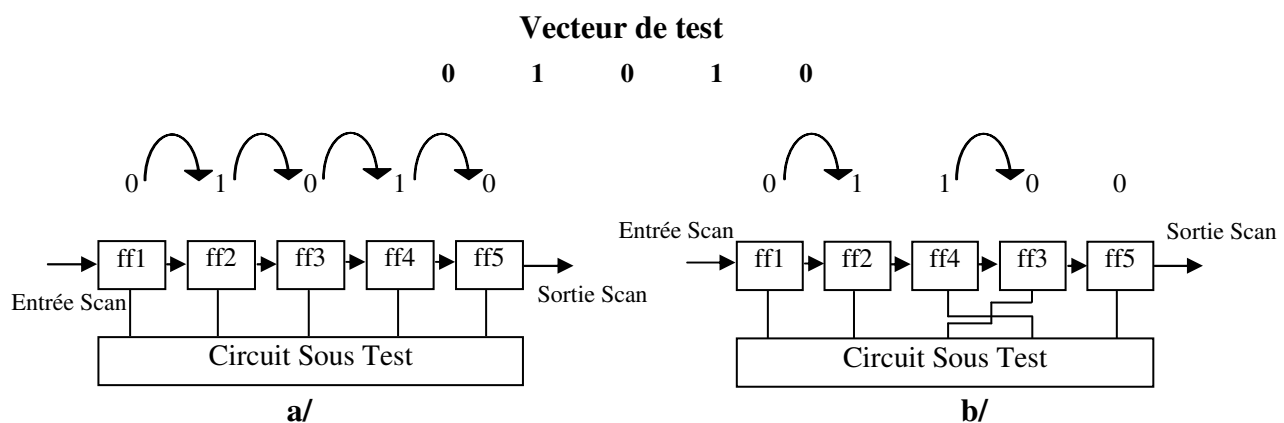


Figure II.2 Exemple d'ordonnements des cellules scan

Plusieurs études concernant la minimisation de la puissance pendant le test ont été basées sur ce principe, et il a été montré dans [BONH02] que le réordonnement des cellules scan permet de réduire de manière significative la puissance moyenne consommée durant les cycles de décalage. De notre part, nous nous sommes basés sur le réordonnement des cellules scan dans le but de réduire la consommation de puissance de pic pendant le cycle de test. Ce problème de réordonnement ciblant la puissance de pic peut être étudié de plusieurs façons différentes. Nous avons développé deux techniques d'optimisation globale afin de réduire la puissance de pic pendant le cycle de test.

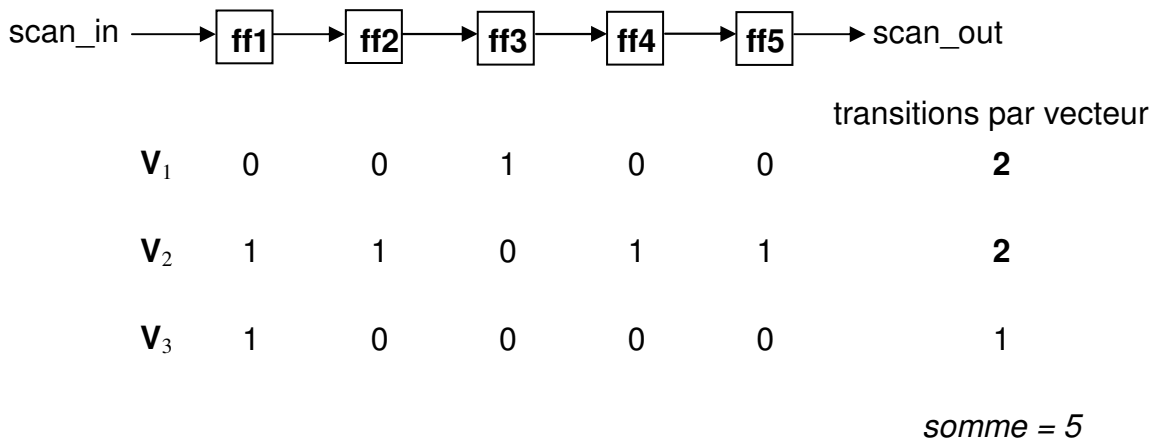
2.3 Optimisation globale sans contrainte

Une des solutions permettant de résoudre le problème de réordonnancement des cellules scan ciblant la puissance de pic pendant le cycle de test est de réduire le nombre de transitions d'une façon globale sur toute la séquence de test. C'est un problème d'*optimisation globale*.

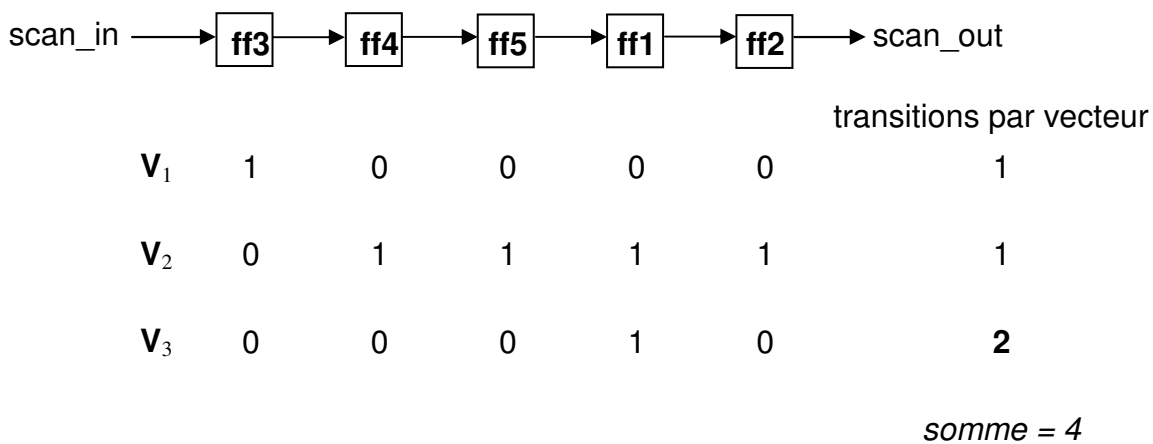
En réduisant la somme des transitions sur une séquence de test nous aurons une forte chance de réduire le nombre de transitions sur chaque vecteur composant la séquence de test considérée. De cette façon nous limiterons l'activité de commutation dans la chaîne de scan pendant le cycle de test. Cette méthode peut conduire aussi à une réduction de l'activité de commutation dans la chaîne de scan durant les cycles de décalage. En effet, un vecteur qui comporte un minimum de transitions provoque moins de commutations dans la chaîne de scan pendant son chargement. Cependant, le problème principal est d'obtenir une réduction significative de la somme des transitions sur tous les vecteurs de la séquence de test, sans pour autant dépasser la limite sur le nombre de transitions autorisées par vecteur de test. Cette limite doit être fixée par le concepteur du circuit et elle correspond à la puissance de pic maximale autorisée pendant le fonctionnement normal du circuit.

Afin d'illustrer ce problème, considérons l'exemple de la Figure II.3. Dans cet exemple nous représentons une séquence de test composée de 3 vecteurs générée pour un ordre des cellules scan "ff₁-ff₂-ff₃-ff₄-ff₅". Les vecteurs de test V₁ et V₂ présentent 2 transitions et V₃ une seule transition. La somme des transitions sur la séquence de test est alors égale à 5 (Figure II.3.a). Supposons que la limite sur le nombre de transitions autorisées par vecteur de test est une seule transition, alors le problème dans ce cas est de trouver un ordre des cellules scan qui réduit la somme des transitions sans générer un nouveau vecteur de test qui présenterait plus d'une transition.

L'ordre "ff₃-ff₄-ff₅-ff₁-ff₂" (Figure II.3.b) permet de réduire la somme des transitions sur la séquence de test de 5 à 4. Toutefois, même si cet ordre réduit d'une manière globale le nombre de transitions sur la séquence de test, il génère un nouveau vecteur V₃ qui dépasse la limite fixée. Alors, il faut essayer de minimiser la somme des transitions sur toute la séquence de test sous la condition (*contrainte de limite*) que chaque vecteur réordonné ne doit comporter plus d'une seule transition.



a/



b/

Figure II.3 Exemple d'optimisation globale

2.3.1 Modélisation du problème d'optimisation du chaînage des cellules scan

La première étape de la méthode d'optimisation globale consiste à construire un graphe pondéré à partir d'une séquence de test. La séquence de test est tout d'abord obtenue à l'aide d'un outil ATPG où le premier bit de chaque vecteur correspond à la première cellule scan, le deuxième bit correspond à la deuxième cellule et ainsi de suite. Une analyse de la séquence de test est effectuée de la façon suivante. Les bits associés à une même cellule (à la même position dans chaque vecteur de test) sont comparés aux bits d'une autre cellule. Cette comparaison permet de déterminer le nombre de bits différents entre chaque cellules scan sur l'ensemble de la séquence de test. De cette manière, le nombre de transitions peut être

minimisé en chaînant deux cellules dont le nombre de bits différents (sur l'ensemble de la séquence de test) est faible.

A titre d'exemple, considérons un circuit avec quatre cellules scan et une séquence de test comportant 8 vecteurs (Figure II.4) qui a été générée pour l'ordre de départ "ff₁-ff₂-ff₃-ff₄" des cellules de la chaîne de scan (le premier bit correspond à la 1^{ère} cellule, le second à la 2^{ème} cellule et ainsi de suite). Ensuite, nous calculons le nombre de différences de bits (distance de Hamming) entre chaque cellule afin de construire un graphe pondéré non orienté. Les nœuds de ce graphe correspondent aux cellules. Les arcs correspondent aux connexions possibles entre les cellules. Comme toutes les connexions sont possibles, le graphe est complet. Les pondérations sur ces arcs représentent le nombre de bits différents entre deux cellules (le nombre de transitions entre deux cellules sur l'ensemble de la séquence de test).

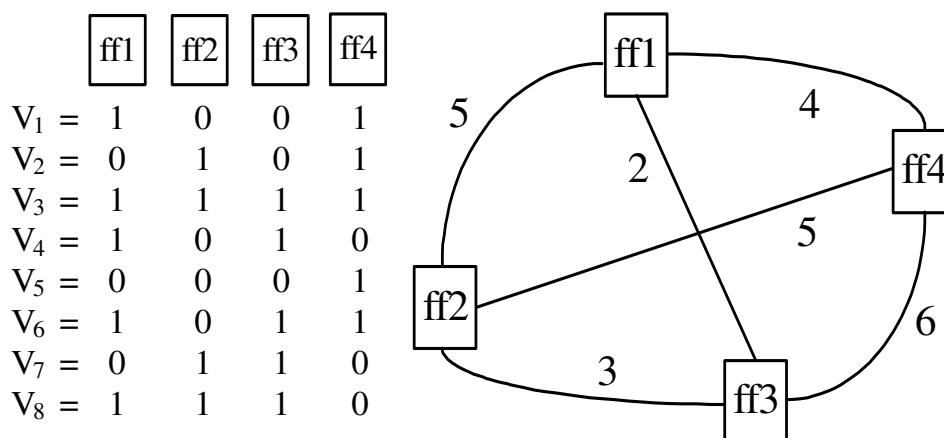


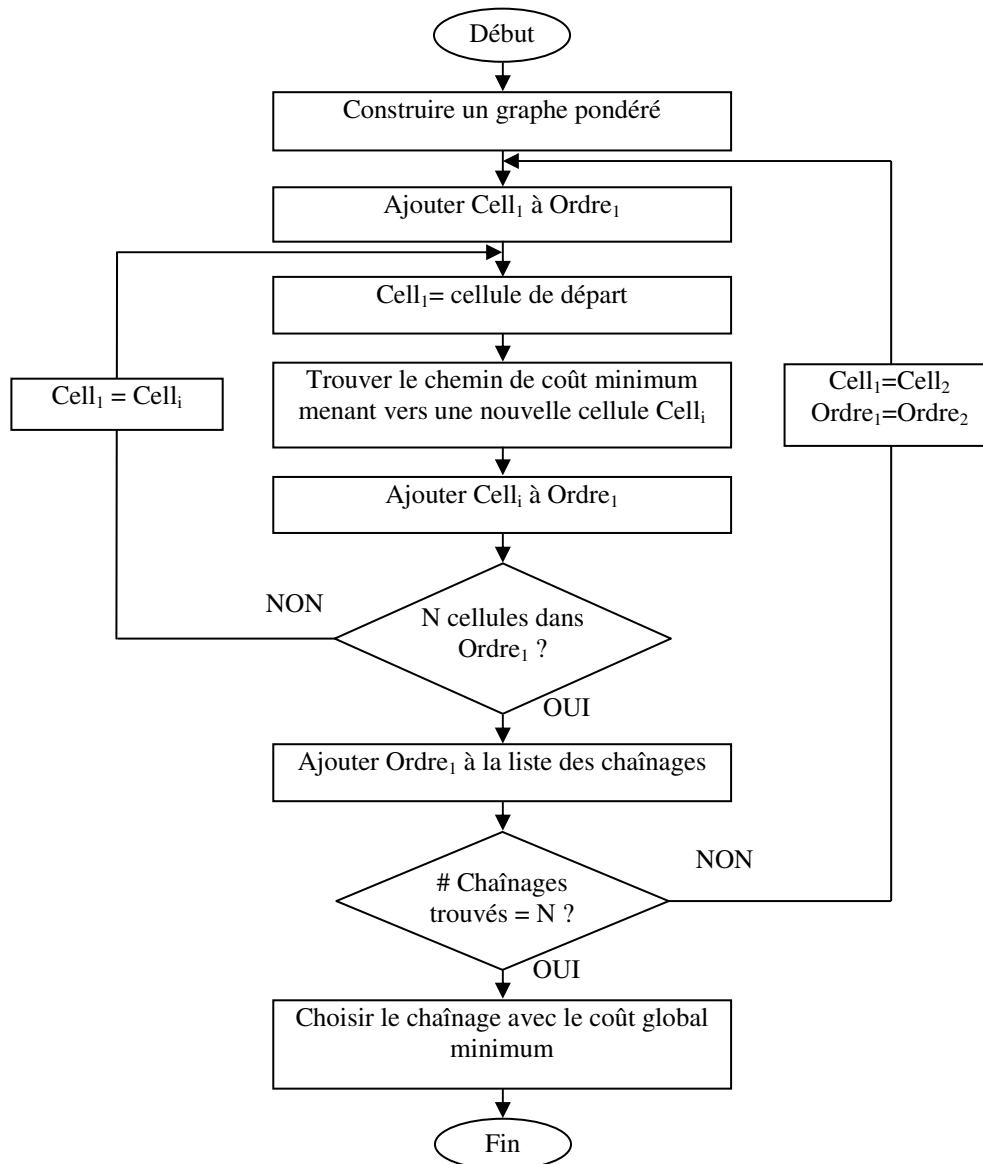
Figure II.4 Exemple de construction d'un graphe pondéré

Pour déterminer le chaînage minimisant l'apparition de transitions pendant le cycle de test (pour la séquence considérée), il suffit de trouver le chemin Hamiltonien (chemin passant une seule fois par tous les nœuds du graphe) de coût minimum dans le graphe. Ce problème, bien connu est celui du voyageur de commerce. C'est un problème NP-difficile. En effet, le nombre de solutions possibles est égal à $(n-1)!$ (où n est le nombre de nœuds dans le graphe).

Ce type de problème peut être résolu à l'aide de différents algorithmes avec une approximation en temps polynomial [JOHN96]. Pour résoudre ce problème, nous avons choisi d'utiliser l'algorithme glouton présenté ci-dessous dans le cadre d'une optimisation globale sans contrainte de limite [BADE05a].

2.3.2 Algorithme glouton

Le principe de cet algorithme est de trouver, à partir d'un graphe pondéré, un ordre optimal des cellules de la chaîne de scan réduisant le nombre de transitions sur tous les vecteurs de la séquence de test.



Organigramme II.1 Algorithme Glouton

L'organigramme II.1 présente les étapes principales du déroulement de l'algorithme glouton. Cet algorithme commence par la construction d'un graphe pondéré à partir d'une séquence de test donnée et d'un nombre N de cellules scan. Ensuite, il recherche un chaînage (ordre) des cellules scan de coût global minimum à partir d'une cellule de départ. Cette recherche commence par l'ajout de la première cellule ($Cell_1$) de la chaîne de scan au premier chaînage ($Ordre_1$) et la sélection de $Cell_1$ comme cellule de départ. Ensuite, l'algorithme

recherche à partir de cette cellule de départ le chemin de coût minimum menant vers une nouvelle cellule i ($Cell_i$). Il ajoute la cellule trouvée au premier chaînage et la sélectionne comme nouvelle cellule de départ. Ce processus est répété jusqu'à avoir toutes les cellules de la chaîne de scan dans ce premier chaînage.

A la fin de cette recherche, une autre boucle permet de réitérer ce processus pour différentes cellules de départ ($N-1$ cellules restantes) et de sélectionner le chaînage final de coût global minimum parmi les N chainages trouvés.

Considérons l'exemple de la Figure II.5. Les différents chainages trouvés par cet l'algorithme glouton pour quatre points de départ sont :

1. Chaînage 1 : $ff_1-ff_3-ff_2-ff_4$ avec un coût global égal à 10.
2. Chaînage 2 : $ff_2-ff_3-ff_1-ff_4$ avec un coût global égal à 9.
3. Chaînage 3 : $ff_3-ff_2-ff_1-ff_4$ avec un coût global égal à 12.
4. Chaînage 4 : $ff_4-ff_1-ff_3-ff_2$ avec un coût global égal à 9.

La solution finale trouvée par l'algorithme glouton sera le Chaînage 2 ($ff_2-ff_3-ff_1-ff_4$) avec un coût global minimum égal à 9.

2.3.3 Résultats expérimentaux

Les résultats présentés dans cette partie permettent de comparer l'approche proposée par rapport à une solution sans considération explicite de la consommation de puissance obtenue avec l'outil industriel de placement routage "Silicon Ensemble" de Cadence [CADE00]. Cette comparaison permettra de quantifier le gain en consommation de puissance de pic que peut apporter la technique proposée par rapport à une solution industrielle.

Le Tableau II.1 présente les résultats de réduction de puissance de pic pendant le cycle de test. Les caractéristiques des circuits étudiés, les caractéristiques de la station de travail et les outils utilisés sont identiques à ceux présentés dans la partie 1.5.1. La première colonne indique le nom du circuit. Les deuxième et troisième colonnes présentent respectivement le maximum des puissances de pic sur toute la séquence de test ($P_{pic_{max}}$) et leur moyenne ($P_{pic_{moy}}$) pour un ordonnancement des cellules scan fourni par Silicon Ensemble. Les quatrième et cinquième colonnes présentent respectivement $P_{pic_{max}}$ et $P_{pic_{moy}}$ pour un ordonnancement des cellules scan fourni par l'algorithme glouton développé. Les deux dernières colonnes présentent les pourcentages de réduction. Le temps nécessaire à la génération de l'ordre des cellules scan par l'algorithme proposé est de l'ordre de quelques

secondes dans tous les cas. Notons que les caractéristiques de test, à savoir le taux de couverture de fautes et le temps de test, restent inchangées par l'utilisation de cette technique.

Circuit	Solution Industrielle		Algorithme Glouton		Réduction (%)	
	Ppic _{max} (mW)	Ppic _{moy} (mW)	Ppic _{max} (mW)	Ppic _{moy} (mW)	Ppic _{max}	Ppic _{moy}
b04s	56,39	35,96	54,86	29,43	2,72	18,2
b09	29,66	18,91	20,96	13,48	29,33	28,7
b10	22,26	14,38	22,57	12,53	-1,38	12,8
b12	104,23	82,13	84,64	63,73	18,79	22,4
b13s	55,60	39,97	42,03	27,60	24,40	30,9
b14s	336,03	197,17	306,85	172,87	8,68	12,3
s420	31,64	14,63	25,68	11,17	18,83	23,7
s526	37,17	25,79	37,56	20,28	-1,07	21,4
s713	18,55	10,20	19,22	8,17	-3,58	20,0
s1196	12,15	4,98	8,95	4,03	26,35	19,0
s5378	187,47	150,86	142,45	118,85	24,02	21,2
s9234	343,13	247,32	302,65	200,74	11,80	18,8
Moyenne					13,24	20,78

Tableau II.1 Résultats comparatifs entre l'algorithme glouton et une solution industrielle

Nous remarquons que la solution proposée permet d'obtenir une réduction significative de la moyenne des puissances de pic sur toute la séquence de test qui peut atteindre 30% (20,78% en moyenne). En revanche, même avec une moyenne de 13,24%, la réduction du maximum des puissances de pic qui correspond à un seul vecteur de test n'est pas toujours garantie. En effet, pour les circuits b10, s526 et s713, nous avons une augmentation de 1,38%, 1,07% et 3,58% respectivement, du maximum des puissances de pic observées pendant l'application de la séquence de test. Ceci est dû à l'optimisation globale sans contrainte effectuée par l'algorithme glouton. En effet, cet algorithme réduit le nombre de transitions sur toute la séquence de test sans aucune contrainte sur le nombre maximum de transitions autorisées par vecteur de test. Le réordonnement génère de nouveaux vecteurs qui présentent un nombre de transitions important (supérieur à la limite).

Pour cette raison, nous avons développé un deuxième algorithme qui permet d'optimiser la réduction de la puissance de pic pendant l'application de toute la séquence de test, avec comme contrainte une limite sur le nombre de transitions autorisées par vecteur de test. Cet algorithme est basé sur le principe du recuit simulé.

2.4 Optimisation globale sous contrainte

Nous avons vu dans le paragraphe précédent que la technique d'optimisation globale sans contrainte permet d'obtenir des réductions significatives de la puissance de pic pendant le cycle de test sur toute la séquence de test. Cependant, dans certains cas, il y a eu une augmentation du maximum de la puissance de pic observée pendant l'application de la séquence de test. Ceci peut s'expliquer par le fait que nous n'avons imposé aucune contrainte de limite sur le nombre de transitions autorisées par vecteur de test. Pour cela, l'objectif de l'étude présentée dans cette partie est de réduire la somme des transitions sur la séquence de test tout en imposant une limite sur le nombre de transitions autorisées par vecteur de test.

Considérons de nouveau l'exemple présenté à la Figure II.3. L'ordre "ff3-ff4-ff5-ff1-ff2" (Figure II.3.b) permet de réduire la somme des transitions sur la séquence de test mais il génère une violation (dépassement) de la limite. En ajoutant une contrainte de limite à ce problème d'optimisation globale nous obtenons l'ordre "ff1-ff2-ff4-ff5-ff3" (Figure II.5). Cet ordre permet de réduire la somme de 5 à 3 transitions tout en respectant la limite d'une seule transition autorisée par vecteur de test.

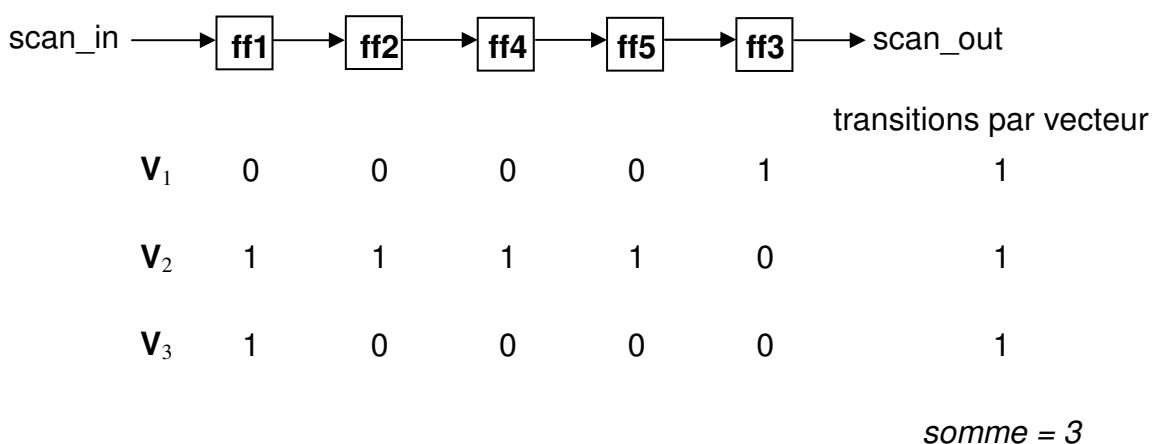


Figure II.5 Exemple d'une optimisation globale sous contrainte

Dans ce contexte, nous avons proposé un deuxième algorithme basé sur une optimisation globale sous contrainte [BADE05c, BADE06a]. Dans cette étude, nous avons utilisé un algorithme approché qui a été proposé pour résoudre des problèmes d'optimisation NP-difficile. Cet algorithme a montré son efficacité pour résoudre des problèmes de conception et de routage des circuits intégrés, il s'agit de l'algorithme de recuit simulé.

2.4.1 Algorithme de Recuit Simulé

Comme son nom l'indique, le recuit simulé est emprunté au domaine de la métallurgie. Si nous souhaitons obtenir un métal avec une structure parfaite de type cristal (c'est l'état fondamental correspondant au minimum d'énergie interne), nous devons procéder de la manière suivante : après avoir porté la matière à l'état liquide, nous allons abaisser la température jusqu'à solidification. Si la décroissance de température se fait de façon très brusque, nous obtenons un "verre". Si par contre elle se fait de façon très progressive, laissant le temps aux atomes d'atteindre l'équilibre statistique, nous tendons vers des structures de plus en plus régulières pour finir dans l'état fondamental : le "cristal", caractérisant le gel du système. Dans le cas où cet abaissement de température ne se fait pas assez lentement, il peut apparaître des défauts. Il faut alors les corriger en réchauffant de nouveau légèrement la matière de façon à permettre aux atomes de retrouver la liberté de mouvement, leur facilitant ainsi un éventuel réarrangement conduisant à une structure plus stable. Ce réchauffement porte le nom de recuit.

L'algorithme approché que nous avons utilisé dans cette partie est basé sur ce principe. Cet algorithme recherche le *minimum global* (solution optimale) pour un problème NP-difficile. Lors de cette recherche, l'algorithme peut rester bloquer autour d'une solution appelée *minimum local*. Afin d'éviter cela, il permet d'accepter un autre minimum (une autre solution) plus mauvais que ce minimum local (ce qui correspond au réchauffement du métal lors de la recherche de l'état cristal). La Figure II.6 illustre ce problème de blocage autour d'un minimum local lors de la recherche du minimum global.

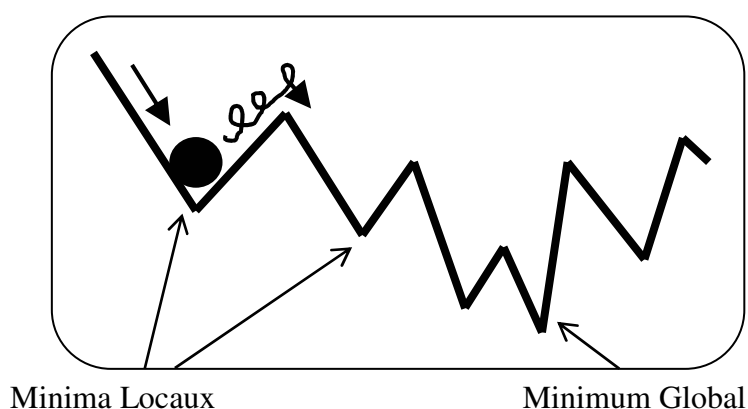


Figure II.6 Recherche du minimum global avec l'algorithme de recuit simulé

Concernant le problème de réduction de la puissance de pic pendant le cycle de test, nous avons développé un algorithme d'optimisation globale sous contrainte locale en se basant sur le principe du recuit simulé. En d'autres termes, l'objectif consiste à réduire la puissance de pic sur tous les vecteurs de la séquence de test sachant qu'il faut toujours rester en dessous d'une certaine limite. Cela revient à réduire le nombre global de transitions sur toute la séquence de test sans dépasser la contrainte locale qui définit le nombre maximum de transitions autorisées par vecteur de test.

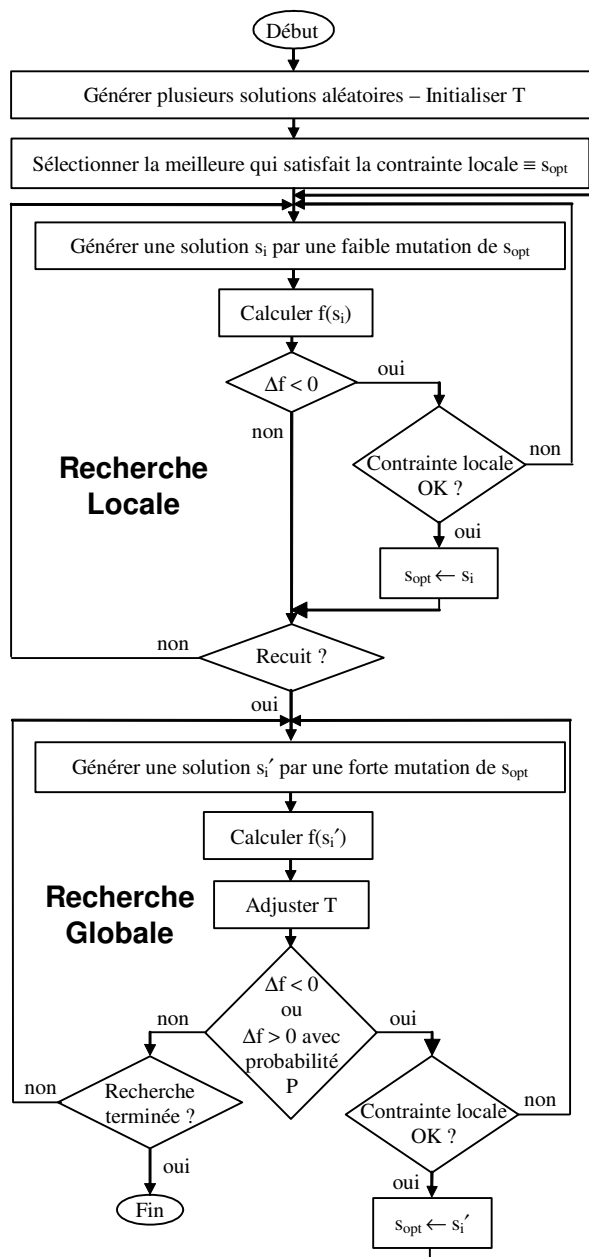
Les différentes étapes du déroulement de cet algorithme sont données dans l'organigramme II.2. L'entrée de l'algorithme est un nombre N de cellules scan et une séquence de vecteurs de test générée pour un ordre "ff1, ff2, ff3..., ffn". Le résultat est un ordre de cellules scan qui permet d'avoir le minimum de puissance de pic pendant les cycles de test sans pour autant dépasser la *contrainte locale* fixée par l'utilisateur.

L'algorithme commence par la génération de plusieurs solutions aléatoires (plusieurs ordres de cellules scan) et la sélection de la meilleure d'entre-elles S_{opt} (meilleur coût global) qui satisfait la contrainte locale. La meilleure solution est celle qui a le minimum de coût $f(S_{opt})$ qui correspond au nombre de différences de bits entre chaque cellules scan sur toute la séquence de test. Ensuite, le déroulement de l'algorithme se divise en deux grandes étapes : la recherche locale et la recherche globale.

Pendant la *recherche locale*, l'algorithme génère de nouvelles solutions en appliquant une faible mutation à la solution courante S_{opt} . Cette mutation consiste à permuter l'emplacement de *quelques cellules* dans la solution courante de façon aléatoire. Ensuite, il sélectionne la meilleure solution S_i parmi les solutions générées ($\Delta f = f(S_i) - f(S_{opt}) < 0$) qui vérifie la contrainte locale. Cette solution S_i devient alors la solution courante S_{opt} . Cette boucle de recherche locale est répétée i fois, où i est un paramètre à fixer par l'utilisateur.

La recherche globale, quand à elle, permet d'éviter de rester bloqué autour des minima locaux. Pendant cette recherche, l'algorithme génère de nouvelles solutions en appliquant une forte mutation à la solution courante S_{opt} . Cette mutation consiste à permuter l'emplacement de *plusieurs cellules* dans la solution courante de façon aléatoire. Ensuite, les solutions meilleures que S_{opt} ($\Delta f < 0$) et qui vérifient la contrainte locale sont acceptées, tandis que les solutions plus mauvaises que S_{opt} ($\Delta f > 0$) peuvent être acceptées avec une certaine probabilité $P = \exp(-\Delta f/T)$. La température T est un paramètre à fixer par l'utilisateur. Elle décroît avec l'avancement de la recherche algorithmique réduisant ainsi la probabilité P d'accepter des

solutions plus mauvaises que S_{opt} . Cette boucle de recherche globale est réitérée j fois, où j est un paramètre à fixer par l'utilisateur.



Organigramme II.2 Procédure du déroulement de l’algorithme de recuit simulé

2.4.2 Résultats expérimentaux

Dans cette partie, nous présentons tout d’abord les résultats des réductions de la puissance de pic obtenus avec l’algorithme de recuit simulé pendant le cycle de test, ensuite pendant les cycles de décalage, comparé à une solution industrielle (sans considération de la

consommation de puissance). La contrainte locale qui consiste à fixer le nombre maximum de transitions autorisées par vecteur de test a été déterminée expérimentalement. Pour chaque circuit nous avons imposé une contrainte locale assez grande au début des simulations. Ensuite, la valeur de cette contrainte diminue de façon progressive tout au long de la procédure de recherche de la solution optimale. Les résultats présentés dans cette partie sont donnés pour une contrainte de limite qui permet d'avoir une solution en un temps raisonnable (moins de 10 minutes pour les grands circuits). Par exemple, pour le circuit b04s cette contrainte a été fixée à 32 transitions autorisées par vecteur de test (129 pour b14s et 118 pour s9234). Le nombre maximum de transitions sur les vecteurs de la séquence de test obtenue pour un ordre fourni par l'outil Silicon Ensemble était égal à 45 pour le circuit b04s (151 pour b14s et 140 pour s9234). Notons par ailleurs que cette limite peut être différente de la limite réelle qui doit être fixée par le concepteur du circuit (absence des caractéristiques électriques de fonctionnement des circuits expérimentés). Les nombres d'itérations i et j des boucles de recherche locale et globale ont été eux aussi déterminés expérimentalement pour un temps de recherche inférieur à 10 minutes.

A la fin de cette partie nous présentons aussi d'autres résultats comparatifs afin de mesurer l'impact de notre approche sur le taux de couverture de fautes de transition.

2.4.2.1 Réduction de puissance de pic pendant le cycle de test

Les résultats expérimentaux sont regroupés dans le Tableau II.2. Ces résultats présentent les réductions de la puissance de pic pendant le cycle de test obtenues avec l'algorithme de recuit simulé comparé à une solution industrielle obtenu avec Silicon Ensemble.

Nous remarquons l'efficacité de notre algorithme dans la réduction de la moyenne des puissances de pic (en moyenne 19,78% de réduction de $P_{pic_{moy}}$) qui peut atteindre plus de 50% (30% dans le cas de l'algorithme glouton). La solution fournie par l'algorithme glouton avait présentée une réduction moyenne de 20,79% de $P_{pic_{moy}}$, ce qui n'est pas très différent des 19,78% obtenus avec l'algorithme de recuit simulé.

L'importance de ces résultats, c'est que nous n'avons en aucun cas une augmentation de $P_{pic_{max}}$ contrairement aux résultats obtenus avec l'algorithme glouton. D'autre part, la réduction de $P_{pic_{max}}$ est passée d'une moyenne de 13,24% avec l'algorithme glouton à 20,51% avec l'algorithme de recuit simulé. Cette solution permet de réduire significativement la puissance de pic sur toute la séquence de test et d'assurer le respect de la limite qui doit être

fixée par le concepteur du circuit. De cette façon, nous pouvons éviter une possible apparition des phénomènes de bruit lors de la capture de la réponse du circuit, et donc une éventuelle perte de rendement.

Circuit	Solution Industrielle		Algorithme de Recuit Simulé		Réduction (%)	
	Ppic _{max} (mW)	Ppic _{moy} (mW)	Ppic _{max} (mW)	Ppic _{moy} (mW)	Ppic _{max}	Ppic _{moy}
b04s	56,39	35,96	45,85	30,98	18,69	13,86
b09	29,66	18,91	17,14	9,22	42,2	51,23
b10	22,26	14,38	20,67	12,76	7,13	11,22
b12	104,23	82,13	87,73	68,85	15,82	16,17
b13s	55,6	39,97	38,03	28,39	31,59	28,97
b14s	336,03	197,17	324,35	186,4	3,47	5,46
s420	31,64	14,63	20,53	10,78	35,12	26,32
s526	37,17	25,79	33,06	20,02	11,06	22,4
s713	18,55	10,2	13,63	8,54	26,55	16,31
s1196	12,15	4,98	8,99	4,12	26	17,28
s5378	187,47	150,86	154,08	127,8	17,81	15,28
s9234	343,13	247,32	306,55	215,59	10,66	12,83
Moyenne					20,51	19,78

Tableau II.2 Résultats comparatifs entre l'algorithme de recuit simulé et une solution industrielle

2.4.2.2 Réduction de puissance de pic pendant les cycles de décalage

Les résultats présentés dans cette partie concernent la réduction de la puissance de pic pendant les cycles de décalage lors du test par scan. Nous avons évalué cette réduction dans le but de mesurer l'impact de la solution fournie par l'algorithme de recuit simulé sur la puissance de pic pendant cette phase. Ces résultats sont regroupés dans le Tableau II.3. La dernière colonne indique les réductions de la moyenne des puissances de pic sur toute la séquence de test obtenues avec l'algorithme de recuit simulé comparé à la solution industrielle fournie par Silicon Ensemble.

Ces résultats montrent que la solution fournie par l'algorithme de recuit simulé produit une faible réduction de la puissance de pic pendant les cycles de décalage par rapport à la solution industrielle (5,7% en moyenne avec un maximum de 15,8%). Toutefois, même si aucune réponse du circuit n'est capturée dans la chaîne de scan à cet instant, il faut éviter des pics de puissance élevés et surtout successifs pendant cette phase afin de limiter les problèmes liés à une forte température dans le circuit. Cela nous a mené à développer d'autres techniques dans le but de réduire la puissance de pic, non seulement pendant le cycle de test, mais aussi

pendant les cycles de décalage et qui sont présentés dans le chapitre suivant.

Circuit	Solution Industrielle	Algorithme de Recuit Simulé	Réduction (%)
	Ppic _{moy} (mW)	Ppic _{moy} (mW)	
b04s	58,07	53,85	7,3
b09	29,05	26,09	10,2
b10	21,37	20,1	5,9
b12	97,89	82,45	15,8
b13s	49,49	46,78	5,5
b14s	335,8	329,7	1,8
s420	21,5	19,57	9
s526	36,15	31,6	12,6
s713	18,39	18,23	0,9
s1196	36,53	36,85	-0,9
s5378	167,23	157,99	5,5
s9234	322,15	317,86	1,3
Moyenne			5,7

Tableau II.3 Réduction de puissance de pic pendant les cycles de décalage

2.4.2.3 Couverture de fautes de transition

Cette partie est consacrée à la présentation de l'impact du réordonnement des cellules scan sur le taux de couverture de fautes de transition. Cet algorithme permet de réduire le nombre de transitions pendant le cycle de test. Cependant, cette réduction du nombre de transitions peut affecter la détection des défauts pendant la capture de la réponse du circuit, en particulier les défauts liés au délai des portes dans le circuit. Pour cela, nous avons mesuré le taux de couverture de fautes de transition, pour chaque circuit, pour un ordonnancement fourni par l'outil Silicon Ensemble et un ordonnancement fourni par notre algorithme de recuit simulé. Pour ce faire, nous avons procédé de la façon suivante. Nous avons commencé par la génération d'une séquence de test (par des outils de programmation) contenant toutes les paires de vecteurs scan correspondant au dernier décalage de chaque vecteur composant la séquence de test de départ. La nouvelle séquence générée contient les vecteurs scan VI_{n-1} et VI_n (la paire de vecteurs scan correspondant au dernier décalage du premier vecteur de la séquence de test), $V2_{n-1}$ et $V2_n$ et ainsi de suite. La dernière étape consiste à appliquer cette nouvelle séquence à l'outil TetraMax pour une simulation de fautes de transition non robuste. Cette procédure est connue dans le domaine du test de production sous le nom *scan-shift delay test*.

Les taux de couverture de fautes de transition obtenus sont reportés dans le Tableau II.4. Nous remarquons que le taux de couverture de fautes de transition non robuste obtenu avec la solution de l'algorithme développé (3^{ème} colonne) est peu différent de celui obtenu avec la solution industrielle (2^{ème} colonne). La dernière ligne donne la moyenne de ces taux de couverture de fautes (57,33% pour la solution industrielle contre 56,48% pour la solution optimisée en puissance de pic). Cette faible différence est expliquée par le fait que notre algorithme de recuit simulé réduit d'une manière globale le nombre de transitions sur toute la séquence de test de façon à obtenir une régularité du nombre de transitions présentes dans chaque vecteur de test. En d'autres termes, si le nombre de transitions dans un vecteur de test donné dépasse la limite autorisée, alors l'algorithme ramène ce nombre de transitions en dessous de cette limite. En revanche, si le nombre de transitions dans un vecteur de test donné est inférieur à la limite fixée, alors l'algorithme peut être amené à augmenter le nombre de transitions de ce vecteur (par un réordonnement des cellules scan) dans le but de réduire le nombre de transitions des autres vecteurs de test qui présentent un nombre de transitions supérieur à la limite (tout en gardant le nombre de transitions de ce vecteur en dessous de la limite). De cette façon, nous aurons pour chaque vecteur de test un nombre de transitions plus ou moins important, mais qui reste toujours inférieur au nombre maximum de transitions autorisées par vecteur de test.

Circuit	Taux de couverture des fautes de transition non robustes (%)	
	Solution Industrielle	Algorithme de Recuit Simulé
b04s	66,17	56,89
b09	66,92	59,85
b10	62,59	64,52
b12	64,61	61,07
b13s	63,6	64,95
b14s	69,1	66,05
s420	41,87	48,01
s526	58,85	60,38
s713	58,73	60,5
s1196	17,98	18,88
s5378	64,83	64,1
s9234	52,67	52,5
Moyenne	57,33	56,48

Tableau II.4 Résultats comparatifs des taux de couverture de fautes de transition non robuste

La Figure II.7 présente un exemple de cette régularité du nombre de transitions sur la séquence de test obtenue avec l'algorithme de recuit simulé pour le circuit s713. La limite du nombre de transitions autorisées par vecteur de test a été fixée, dans ce cas, à 11.

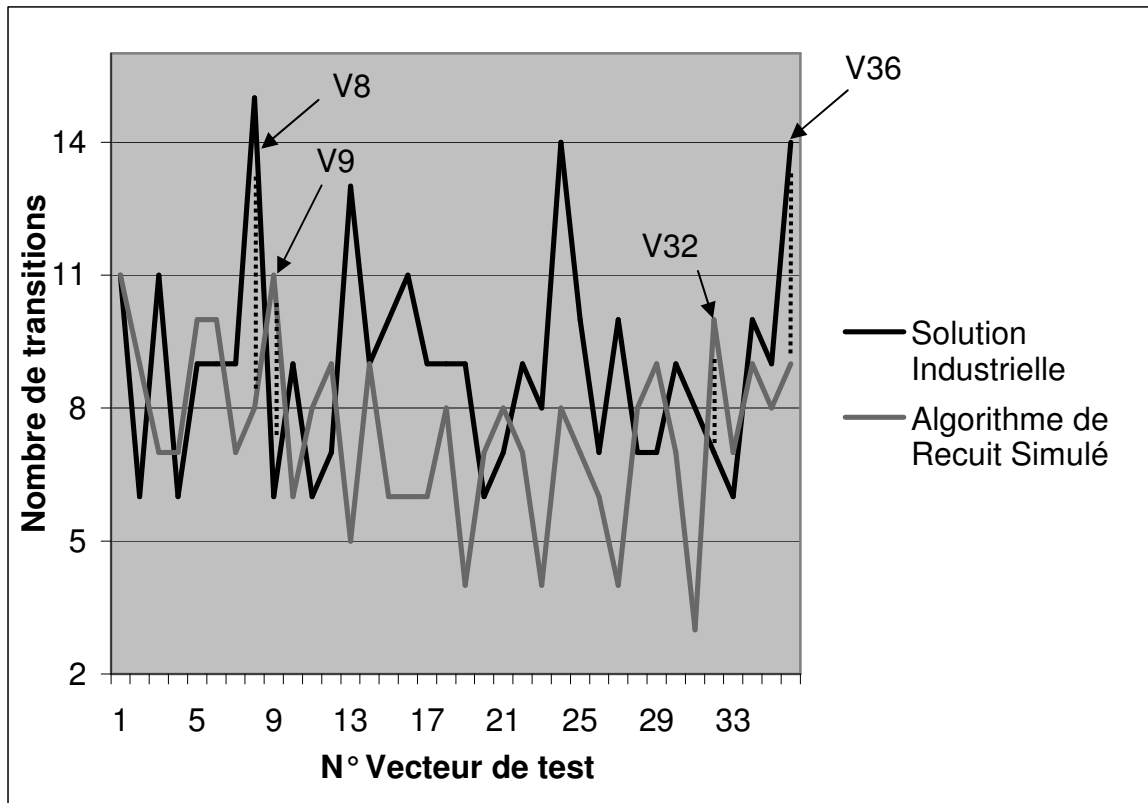


Figure II.7 Exemple de réduction du nombre de transitions par vecteur de test avec l'algorithme de recuit simulé pour le circuit s713

Nous remarquons que le nombre de transitions pour les vecteurs de test V_8 et V_{36} avec la solution industrielle est supérieur à 11 (15 et 14 respectivement). Dans ce cas, l'algorithme de recuit simulé a réduit ce nombre de transitions en dessous de la limite fixée (8 transitions pour V_8 et 9 pour V_{36}). Par contre, pour les vecteurs de test V_9 et V_{32} dont le nombre de transitions est inférieur à 11 avec la solution industrielle (6 et 7 respectivement), l'algorithme de recuit simulé a augmenté ce nombre de transitions tout en respectant la limite fixée à 11 (11 transitions pour V_9 et 10 pour V_{32}). De cette façon, le nombre de transitions total sur toute la séquence de test obtenu avec l'algorithme de recuit simulé reste dans le même ordre de grandeur que celui obtenu avec la solution industrielle, ce qui explique la faible différence des taux de couverture de fautes de transition obtenus avec et sans optimisation de la consommation de puissance de pic.

2.5 Conclusion

Nous avons vu dans ce chapitre une technique efficace pour élaborer un ordonnancement des cellules de la chaîne de scan permettant de réduire significativement la puissance de pic pendant le cycle de test. En effet, Les résultats obtenus permettent de réduire jusqu'à 50% la puissance de pic dissipée dans le circuit pendant le cycle de test par rapport à l'ordonnancement réalisé avec un outil industriel sans considération de la consommation de puissance.

Nous avons développé un algorithme glouton pour une optimisation globale sans contrainte. Les résultats obtenus sont intéressants dans le cas d'une réduction globale de la puissance de pic sur toute la séquence de test. Cependant, cet algorithme ne garantit pas la non violation de la limite sur la puissance de pic qui doit être fixée par le concepteur du circuit.

Aussi, nous avons proposé un deuxième algorithme en vue d'une optimisation globale sous contrainte basé sur le principe du recuit simulé. Les résultats obtenus sont intéressants, car ils montrent qu'une réduction significative de la puissance de pic peut être obtenue pour tous les vecteurs composant la séquence de test, tout en respectant la contrainte de limite sur la puissance de pic. Par ailleurs, nous avons montré que cette technique n'affecte pas le taux de couverture de fautes de transition. En effet, la technique développée permet de garder le nombre total des transitions présentes sur toute la séquence de test dans le même ordre de grandeur que celui obtenu avec la solution industrielle.

Cette technique présente de nombreux avantages. Elle est facilement intégrable dans les outils de conception et donc facilement insérable dans le flot de conception. Elle ne modifie pas le temps de test et le taux de couverture reste inchangé. Elle ne nécessite l'ajout d'aucune logique supplémentaire.

De plus, cette technique permet d'obtenir une séquence possédant des propriétés intéressantes pour les algorithmes de compression. En effet, la séquence résultante possède des séries de bits identiques, caractéristique qui permet d'obtenir de très bon taux de compression.

Un des inconvénients de cette technique est lié au routage résultant des interconnexions reliant les cellules de la chaîne de scan entre elles. Il est clair qu'en modifiant l'ordre des cellules pour réduire la puissance dissipée lors du test, le routage final de la chaîne

de scan sera augmenté. Dans ce cas, la technique présentée dans [BONH03] peut être utilisée. Cette technique consiste à partitionner le circuit en clusters afin d'optimiser le routage des cellules scan.

Un autre inconvénient de l'approche proposée est qu'elle ne permet pas d'avoir une bonne réduction de la puissance de pic pendant les cycles de décalage. Les résultats présentés dans ce chapitre ont montré que la solution de réordonnancement fournie par l'algorithme de recuit simulé proposé peut atteindre seulement 5,7% en moyenne, avec un maximum de 15,8% de réduction de la puissance de pic pendant les cycles de décalage, par rapport à une solution industrielle. Cependant, nous devons réduire la puissance de pic pendant les cycles de décalage pour éviter l'apparition des phénomènes liés à une forte température mise en jeu dans le circuit sous test. Cela nous a mené à développer d'autres techniques dont l'objet consistait à réduire la puissance de pic non seulement pendant le cycle de test, mais aussi pendant les cycles de décalage, tout en éliminant le problème du routage des cellules de la chaîne de scan. Ces techniques sont présentées dans le chapitre suivant.

Chapitre III

Réduction de puissance par assignation de bits non spécifiés

Chapitre III : Réduction de puissance par assignation de bits non spécifiés *

3.1 Introduction

Ce chapitre est consacré à la présentation de la technique d'assignation de bits non spécifiés (bits à 'X') des séquences de test que nous avons développée pour réduire la consommation de puissance dans les circuits durant le test. Cette technique consiste à spécifier les vecteurs de test en affectant des valeurs logiques '0' ou '1' aux bits non spécifiés, de manière à optimiser l'impact de ces assignations sur la consommation de puissance lors du test. Cela permet aussi d'éviter le problème de routage qui est lié au réordonnement des cellules de la chaîne de scan.

La première partie de ce chapitre montre l'impact sur la consommation de puissance de l'affectation des bits non spécifiés d'une séquence de test. Dans la deuxième partie, nous présenterons les techniques classiques d'assignation de bits à 'X' ainsi que les résultats correspondants. Nous mettrons en évidence l'efficacité de ces techniques et la nécessité de prendre en compte la structure interne du circuit lors de l'assignation des bits non spécifiés. La troisième partie sera consacrée à la présentation de la technique que nous avons développée pour permettre l'assignation de bits à 'X' en se basant sur des informations structurelles. Dans la dernière partie, des résultats comparatifs entre les différentes techniques en terme de réduction de puissance de pic pendant le cycle de test et pendant les cycles de décalage, ainsi qu'en terme d'impact sur le volume de données de test et sur le taux de couverture de fautes seront présentés.

* Ce travail a été réalisé dans le cadre d'une collaboration avec le Professeur Hans-Joachim WUNDERLICH de l'Université de Stuttgart en Allemagne.

3.2 Impact de l'assignation de bits non spécifiés sur la consommation de puissance lors du test

Lors de la génération d'un vecteur de test, l'algorithme utilisé par l'outil ATPG affecte des valeurs logiques '0' ou '1' aux cellules de la chaîne de scan et aux entrées primaires du circuit sous test nécessaires pour la détection de la faute ciblée. Ces valeurs sont appelées les *bits spécifiés*. Les bits correspondants aux cellules scan et aux entrées primaires restantes ne sont pas assignés. Ces bits sont mentionnés par des 'X' et sont appelés les *bits non spécifiés*.

Les bits non spécifiés d'un vecteur de test sont généralement affectés de manière aléatoire à des valeurs logiques '0' ou '1' par l'outil ATPG. Le vecteur de test résultant (avec tous les bits spécifiés à une valeur logique) est ensuite simulé afin de valider la détection de la faute ciblée et de déterminer les autres fautes détectées qui n'étaient pas explicitement ciblées lors de la génération de ce vecteur.

Les modules combinatoires soumis aux ATPG présentent une très faible profondeur logique (nombre de portes présentes entre les entrées et les sorties du module, c'est-à-dire entre les chaînes de scan) afin de répondre à des fréquences de fonctionnement de plus en plus élevées [WANG05]. Dans ces conditions, les vecteurs générés pour le test de ces modules présentent généralement un nombre élevé de bits à 'X'. Les études présentées dans [WOHL03, HIRA03] ont montré que les séquences générées pour le test de plusieurs circuits industriels contenaient de 95% à 99% de bits non spécifiés. La Figure III.1 donne un exemple du pourcentage de bits spécifiés dans une séquence générée pour le test d'un circuit industriel composé de plus de 800k portes logiques [BUTL04]. Elle indique pour cette séquence la proportion de vecteurs de test composée d'une certaine fraction de bits spécifiés.

Cette figure montre que la majorité des vecteurs de la séquence de test (plus de 70%) est constituée de moins de 1% de bits spécifiés, ce qui se traduit par une fraction de plus de 99% de bits non spécifiés dans ces vecteurs. Parallèlement, cette figure montre que les vecteurs contenant de 25% à 50% de bits spécifiés représentent moins de 2% du nombre total de vecteurs de la séquence de test. Dans ces conditions, nous voyons clairement que le nombre important de bits non spécifiés peut être exploité afin de limiter la consommation de puissance durant le test.

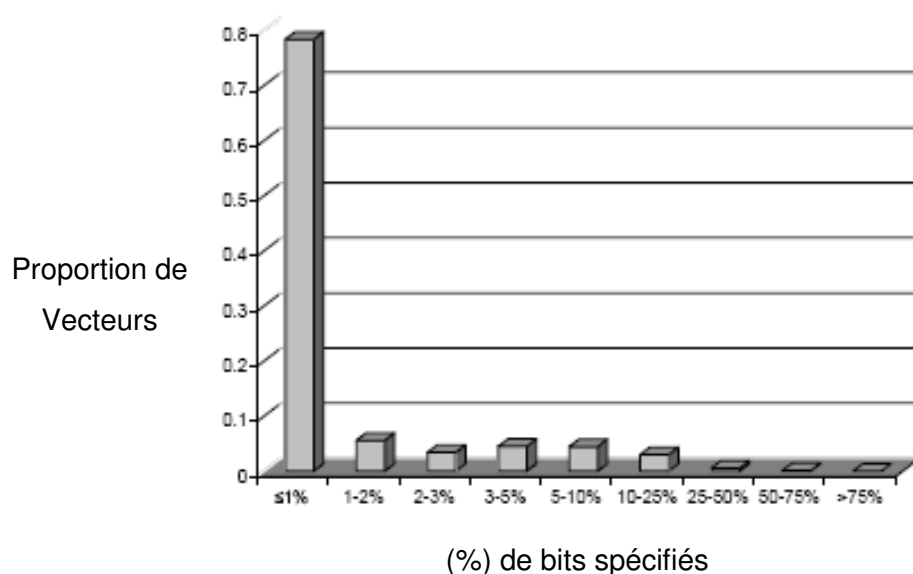


Figure III.1 Pourcentage de bits spécifiés dans les vecteurs de test

L'affectation aléatoire de bits non spécifiés par les outils ATPG peut conduire à une forte activité de commutation dans la chaîne de scan et par conséquent à une consommation de puissance excessive dans le circuit sous test. Afin de réduire la consommation de puissance de pic pendant le cycle de test, nous avons proposé une technique d'assignation de bits non spécifiés qui est présentée dans la partie suivante. Pour montrer l'influence de l'assignation de bits non spécifiés sur l'activité de commutation dans la chaîne de scan pendant le cycle de test, considérons le vecteur de test "0XXX1XX0XX0XX". Supposons qu'une assignation aléatoire des bits non spécifiés de ce vecteur de test par un outil ATPG a donné le résultat suivant "0101101001001". Le nombre de transitions (différences de bits) dans ce vecteur de test est égal à 9. Maintenant, si tous les bits non spécifiés sont assignés à '0' ('1'), le vecteur de test résultant "0000100000000" ("0111111011011") contiendrait 2 (5) transitions.

Nous remarquons clairement que le nombre de transitions dans ce vecteur de test change suivant la technique d'assignation utilisée. Ce nombre est passé de 9 dans le cas d'une assignation aléatoire à 2 (5) avec une assignation à '0' ('1'). Sachant que le nombre de transitions dans un vecteur de test a un impact direct sur la consommation de puissance pendant le cycle de test (cf. partie I.5), nous pouvons dire que l'assignation de tous les bits non spécifiés de ce vecteur de test à '0' entraînera une réduction de la consommation de puissance pendant le cycle de test comparée à une assignation aléatoire par un outil ATPG.

La partie suivante présente les techniques classiques d'assignation de bits non spécifiés proposées dans [BUTL04] et la technique que nous avons développée dans le cadre de cette thèse, ainsi que les résultats obtenus avec l'application de ces techniques.

3.3 Techniques d'assignation de bits non spécifiés

Afin de réduire la consommation de puissance de pic pendant le cycle de test, l'idée est d'utiliser un processus de génération de séquences de test durant lequel les bits à 'X' de chaque vecteur de test ne sont pas affectés aléatoirement par l'outil ATPG. Ensuite, nous affectons ces bits à 'X' à des valeurs spécifiques à l'aide de plusieurs techniques d'assignation.

Lors de l'application de ces techniques d'assignation, nous avons pris en compte deux paramètres importants (en plus de la réduction de puissance) : l'impact sur le volume de données de test et sur le taux de couverture de fautes.

En effet, lors d'une assignation non aléatoire des bits à 'X' d'un vecteur d'une séquence de test, le volume total de données dans cette séquence peut augmenter car un mauvais choix des valeurs d'assignation peut limiter la capacité de ce vecteur à détecter des fautes non ciblées. Ceci entraînera une augmentation du nombre total de vecteurs dans cette séquence si l'on désire garder le même taux de couverture de fautes. A l'inverse, un bon choix des valeurs logiques pour une affectation non aléatoire des bits à 'X' d'une séquence de test peut réduire le nombre total de vecteurs dans cette séquence. De la même manière, en ce qui concerne le taux de couverture de fautes, une affectation non aléatoire des bits à 'X' d'une séquence de test peut réduire ou augmenter le nombre total de fautes détectées par cette séquence, et par conséquent conduire à une amélioration ou à une dégradation du taux de couverture de fautes.

3.3.1 Techniques classiques d'assignation

L'objectif de cette étude consiste à réduire la consommation de puissance de pic pendant le cycle de test par l'assignation des bits à 'X' d'une séquence de test donnée à des valeurs logiques fixes '0' ou '1'. Les bits non spécifiés de chaque vecteur de test composant cette séquence sont affectés à l'aide des trois techniques suivantes :

- Minimum Transition Filling (appelée aussi Adjacent Filling) : Chaque bit à 'X' est affecté à la valeur du dernier bit spécifié rencontré (en décrivant le vecteur de gauche à droite).
- 0-Filling : Tous les bits à 'X' sont affectés à '0'.
- 1-Filling : Tous les bits à 'X' sont affectés à '1'.

Considérons le vecteur de test "0XXX1XX0XX0XX". Si nous appliquons les trois techniques précédentes à ce vecteur de test, le résultat sera comme suit :

- 00001110000000 avec MT-Filling ("0XXX1XX0XX0XX"),
- 00001000000000 avec 0-Filling,
- 01111110110111 avec 1-Filling.

Ces trois techniques d'assignation 0, 1 et MT-Filling ont été proposées dans [BUTL04]. Les auteurs dans [BUTL04] ont évalués ces techniques dans leur capacité à réduire *l'activité de commutation moyenne* pendant *les cycles de décalage* du test par scan de plusieurs circuits industriels. Les séquences de test utilisées ont été générées pour la détection *de fautes de transition*. Les résultats ont montré que la technique MT-Filling apporte la meilleure réduction de l'activité de commutation moyenne dans le circuit sous test pendant les cycles de décalage. Cette réduction est obtenue avec une faible augmentation du nombre de vecteurs dans la séquence de test résultante (0,98% à 2,72%) par rapport à une séquence standard (les bits à 'X' sont assignés de manière aléatoire par l'outil ATPG).

Dans notre étude, nous avons évalué ces mêmes techniques dans leur capacité à réduire *la consommation de puissance de pic* pendant *le cycle de test* par rapport à une séquence standard générée pour la détection *de fautes de collage*. Pour ce faire, nous avons développé la procédure présentée sur la Figure III.2 [BADE05b, BADE06d].

Cette procédure commence par la génération à l'aide d'un outil ATPG d'une séquence de test sans affecter les bits non spécifiés. Ensuite, pour chaque vecteur de cette séquence de test, les bits à 'X' sont affectés à l'aide des trois techniques d'assignation 0, 1 et MT-Filling. Une simulation de fautes est ensuite exécutée afin d'éliminer les vecteurs de test de la séquence résultante qui ne détectent pas de fautes. Après, la procédure évalue la puissance de pic consommée dans le circuit pendant le cycle de test, pendant les cycles de décalage, ainsi que le calcul du taux de couverture de fautes et la longueur de la séquence de test résultante (nombre de vecteurs dans la séquence). La dernière étape consiste à évaluer les réductions de puissance de pic pendant le cycle de test puis pendant les cycles de décalage, ainsi que l'impact sur les taux de couverture de fautes et sur le volume de données de test par rapport à une séquence de test standard.

```

- Générer par un outil ATPG une séquence de test sans affecter les bits à 'X' ;
- Pour chaque vecteur de la séquence de test {
    Affecter tous les bits à 'X' par la technique
        MT-Filling ;
        0-Filling ;
        1-Filling ;
    Sauvegarder les nouveaux vecteurs ;
}
- Exécuter une simulation de fautes afin d'éliminer les vecteurs non nécessaires;
- Evaluer la puissance de pic durant le cycle de test et les durant les cycles de
décalage, le taux de couverture de fautes et le volume de données de test ;
- Evaluer les réductions de puissance de pic pendant le cycle de test puis pendant
les cycles de décalages, et l'impact sur le taux de couverture de fautes et sur le
volume de données de test par rapport à une séquence de test standard;

```

Figure III.2 Procédure d'évaluation des techniques 0, 1 et MT-Filling

Les résultats obtenus avec l'application de ces trois techniques d'assignation de bits non spécifiés sont présentés dans la partie suivante.

3.3.2 Résultats expérimentaux

Dans cette partie, nous présentons les résultats comparatifs entre les techniques d'assignation 0, 1 et MT-Filling en terme de réduction de puissance de pic pendant le cycle de test. D'autres résultats concernant la réduction de puissance de pic pendant les cycles de décalage ainsi que l'impact de ces techniques sur le taux de couverture de fautes et sur le volume de données de test sont présentés dans la partie suivante.

Pour chaque circuit, nous avons commencé par générer une séquence de test standard ciblant la détection de fautes de collage avec l'outil TetraMax [SYNO01]. Dans cette séquence de test, les bits non spécifiés sont affectés de manière aléatoire. Ensuite, nous avons généré une deuxième séquence de test avec l'outil TetraMax sans l'assignation des bits non spécifiés (séquence partiellement spécifiée). A cette séquence, nous avons appliqué les trois techniques d'assignation 0, 1 et MT-Filling. Les séquences résultantes de ces assignations non aléatoires ont été appliquées à l'outil TetraMax pour une simulation de fautes afin d'éliminer les vecteurs de test non nécessaires. Cette dernière étape permet de récupérer les séquences finales qui doivent être utilisées pour l'estimation de la consommation de puissance dans le circuit.

Le tableau III.1 présente les caractéristiques des séquences partiellement spécifiées générées par l'outil TetraMax pour le test des circuits expérimentés. La première colonne

indique le nom du circuit. La deuxième colonne reprend le nombre de vecteurs dans la séquence de test avec une assignation aléatoire par l'outil TetraMax (cf. partie 1.5.1). La troisième colonne présente le nombre de vecteurs dans la séquence de test sans l'assignation des bits non spécifiés et la dernière colonne montre le pourcentage moyen de bits spécifiés dans cette séquence.

Circuit	# vecteurs avec assignation aléatoire	# vecteurs sans assignation	% de bits spécifiés dans la séquence partiellement spécifiée
b04s	55	61	35,19
b09	28	29	64,44
b10	43	44	39,12
b11s	60	65	39,71
b12	93	102	24,25
b13s	30	35	32,01
b14s	409	428	23,8
b17s	692	820	6,54
s1196	128	138	43,86
s5378	145	205	16,68
s9234	159	179	25,94
s13207	254	281	5,97
s38417	144	555	4,63

Tableau III.1 Caractéristiques des séquences de test partiellement spécifiées

Les séquences de test partiellement spécifiées présentent une moyenne de 27,85% de bits spécifiés avec un minimum de 4,63% et un maximum de 64,44%. Ces valeurs sont différentes de celles des circuits produits dans l'industrie qui présentent un pourcentage de bits spécifiés dans les séquences de test compris entre 1% et 5% [WOHL03, HIRA03]. Ceci est dû principalement, d'une part, à la grande profondeur logique dans les circuits expérimentés contrairement aux circuits industriels qui présentent une faible profondeur logique [WANG05], et d'autre part, à la faible taille des circuits expérimentés.

Notons aussi que la longueur de ces séquences est différente de celle des séquences de test standard. En moyenne, ces séquences partiellement spécifiées présentent une augmentation de 33,26% du nombre de vecteurs de test par rapport à la séquence standard. Comme cela a été expliqué dans le paragraphe 3.3, lors d'une assignation non aléatoire des bits à 'X' d'un vecteur d'une séquence de test, le volume total de données dans cette séquence peut augmenter car un mauvais choix des valeurs d'assignation peut limiter la capacité de ce vecteur à détecter des fautes non ciblées. Ceci entraînera une augmentation du nombre total de vecteurs dans cette séquence de test afin de garder le même taux de couverture de fautes.

Le Tableau III.2 regroupe les résultats de réductions de consommation de puissance de pic pendant le cycle de test en affectant les bits à 'X' avec les trois techniques d'assignation 0, 1 et MT-Filling. Les caractéristiques des circuits étudiés, les caractéristiques de la station de travail et les outils utilisés pour l'estimation de la puissance consommée dans le circuit sont identiques à ceux présentés au paragraphe I.5.1. La première colonne indique le nom du circuit. La deuxième colonne présente la moyenne des puissances de pic consommées dans le circuit sous test durant tous les cycles de test (sur toute la séquence de test) avec une séquence standard. Les colonnes 3, 4, et 5 présentent la moyenne des puissances de pic consommées dans le circuit pendant les cycles de test avec les séquences modifiées, où les bits non spécifiés sont assignés à des valeurs fixes '0' ou '1' en fonction de la technique d'assignation utilisée. Les trois dernières colonnes, présentent les réductions de puissance de pic obtenues par rapport à la séquence standard. Le temps nécessaire à ces simulations est inférieur à 1 minute.

Circuit	Puissance de Pic pendant le cycle de test (en mW)				Réduction de Puissance de Pic (en %)		
	Séquence Standard	MT-Filling	0-Filling	1-Filling	MT-Filling	0-Filling	1-Filling
b04s	44,03	31,32	22,18	40,88	28,86	49,63	7,15
b09	19,80	15,56	15,99	16,31	21,41	19,27	17,65
b10	14,74	8,79	13,67	11,29	40,36	7,28	23,38
b11s	28,43	22,59	22,98	26,28	20,55	19,17	7,57
b12	82,21	26,69	26,16	34,68	67,54	68,18	57,82
b13s	39,04	15,62	21,89	19,62	59,99	43,93	49,75
b14s	178,00	131,23	123,30	197,08	26,28	30,73	-10,72
b17s	961,86	191,07	214,09	233,35	80,14	77,74	75,74
s1196	4,71	0,97	1,97	1,60	79,33	58,24	66,15
s5378	146,97	37,90	47,71	43,03	74,21	67,54	70,72
s9234	240,02	80,39	132,89	140,04	66,51	44,63	41,65
s13207	402,62	42,33	62,61	48,46	89,49	84,45	87,96
s38417	978,10	275,82	171,30	155,34	71,80	82,49	84,12
Moyenne					55,88	50,25	44,53

Tableau III.2 Réduction de puissance de pic pendant le cycle de test

Ces résultats montrent qu'une moyenne de 55,88% de réduction de puissance de pic pendant le cycle de test peut être obtenue avec la technique d'assignation MT-Filling (50,25% pour 0-Filling et 44,53% pour 1-Filling) avec un maximum de 89,49% (84,45% pour 0-Filling et 87,96% 1-Filling). Ils montrent ainsi l'efficacité des techniques expérimentées en terme de réduction de puissance de pic durant le cycle de test. Cependant, nous remarquons que la

réduction de puissance obtenue est plus ou moins importante suivant la technique d'assignation utilisée. Pour la majorité des circuits, la technique d'assignation MT-Filling produit la meilleure réduction de puissance de pic par rapport aux deux autres techniques 0 et 1-Filling. Toutefois, cette remarque n'est pas toujours vraie, car la structure interne du circuit sous test, et en particulier le type de portes logiques directement connectées aux cellules de la chaîne de scan peut favoriser une technique par rapport à une autre.

Afin de mieux expliquer la différence des résultats obtenus avec les différentes techniques d'assignation, considérons le cas du circuit b14s. Ce circuit présente plus de 500 portes AND/NAND connectées aux cellules de la chaîne de scan et seulement 47 portes OR/NOR connectées aux cellules de la chaîne de scan. De ce fait, il était prévisible que la technique 0-Filling apporterait une meilleure réduction de puissance de pic (30,73%) par rapport aux deux autres techniques (réduction de 26,28% avec MT-Filling et augmentation de 10,72% avec 1-Filling de la puissance de pic). Ce résultat peut être expliqué par le fait que les sorties des portes AND/NAND connectées aux cellules scan sont restées stables à la même valeur logique à cause de l'application de plusieurs valeurs prioritaires sur leurs entrées (la valeur prioritaire dans ce cas est '0'), empêchant ainsi la propagation des commutations présentes dans la chaîne de scan vers la partie combinatoire du circuit sous test.

Cette remarque nous a mené à développer une nouvelle technique d'assignation de bits non spécifiés qui prend en compte la structure interne du circuit lors de l'affectation de bits à 'X'. Cette technique est présentée dans la partie suivante.

3.3.3 Assignation basée sur des informations structurelles

Dans la partie précédente, nous avons montré que l'assignation de bits non spécifiés est une solution intéressante en terme de réduction de la puissance de pic pendant le cycle de test. Cependant, les performances des trois techniques d'assignation 0, 1 et MT-Filling dépendent essentiellement du type de portes directement connectées aux cellules de la chaîne de scan. À partir de cette observation, nous avons orienté nos recherches vers le développement d'une nouvelle technique d'assignation de bits non spécifiés prenant en compte ce type d'observation lié à la structure du circuit.

La technique que nous avons développée, appelée *Structural-Based Filling*, affecte les bits à 'X' en se basant sur la structure interne du circuit sous test [BADE06c, BADE06e]. En fonction du type de portes directement connectées aux cellules de la chaîne de scan, les bits non spécifiés sont affectés de manière à bloquer la propagation des commutations présentes

dans la chaîne de scan au niveau de la première couche logique du circuit qui est connectée aux cellules de la chaîne de scan.

Considérons l'exemple de la Figure III.3. La porte G_1 a deux de ses entrées connectées aux cellules scan tandis que la porte G_2 a une seule entrée connectée aux cellules scan.

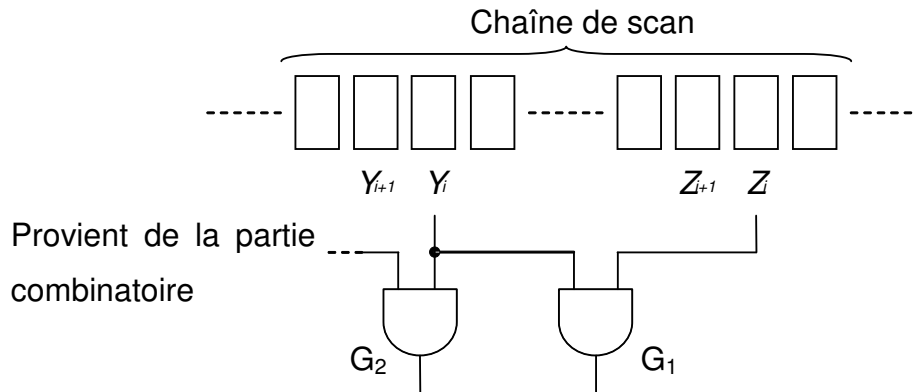


Figure III.3 Exemple de portes connectées aux cellules de la chaîne de scan

Pour garantir la stabilité de la sortie de la porte G_1 (pas de transitions), nous devons satisfaire l'une des quatre conditions suivantes :

- ✓ Une succession de '0' sur l'entrée Y ($Y_i=Y_{i+1}=0$).
- ✓ Une succession de '0' sur l'entrée Z ($Z_i=Z_{i+1}=0$).
- ✓ Une succession de '0' sur deux entrées différentes ($Y_i=0$ et $Z_{i+1}=0$ ou $Z_i=0$ et $Y_{i+1}=0$).
- ✓ Les deux entrées restent stables à '1' ($Y_i=Y_{i+1}=Z_i=Z_{i+1}=1$).

Nous avons formulé ces quatre conditions sous forme de l'équation (1).

$$\overline{Y_i} \cdot \overline{Y_{i+1}} + \overline{Z_i} \cdot \overline{Z_{i+1}} + \overline{Y_i} \cdot \overline{Z_{i+1}} + \overline{Z_i} \cdot \overline{Y_{i+1}} + Y_i \cdot Y_{i+1} \cdot Z_i \cdot Z_{i+1} = 1 \quad (1)$$

Chaque terme de cette équation correspond à la succession de valeurs aux entrées de la porte G_1 qui garantissent la stabilité de sa sortie. Les deux premiers termes $\overline{Y_i} \cdot \overline{Y_{i+1}}$ et $\overline{Z_i} \cdot \overline{Z_{i+1}}$ correspondent à la succession de deux '0' sur la même entrée de la porte G_1 qui permet d'assurer la stabilité de sa sortie à '0'. Les deux suivants $\overline{Y_i} \cdot \overline{Z_{i+1}}$ et $\overline{Z_i} \cdot \overline{Y_{i+1}}$ correspondent à la succession de deux '0' sur deux entrées différentes de G_1 qui garantissent la stabilité de sa sortie à '0' (sachant que le décalage de valeurs dans la chaîne de scan se fait de gauche à droite). Le dernier terme $Y_i \cdot Y_{i+1} \cdot Z_i \cdot Z_{i+1}$ signifie que toutes les entrées de la porte G_1 restent

stables à '1' pour assurer la stabilité de sa sortie à '1'.

D'un point de vu technique, il est plus facile de traiter une équation sous forme de produit de sommes qu'une équation sous forme de somme de produits. Pour cela, nous avons réécrit l'équation (1) en produit de sommes, où chaque terme, appelé *clause*, doit être satisfaite pour garantir la stabilité de la sortie de la porte considérée. L'équation (1) devient alors :

$$(\overline{Y}_i + \overline{Z}_i + Y_{i+1}) \cdot (\overline{Y}_i + \overline{Z}_i + Z_{i+1}) \cdot (\overline{Y}_{i+1} + \overline{Z}_{i+1} + Y_i) \cdot (\overline{Y}_{i+1} + \overline{Z}_{i+1} + Z_i) = 1 \quad (2)$$

Notons que dans le cas où la porte appartenant à la première couche logique est connectée à une seule cellule scan (porte G_2), l'équation à satisfaire est :

$$\overline{Y}_i \cdot \overline{Y}_{i+1} = 1 \quad (3)$$

ceci qui correspond à la succession de deux '0' sur la même entrée de la porte G_2 (qui est connectée à la chaîne de scan) qui garantissent la stabilité de sa sortie à '0'.

Pour une analyse complète du circuit sous test, nous devons établir l'équation correspondant à chaque porte connectée à la chaîne de scan. Notons par ailleurs que si la porte connectée à la chaîne de scan est un inverseur, nous considérons la porte connectée à la sortie de cet inverseur en inversant la valeur logique de sa sortie, et que les portes de type XOR/NXOR sont transformées en portes primitives.

Le Tableau III.3 présente l'exemple des équations de satisfiabilité pour des portes primitives avec une ou deux entrées connectées aux cellules scan. Des équations similaires peuvent être établies pour des portes avec plus de deux entrées connectées aux cellules scan.

Type de portes	Nombre d'entrées connectées à la chaîne de scan	Équations de satisfiabilité garantissant la stabilité de la sortie de la porte
AND / NAND	2	$(\overline{Y}_i + \overline{Z}_i + Y_{i+1}) \cdot (\overline{Y}_i + \overline{Z}_i + Z_{i+1}) \cdot (\overline{Y}_{i+1} + \overline{Z}_{i+1} + Y_i) \cdot (\overline{Y}_{i+1} + \overline{Z}_{i+1} + Z_i) = 1$
	1	$\overline{Y}_i \cdot \overline{Y}_{i+1} = 1$
OR / NOR	2	$(Y_i + Z_i + \overline{Y}_{i+1}) \cdot (Y_i + Z_i + \overline{Z}_{i+1}) \cdot (Y_{i+1} + Z_{i+1} + \overline{Y}_i) \cdot (Y_{i+1} + Z_{i+1} + \overline{Z}_i) = 1$
	1	$Y_i \cdot Y_{i+1} = 1$

Tableau III.3 Équations de satisfiabilité pour les portes AND/NAND – OR/NOR avec une ou deux entrées connectées aux cellules de la chaîne de scan

La technique SB-Filling commence par le calcul des équations de toutes les portes connectées aux cellules de la chaîne de scan. Ensuite, le problème d'optimisation à considérer

est de satisfaire le nombre maximum d'équations afin d'assurer la stabilité de la sortie d'un maximum de portes. Ce problème est similaire à un problème de satisfiabilité [GARE78]. L'utilisation d'un algorithme de satisfiabilité [CAST96, SPEA96] n'est pas envisageable à cause de la faible probabilité de garantir la vérification de toutes les clauses des équations de toutes les portes connectées à la chaîne de scan. Aussi, nous avons proposé une autre solution qui garantit la stabilité de la sortie d'un maximum de portes. Comme ce problème d'optimisation est NP-Difficile, nous avons utilisé l'algorithme glouton présenté dans la Figure III.4.

A partir des équations correspondant aux portes connectées à chaque cellule de la chaîne de scan, la boucle interne de l'algorithme est exécutée pour chaque vecteur de la séquence de test. L'algorithme utilise d'abord les bits spécifiés du vecteur de test afin d'actualiser l'équation de chaque porte. Ensuite, il commence par le traitement du premier bit à 'X' du vecteur de test considéré. Ce bit à 'X' est affecté à '0' ensuite à '1'. L'algorithme calcule alors dans les deux cas le nombre *d'équations satisfaites* (n), le nombre *d'équations restantes* (p) et le nombre de *clauses totales restantes* dans toutes les équations (nc).

A partir de ces trois paramètres, l'affectation de ce bit à 'X' se fait, successivement, selon les quatre règles suivantes :

1. Maximiser le nombre d'équations satisfaites (n).
2. Maximiser le nombre d'équations restantes (p).
3. Minimiser le nombre de clauses totales restantes (nc).
4. Si aucune solution n'a été trouvée, remplacer le bit à 'X' par la valeur du bit précédent dans la chaîne de scan.

Ensuite, les équations sont actualisées avec la nouvelle valeur du bit à 'X' considéré. Lorsque tous les bits à 'X' sont affectés, la séquence résultante est simulée afin d'éliminer les vecteurs de test non nécessaires (qui ne détectent pas de fautes). Cette séquence est ensuite utilisée pour l'évaluation de la consommation de puissance de pic durant le cycle de test, durant les cycles de décalage, ainsi que pour le calcul du taux de couverture de fautes et la longueur des séquences de test résultantes. La dernière étape consiste à évaluer les réductions de puissance de pic pendant le cycle de test, pendant les cycles de décalage, ainsi que l'impact sur les taux de couverture de fautes et sur le volume de données de test par rapport à une séquence standard.

```

- Déterminer les portes connectées aux cellules de la chaîne de scan ;
- Calculer l'équation de chaque porte ;
- Générer une séquence de test sans affecter les bits à 'X' ;
- Pour chaque vecteur de la séquence de test {
  Actualiser les équations avec les valeurs des bits spécifiés ;
  Pour chaque bit à 'X' {
    Affecter '0' à 'X' {
      Calculer le nombre d'équations satisfaites (n0) ;
      Calculer le nombre d'équations restantes (p0) ;
      Calculer le nombre de clauses totales restantes (nc0) ;
    }
    Affecter '1' à 'X' {
      Calculer (n1) (p1) (nc1) ;
    }
    si      (n0 > n1)      Remplacer 'X' par '0' ;
    sinon si (n1 > n0)      Remplacer 'X' par '1' ;
    sinon si (p0 > p1)      Remplacer 'X' par '0' ;
    sinon si (p1 > p0)      Remplacer 'X' par '1' ;
    sinon si (nc0 > nc1)    Remplacer 'X' par '1' ;
    sinon si (nc1 > nc0)    Remplacer 'X' par '0' ;
    sinon                                     Remplacer 'X' par la valeur du bit
                                                précédent dans la chaîne de scan ;
    Actualiser les équations ;
  }
}
- Exécuter une simulation de faute pour éliminer les vecteurs non nécessaires ;
- Evaluer la puissance de pic durant le cycle de test, durant les cycles de décalage, le
  taux de couverture de fautes et le volume de données de test ;
- Evaluer les réductions de puissance de pic et l'impact sur le taux de couverture de
  fautes et sur le volume de données de test par rapport à une séquence standard ;

```

Figure III.4 Procédure du déroulement de la technique SB-Filling

Les résultats comparatifs entre la technique SB-Filling et les autres techniques d'assignation de bits non spécifiés 0, 1 et MT-Filling sont présentés dans la partie suivante.

3.3.4 Résultats expérimentaux

L'objectif des résultats présentés dans cette partie est de comparer l'approche SB-Filling aux techniques 0, 1 et MT-Filling. Cette comparaison permettra de sélectionner la technique d'assignation qui apporte le meilleur résultat, en prenant en compte la réduction de consommation de puissance de pic pendant le test, le volume de données de test et le taux de couverture de fautes.

Nous avons appliqué la technique SB-Filling à une séquence de test où les bits non spécifiés ne sont pas affectés aléatoirement par l'outil TetraMax. Ensuite, nous avons évalué

la consommation de puissance de pic pendant le cycle de test et pendant les cycles de décalage dans le circuit sous test avec la séquence résultante. Nous avons aussi calculé le taux de couverture de fautes (fautes de collage et de transition) et le volume de données de test. Le temps de simulation ne dépasse pas 2 minutes dans tous les cas.

Le tableau III.4 présente le nombre d'équations traitées par la technique SB-Filling (nombre de portes appartenant à la première couche logique) pour chaque circuit expérimenté. La première colonne indique le nom du circuit. La deuxième colonne présente le nombre de cellules dans la chaîne de scan et la dernière colonne indique le nombre d'équations à satisfaire.

Circuit	# cellules scan	# équations à satisfaire
b04s	66	150
b09	28	75
b10	17	54
b11s	31	122
b12	121	327
b13s	53	116
b14s	245	604
b17s	1415	7644
s1196	18	19
s5378	179	348
s9234	228	425
s13207	669	635
s38417	1636	822

Tableau III.4 Nombre d'équations à satisfaire

Nous remarquons que le nombre d'équations à satisfaire change selon le circuit étudié. Ce nombre est directement proportionnel au nombre de cellules dans la chaîne de scan. Cependant, le nombre d'équations à satisfaire est pour la majorité des circuits expérimentés supérieur au nombre de cellules scan, car chaque cellule peut être connectée à plusieurs portes de la première couche logique. Ce nombre d'équations à traiter par la technique SB-Filling montre ainsi la complexité du problème d'optimisation à considérer (jusqu'à 7644 équations à satisfaire).

3.3.4.1 Réduction de puissance de pic pendant le cycle de test

Le Tableau III.5 présente les résultats comparatifs en terme de réduction de consommation de puissance de pic pendant le cycle de test entre les différentes techniques d'assignation développées. La première colonne indique le nom du circuit. La deuxième

colonne présente la moyenne des puissances de pic consommées pendant les cycles de test avec la technique SB-Filling. Les quatre colonnes suivantes présentent le pourcentage de réduction obtenu pour chaque circuit avec les différentes techniques par rapport à une séquence standard fournie par l'outil TetraMax. Les deux dernières lignes présentent la moyenne des réductions de consommation de puissance de pic pendant le cycle de test de tous les circuits et l'écart type (dispersion des réductions de puissance de pic obtenues pour tous les circuits par rapport à la moyenne de ses puissances de pic).

Circuit	Puissance de pic (en mW) SB-Filling	Réduction de puissance de pic pendant le cycle de test (en %)			
		SB-Filling	MT-Filling	0-Filling	1-Filling
b04s	22,44	44,5	28,86	49,63	7,15
b09	15	24,3	21,41	19,27	17,65
b10	8,93	39,4	40,36	7,28	23,38
b11s	22,48	20,9	20,55	19,17	7,57
b12	24,59	70,1	67,54	68,18	57,82
b13s	18,23	53,3	59,99	43,93	49,75
b14s	129,62	27,2	26,28	30,73	-10,72
b17s	203,46	78,9	80,14	77,74	75,74
s1196	1,59	66,3	79,33	58,24	66,15
s5378	37,9	74,2	74,21	67,54	70,72
s9234	134,99	43,8	66,51	44,63	41,65
s13207	59,16	85,3	89,49	84,45	87,96
s38417	465,11	52,5	71,80	82,49	84,12
Moyenne		52,36	55,88	50,25	44,53
Ecart type		20,55	23,87	24,61	31,35

Tableau III.5 Réduction de puissance de pic pendant le cycle de test

A partir de ces résultats, nous remarquons que la technique SB-Filling atteint 52,36% en moyenne de réduction de puissance de pic pendant le cycle de test, tandis que la technique MT-Filling atteint 55,88%. Cependant, pour chaque circuit, la technique SB-Filling apporte une réduction proche ou égale à la meilleure obtenue avec les trois autres techniques d'assignation avec le minimum d'écart type (20,55 contre 23,87, 24,61 et 31,35 pour MT, 0 et 1-Filling respectivement). En d'autres termes, la technique SB-Filling garantit une bonne réduction de consommation de puissance de pic pendant le cycle de test pour chaque circuit contrairement aux réductions obtenues avec les autres techniques qui sont plus variables d'un circuit à l'autre (dépendance de la structure du circuit).

3.3.4.2 Réduction de puissance de pic pendant les cycles de décalage

Le Tableau III.6 regroupe les résultats de réduction de puissance de pic pendant les cycles de décalage pour les circuits expérimentés. Pour chaque circuit, ces résultats montrent le pourcentage de réduction de consommation de puissance de pic pendant les cycles de décalage avec l'application des quatre techniques d'assignation par rapport à une séquence de test standard. La dernière ligne montre la moyenne des réductions de consommation de puissance de pic pendant les cycles de décalage de tous les circuits. Nous n'avons pas réalisé de simulations pour les trois grands circuits b17s, s13207 et s38417 pour des raisons de temps (qui est assez long) pour l'estimation de la consommation de puissance dans ces circuits avec l'outil PowerMill [SYNO00].

Circuit	Réduction de puissance de pic pendant les cycles de décalage (en %)			
	SB-Filling	MT-Filling	0-Filling	1-Filling
b04s	28,87	20,65	33,73	12,42
b09	19,69	19,66	14,64	18,71
b10	17,73	20,75	11,33	12,39
b11s	18,78	10,87	19,13	2,46
b12	58,68	58,43	57,17	51,05
b13s	45,73	39,72	42,79	32,56
b14s	37,24	19,53	42,44	6,31
s1196	6,66	7,84	9,35	11,09
s5378	52,66	52,58	45,04	57,24
s9234	31,97	34,06	31,33	30,51
Moyenne	31,80	28,41	30,70	23,47

Tableau III.6 Réduction de puissance de pic pendant les cycles de décalage

Ces résultats montrent que la technique SB-Filling apporte la meilleure réduction de consommation de puissance de pic pendant les cycles de décalage avec en moyenne 31,8% de réduction par rapport à la séquence standard (28,41%, 30,7%, et 23,47% pour MT, 0 et 1-Filling respectivement). En d'autres termes, la technique SB-Filling permet davantage d'éviter des pics de puissance successifs durant la phase de chargement/déchargement lors du test par scan.

3.3.4.3 Impact sur le volume de données de test

Le Tableau III.7 présente le nombre de vecteurs de test obtenus pour chaque circuit après l'affectation de bits non spécifiés par les quatre techniques d'assignation présentées dans ce chapitre. Pour chaque circuit, ce tableau présente la longueur de la séquence de test

standard (nombre de vecteurs de test dans la séquence) et la longueur de la séquence de test obtenue après l'application des quatre techniques d'assignation SB, MT, 0 et 1-Filling. La dernière ligne indique le pourcentage moyen d'augmentation de la longueur de la séquence de test par rapport à la séquence standard. Les taux de couverture de fautes obtenus avec les séquences modifiées sont présentés dans la partie suivante.

Circuit	Nombre de vecteurs dans la séquence de test				
	Séquence Standard	SB-Filling	MT-Filling	0- Filling	1-Filling
b04s	55	54	55	56	54
b09	28	28	28	28	28
b10	43	43	43	43	42
b11s	60	59	60	61	59
b12	93	100	100	99	99
b13s	30	32	33	35	33
b14s	409	416	414	415	415
b17s	692	719	721	739	738
s1196	128	131	131	134	131
s5378	145	164	167	165	193
s9234	159	165	168	174	160
s13207	254	255	254	270	263
s38417	144	163	164	343	362
Augmentation de la séquence de test		3,97%	4,37%	14,38%	15,04%

Tableau III.7 Résultats comparatifs de la longueur des séquences de test

Ces résultats montrent qu'en plus de la réduction de consommation de puissance de pic pendant le cycle de test et pendant les cycles de décalage, l'application de la technique SB-Filling fournit la séquence de test avec le minimum d'augmentation du nombre de vecteurs, avec une moyenne de 3,97% d'augmentation par rapport à la séquence standard comparée aux trois autres techniques (4,37%, 14,38% et 15,04% pour MT, 0 et 1-Filling respectivement).

3.3.4.4 Impact sur le taux de couverture

Afin de mesurer l'impact des techniques d'assignation sur le taux de couverture de fautes de collage et de transition, nous avons réalisé d'autres simulations pour mesurer ces deux paramètres avec l'application de ces techniques pour chaque circuit. Le Tableau III.8 regroupe les résultats obtenus en terme du taux de couverture de fautes de collage. La dernière ligne présente la moyenne de ces taux de couverture de fautes obtenus pour tous les circuits.

Circuit	Taux de couverture de fautes de collage (en %)				
	Séquence Standard	SB-Filling	MT-Filling	0-Filling	1-Filling
b04s	99,14	99,11	99,14	99,02	99,26
b09	100	100	100	100	100
b10	100	100	100	100	100
b11s	100	97,91	97,96	97,87	98,14
b12	100	99,98	99,93	99,85	99,99
b13s	100	98,02	97,12	97,04	97,24
b14s	99,52	98,3	98,29	98,15	98,55
b17s	98,99	98,51	98,22	98,11	98,24
s1196	100	100	100	100	100
s5378	100	99,27	99,67	99,24	99,35
s9234	99,76	94,67	94,39	93,95	94,12
s13207	99,99	99,24	99,30	99,15	99,12
s38417	100	99,81	99,76	99,66	99,52
Moyenne	99,8%	98,83%	98,75%	98,62%	98,73%

Tableau III.8 Résultats comparatifs du taux de couverture de fautes de collage

Ces résultats montrent que pour la majorité des circuits expérimentés, le taux de couverture de fautes de collage obtenu avec l'application des différentes techniques d'assignation est inférieur à celui obtenu avec la séquence de test standard (une réduction de 1% en moyenne). Cependant, si nous voulons atteindre le même taux de couverture de fautes de collage avec l'application de ces techniques d'assignation que celui obtenu avec la séquence standard, une des solutions possibles est d'ajouter de nouveaux vecteurs à la séquence de test qui cibleront d'autres fautes de collage (ce qui augmentera le volume de données de test). Une deuxième solution est d'établir un compromis entre la réduction de consommation de puissance de pic et le taux de couverture de fautes de collage obtenu. Dans cette deuxième solution, une partie des bits non spécifiés d'un vecteur de test sera remplacée de façon aléatoire afin d'augmenter la potentialité pour ce vecteur de détecter des fautes non ciblées.

Nous avons aussi mesuré l'impact de ces techniques d'assignation sur le taux de couverture de fautes de transition. Pour cela, nous avons procédé de la manière suivante. Nous avons commencé par récupérer la séquence finale après l'application des quatre techniques d'assignation et une simulation de fautes dans l'outil TetraMax afin d'éliminer les vecteurs non nécessaires. Ensuite, nous avons généré une nouvelle séquence (par des outils de programmation) contenant toutes les paires de vecteurs scan correspondant au dernier cycle de décalage de chaque vecteur de test composant la séquence finale (cette procédure est décrite dans le paragraphe 2.4.2.3).

Le Tableau III.9 présente les résultats comparatifs entre les différentes techniques d'assignation en terme de taux de couverture de fautes de transition non robuste par rapport à une séquence de test standard. Cette séquence de standard a été générée par l'outil TetraMax pour détecter les fautes de collage. La dernière ligne de ce tableau présente la moyenne de ces taux de couverture de fautes obtenus pour tous les circuits.

Circuit	Taux de couverture de fautes de transition non robuste (en %)				
	Séquence Standard	SB-Filling	MT-Filling	0-Filling	1-Filling
b04s	95,83	70,25	71,32	69,75	77,03
b09	94,69	80,46	81,08	76,66	81,75
b10	96,88	68,11	67,65	68,2	69,49
b11s	93,6	81,13	80,69	77,2	80,35
b12	95,41	77,86	77,99	73,53	77,56
b13s	94,2	68,14	67,5	68,2	74,76
b14s	97,05	79,48	82,21	75,84	84,08
b17s	95,41	75,59	74,85	70,7	71,63
s1196	94,73	13,75	7,02	14,83	13,73
s5378	97,43	63,99	64,58	68,69	67,29
s9234	92,36	67,12	64,49	66,82	67,08
s13207	84,48	74,85	68,91	75,56	74,71
s38417	90,14	85,94	77,14	79,66	82,77
Moyenne	94,04%	69,74%	68,11%	68,13%	70,94%

Tableau III.9 Résultats comparatifs du taux de couverture de fautes de transition non robuste

Ces résultats montrent une nette dégradation du taux de couverture de fautes de transition non robuste. Les taux de couverture obtenu avec la séquence de test standard sont passés d'une moyenne de 94,04% à une moyenne de l'ordre de 70% avec les séquences de test générées par les quatre techniques d'assignation développées. Dans ce cas, nous devons faire un compromis entre la réduction de puissance (réduction de l'activité de commutation) et la détection de fautes de transition (augmentation du nombre de transitions dans chaque vecteur de test). Cependant, comme pour le taux de couverture de fautes de collage, pour améliorer le taux de couverture de fautes de transition nous pouvons procéder de plusieurs façons. Nous pouvons ajouter d'autres vecteurs à la séquence de test afin d'améliorer la couverture de fautes de transition, ce qui augmentera le volume de données de test. Nous pouvons faire un compromis entre la réduction de puissance de pic et la détection de fautes de transition en affectant quelques bits non spécifiés de la séquence de test à des valeurs logiques '0' ou '1' dans le but de détecter plus de fautes de transition. Nous pouvons aussi générer une autre séquence de test qui ciblera plutôt la détection de ce type de fautes.

3.4 Conclusion

Dans ce chapitre, nous avons analysé trois techniques d'assignation de bits non spécifiés 0, 1 et MT-Filling permettant de réduire la consommation de puissance de pic dans le circuit durant le cycle de test. Les résultats obtenus avec ces techniques dépendent de la structure interne du circuit sous test, et en particulier du type de portes directement connectées aux cellules de la chaîne de scan. Dans le but d'améliorer les résultats obtenus avec ces trois techniques, nous avons proposé une nouvelle technique d'affectation de bits non spécifiés basée sur des informations structurelles.

Cette technique se base sur des informations structurelles du circuit sous test pour affecter les bits non spécifiés. En fonction du type des portes directement connectées aux cellules de la chaîne de scan, la technique SB-Filling affecte les bits à 'X' de façon à bloquer la propagation des commutations présentes dans la chaîne de scan vers le circuit sous test en vue d'une réduction de la consommation de puissance lors du test.

Cette technique présente de nombreux avantages. Elle permet de réaliser le meilleur compromis entre la réduction de consommation de puissance de pic et l'augmentation du volume de données de test par rapport aux autres techniques d'assignation. Elle permet aussi de ne pas dégrader fortement le taux de couverture de fautes de collage par rapport à celui obtenu avec une séquence de test standard.

De plus, contrairement aux solutions de réordonnement des cellules de la chaîne de scan présentées dans le chapitre précédent, la technique SB-Filling apporte une réduction significative de la consommation de puissance de pic pendant les cycles de décalage (en moyenne 32%), ce qui permet d'éviter les problèmes liés à une forte température dans le circuit sous test pendant cette phase. D'autre part, elle permet d'éliminer le problème du routage des cellules de la chaîne de scan qui est dû au réordonnement de ces cellules. Cette technique ne nécessite l'ajout d'aucune logique supplémentaire ni la modification du flot de conception.

Un des inconvénients de cette technique est la dégradation du taux de couverture de fautes de transition. Toutefois, ce taux de couverture de fautes peut être amélioré par l'ajout de nouveaux vecteurs à la séquence de test afin de cibler la détection d'autres fautes de transition, ou de générer la séquence de test initiale dans le but de détecter ce type de fautes. Une autre façon d'améliorer ce taux de couverture de fautes est d'affecter quelques bits non spécifiés des vecteurs de la séquence de test initiale à des valeurs logiques fixes afin de

détecter plus de fautes de transition, mais cela limitera la réduction de la consommation de puissance.

Un autre inconvénient de la technique SB-Filling est qu'en plus du fait qu'elle augmente le volume final de données de test, elle ne permet pas d'avoir un bon taux de compression de ces données. En effet, la majorité des techniques de compression se basent sur l'assignation des bits non spécifiés de la séquence de test initiale afin de réduire le volume total de ses données. La technique présentée dans le chapitre suivant permet de répondre à ce problème.

Chapitre IV

Réduction de puissance et compression de données de test

Chapitre IV : Réduction de puissance et compression de données de test *

4.1 Introduction

Le coût du test d'un circuit intégré peut être minimisé en optimisant l'exploitation du testeur utilisé. La compression de données est une solution particulièrement intéressante qui répond à ce problème d'optimisation. Cette solution consiste à compresser la séquence de test générée par un outil ATPG en une séquence plus petite en terme de données de test. La séquence compressée est ensuite stockée dans la mémoire du testeur. Un décodeur inséré dans le circuit sous test permet de régénérer la séquence originale à partir de la séquence compressée lors du test. Ce type de solution permet donc une réduction du volume de données de test, conduisant à l'utilisation d'un testeur ayant un faible nombre de canaux et une capacité mémoire plus faible.

La première partie de ce chapitre est consacrée à la présentation du principe de la compression de données de test. La deuxième partie propose un état de l'art des techniques combinant la compression de données et la réduction de puissance pendant le test. La troisième partie présente la technique de compression proposée dans [WANG05] à partir de laquelle nous avons travaillé afin de combiner réduction de puissance et compression de données. Dans la quatrième partie, nous détaillerons le problème de consommation de puissance avec cette technique, ainsi que la modification que nous lui avons apportée afin de réduire simultanément le volume de données et la consommation de puissance. Nous présenterons également des résultats comparatifs entre la technique de compression initiale et la technique de compression proposée pour plusieurs circuits académiques et industriels.

* Ce travail a été réalisé dans le cadre d'une collaboration avec le Professeur Krishnendu CHAKRABARTY de l'Université de Duke aux Etats-Unis.

4.2 Compression de données de test

A l'heure actuelle, le coût de fabrication des circuits intégrés est largement dominé par le coût de leur test [POIR04]. Ce dernier peut être réduit en optimisant l'exploitation du testeur utilisé. Cependant, deux facteurs essentiels limitent l'exploitation d'un testeur :

- ✓ *La profondeur mémoire* : Le volume de données à transmettre au circuit sous test peut être très important devant la quantité de données que peut stocker le testeur. Si la quantité de données requise est trop importante, le transfert du flot total doit s'effectuer en plusieurs étapes par rechargements successifs de la mémoire du testeur. Ce chargement/déchargement de données de la mémoire augmente le temps d'utilisation du testeur.
- ✓ *Le nombre de canaux* : Les circuits à tester ont un nombre d'entrées/sorties qui peut être supérieur au nombre d'entrées/sorties du testeur. Le coût d'une broche additionnelle sur le testeur (canal testeur) étant très élevé, il devient alors nécessaire de réduire le nombre de canaux du testeur.

Compte tenu de ces facteurs contraignant l'exploitation d'un testeur, toute technique permettant de réduire l'espace mémoire et/ou le nombre de canaux du testeur sans pour autant dégrader la qualité du test sera utile à la réduction du coût de fabrication des circuits intégrés.

La compression de données de test est une solution particulièrement intéressante qui répond à ce problème de réduction du coût de test des circuits intégrés. Dans cette solution, une séquence de test T_D est compressée en une séquence T_E plus petite en terme de volume de données. Cette séquence compressée est ensuite stockée dans la mémoire de l'ATE (Automatic Test Equipment). Un décodeur implanté à l'intérieur du circuit sous test permet lors du test de régénérer la séquence originale T_D à partir de la séquence compressée T_E .

La Figure IV.1 présente l'architecture générique utilisée lorsque les données de test mémorisées au sein de la mémoire du testeur, sont soit mémorisées sans compression préalable et donc directement appliquées au circuit sous test, soit mémorisées sous forme compressée et demandent donc à être décompressées à travers un décodeur avant d'être appliquées au circuit sous test.

Dans la Figure IV.1.a, nous retrouvons l'interface classique ATE-circuit sous test. Le nombre de canaux du testeur M est le même que celui du nombre d'entrées du circuit N . Dans ce cas, si le nombre d'entrées du circuit est élevé, il faut un nombre tout aussi important de

canaux du testeur pour les contrôler.

Dans la Figure IV.1.b, un bloc décodeur a été inséré dans le circuit sous test pour permettre la décompression de données de test (vecteurs de test) codées sur des mots de M bits vers des mots de N bits, avec $M < N$. Le nombre de canaux du testeur nécessaire au test du circuit considéré se trouve, dans ce cas, minimisé.

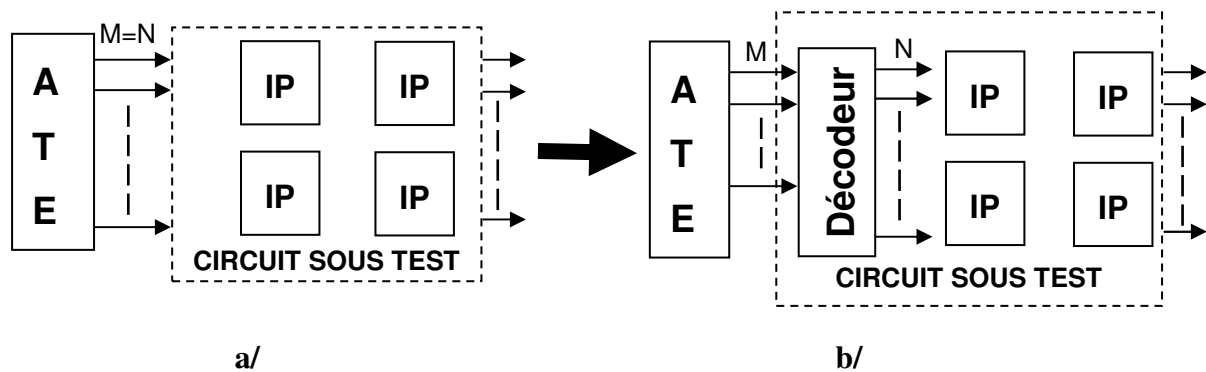


Figure IV.1 Architecture générique de compression de données de test

Deux voies sont possibles pour réduire le volume de données de test. La première consiste à réduire le nombre de vecteurs de test à transmettre, ce qui revient à réduire la profondeur mémoire utile du testeur. Pour réduire le nombre de vecteurs, il est possible d'utiliser des méthodes de *compression verticale* [CHAN01, ELMA01, DAST03].

La deuxième voie possible pour la compression de données de test est la réduction de la largeur de la séquence de test utilisée, ce qui permet par ailleurs l'utilisation d'un nombre de canaux du testeur réduit. Dans ce cas, il faut faire appel aux méthodes de *compression horizontale* [HAMZ99, RAJS04, WURT04].

La Figure IV.2 reprend le principe des deux méthodes de compression horizontale et verticale. A partir d'une séquence de test T_D composée de V vecteurs de M bits, une compression verticale consiste à réduire le nombre de vecteurs de V à V_C ($V_C < V$) tout en gardant le même nombre de bits M par vecteurs (Figure IV.2.a). La compression horizontale, quand à elle, consiste à réduire le nombre de bits par vecteur de M à M_C ($M_C < M$) tout en gardant le même nombre de vecteurs V (Figure IV.2.b). Notons par ailleurs que ces deux méthodes de compression peuvent être utilisées conjointement.

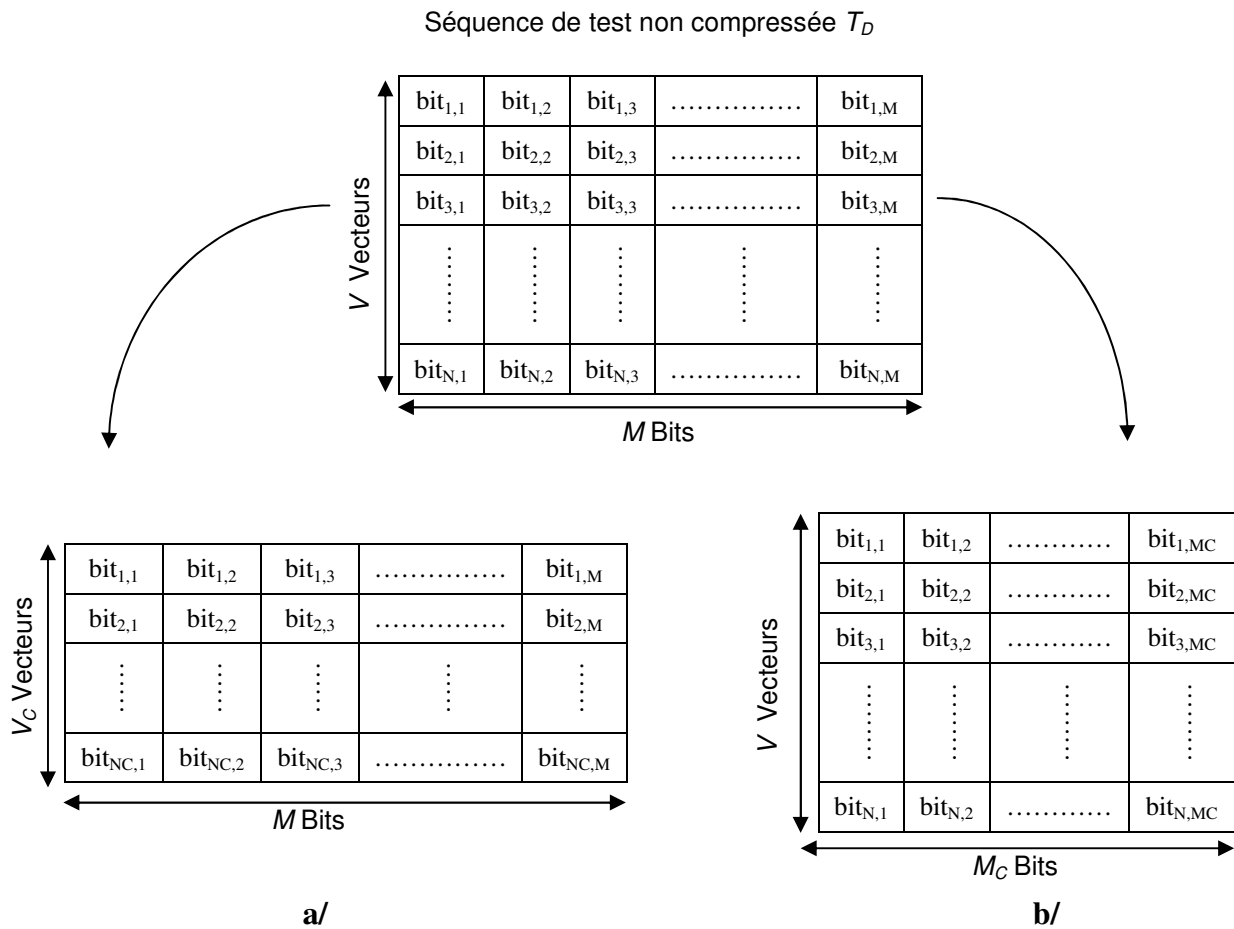


Figure IV.2 Compression horizontale et verticale

4.3 Réduction de puissance et compression de données de test

Plusieurs techniques ont été proposées pour réduire le volume de données de test. Ces techniques utilisent soit la compression horizontale ou la compression verticale. La majorité de ces techniques utilise l'assignation des bits non spécifiés des séquences de test afin d'atteindre des taux de compression acceptables. Cependant, il apparaît un conflit d'objectifs entre la réduction du volume de données et la réduction de la consommation de puissance pendant le test. En effet, d'une part, la séquence de test compressée génère une forte activité de commutation dans la chaîne de scan, ce qui augmentera la consommation de puissance dans le circuit sous test [SANK00, BAIK05]. D'autre part, les techniques proposées pour la génération automatique de séquences de test limitant la consommation de puissance tendent à augmenter la longueur des séquences ainsi générées [WANG97, ALYA03]. D'autres études ont proposé des solutions pour combiner la compression de données et la réduction de la

consommation de puissance lors du test. Dans cette partie nous présentons quelques unes d'entre elles.

4.3.1 Codage de Golomb et réordonnancement des vecteurs de test [CHAN01]

Dans cette technique, les auteurs proposent d'assigner tous les bits non spécifiés d'une séquence de test à '0' et de réordonner les vecteurs de test résultants afin de limiter la consommation de puissance. Ensuite, la séquence modifiée est compressée à l'aide d'un codage de Golomb.

La réduction de l'activité de commutation dans la chaîne de scan peut être obtenue grâce à un réordonnancement des vecteurs composant la séquence de test. Cette technique propose d'assigner tous les bits non spécifiés à '0' pour ensuite réordonner les vecteurs de test résultant en fonction du nombre de '0' qu'ils contiennent. Le premier vecteur ajouté à la séquence de test est celui qui contient le plus de '0' par rapport aux autres vecteurs. Le deuxième est celui qui contient le plus de '0' par rapport aux vecteurs restants et ainsi de suite. De cette façon, le nombre de transitions entre deux vecteurs de test successifs sera minimisé, limitant ainsi l'activité de commutation dans la chaîne de scan. La Figure IV.3 présente un exemple de cette procédure de réordonnancement où T_D représente la séquence de test et t_i chacun de ses vecteurs.

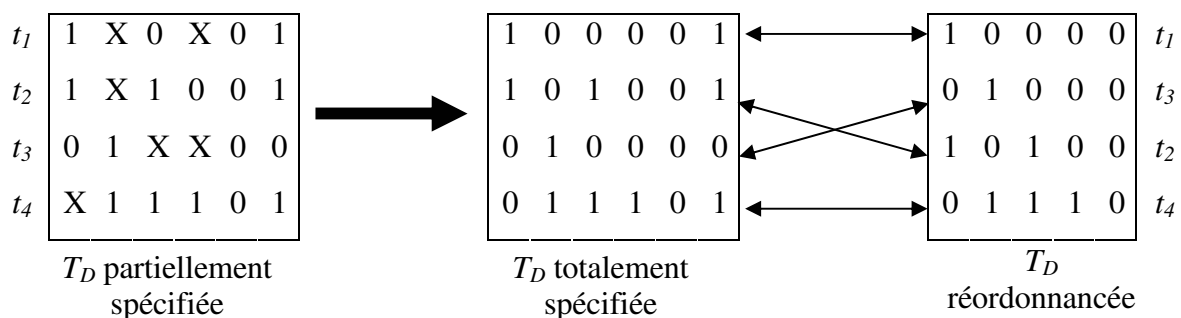


Figure IV.3 Exemple de réordonnancement des vecteurs de test

La compression de données, quand à elle, est réalisée à l'aide d'un codage de Golomb. Le codage de Golomb [GOLO66] se définit par deux parties distinctes : un préfixe et une queue. Le principe est de coder les suites de '0' de longueur l par un code (suite) plus petit. Le taux de compression dépend alors du paramètre k qui détermine le nombre de codes possédant la même longueur. La valeur optimale de k peut être obtenue en étudiant le nombre de '0' dans la séquence de test ou par expérimentation.

Cette technique ne considère pas la compression des réponses du circuit sous test aux vecteurs de la séquence de test appliquée. Cependant, comparée à une séquence standard générée par un outil ATPG (les bits non spécifiés sont assignés de manière aléatoire), cette technique permet de réduire le volume de données de test d'un facteur 3 (principalement par le codage de Golomb) avec une moyenne de 76% de réduction de la consommation de puissance moyenne (obtenu par réordonnancement).

4.3.2 Illinois Scan Architecture [HAMZ99]

La technique Illinois Scan Architecture (ISA) proposée dans [HAMZ99] permet de réduire le volume de données et le temps de test. Cette technique consiste à diviser chaque chaîne de scan en plusieurs segments. Le nombre important de bits non spécifiés dans une séquence de test permet d'avoir plusieurs segments scan avec des valeurs compatibles (pas de différence de bit à la même position) pour une majorité de vecteurs de la séquence de test. Dans ce cas, tous les segments compatibles d'une chaîne de scan sont configurés de façon à recevoir le même vecteur de test, on alimente alors toutes les chaînes de scan par une seule entrée (on peut aussi généraliser à plusieurs entrées disponibles). Cela permet d'obtenir un facteur de réduction du volume de données de test équivalent au nombre de segments compatibles. La Figure IV.4 présente le schéma de base de l'ISA. Notons au passage que la technique ISA est utilisée industriellement par la société Cadence.

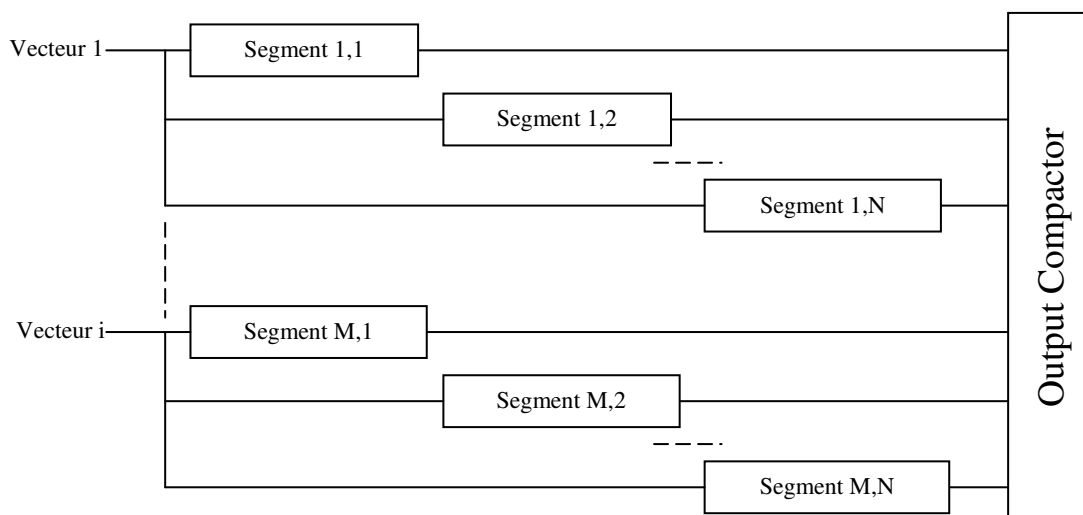


Figure IV.4 Illinois Scan Architecture

L'inconvénient de cette technique est qu'elle ne prend pas en compte le cas où seulement quelques segments d'une même chaîne de scan sont compatibles. Dans ce cas, le vecteur de test sera chargé en série en reconfigurant les différents segments en une seule chaîne de scan.

Plusieurs études ont été proposées pour améliorer l'architecture ISA, en particulier, en ce qui concerne la consommation de puissance. Dans [SINA02], les auteurs proposent une modification de l'architecture ISA afin de combiner la compression de données et la réduction de la consommation de puissance pendant le test. Cette modification consiste à inhiber les segments scan qui ne contiennent pas de bits spécifiés. De cette façon, seulement les segments contenant un ou plusieurs bits spécifiés seront chargés limitant ainsi l'activité de commutation dans la chaîne de scan et le volume de données de test. La technique présentée dans [ALYA05] propose de rendre compatibles toutes les combinaisons de segments d'une même chaîne de scan en ajoutant un décodeur pour adresser les segments compatibles. De cette façon, lors du chargement d'un groupe de segments compatibles, les autres segments ne sont pas chargés, ce qui permet de réduire le volume de données et l'activité de commutation dans la chaîne de scan. Les résultats expérimentaux obtenus grâce à ces améliorations de l'architecture ISA annoncent un facteur de compression de 5x du volume de données avec plus de 80% de réduction de la puissance consommée pendant le test.

4.3.3 Random Access Scan [ANDO80]

Cette architecture propose de réduire le volume de données de test en offrant un accès indépendant à chaque cellule scan afin d'actualiser (charger) seulement une partie d'entre elles. Elle propose aussi de réduire l'activité de commutation dans la chaîne de scan par l'inhibition des cellules non chargées.

Dans l'architecture RAS, un décodeur intégré au circuit permet d'adresser chaque cellule scan. A chaque instant, une seule cellule scan est accédée (chargée) alors que les autres cellules restent stables au même état logique. De cette façon, l'activité de commutation dans la chaîne de scan est réduite. Les cellules scan sont configurées en une grille de $n \times m$ cases. Le nombre de lignes et de colonnes est déterminé en fonction de la géométrie du circuit sous test ou le nombre de canaux testeur disponibles.

Plusieurs améliorations de l'architecture RAS ont été proposées. Dans [MUDL05], les auteurs proposent de générer une séquence de test par un ATPG classique, ensuite d'identifier les bits non spécifiés dans chaque vecteur de la séquence de test par des simulations de fautes

successives. Les bits non spécifiés dans chaque vecteur de test ne seront pas chargés dans les cellules scan, limitant ainsi le volume de données à charger et le temps de test. Dans [BAIK05], une technique de réordonnement des vecteurs de test est proposée afin de réduire l'activité de commutation dans la chaîne de scan lors de l'utilisation de l'architecture RAS. Cette technique se base sur l'analyse de la séquence de test pour déterminer le nombre de bits à '0' dans chaque vecteur de test. Ensuite, le réordonnement des vecteurs de test se fait selon le même principe que celui proposé par [CHAN01] (présenté à la Figure IV.3). Les résultats présentés dans [MUDL05] et [BAIK05] annoncent un taux de compression de l'ordre de 2,5x et une réduction de plus de 90% de l'activité de commutation dans la chaîne de scan. L'inconvénient de ces techniques est la surface du décodeur utilisé qui reste tout de même importante et peut dépasser 10% de la surface du circuit sous test.

4.3.4 Bilan et objectifs

Nous avons vu dans le paragraphe précédent trois méthodes qui ont été proposées pour réduire simultanément la consommation de puissance et le volume de données lors du test. D'autres solutions ont été proposées pour combiner la compression de données et la réduction de puissance pendant le test. Ces solutions sont basées sur la segmentation et l'inhibition de la chaîne de scan [SAMA03], sur l'analyse de la séquence de test et l'utilisation d'algorithmes de codage [NARA94, DABH98], ou sur la modification de la procédure de test et/ou du circuit sous test [SANK00]. Ces solutions permettent d'atteindre de 70% à 90% de réduction de la consommation de puissance dans le circuit sous test par rapport à une solution de test standard (sans la compression de données ni la modification du circuit sous test). Cependant, l'inconvénient principal de ces techniques est le faible taux de compression obtenu qui varie de 2x à 5x.

L'objectif de départ de l'étude présentée dans ce chapitre était de rechercher le meilleur compromis entre la compression de données et la réduction de la consommation de puissance durant le test. Pour cette raison, nous avons orienté nos recherches vers l'amélioration d'une technique de compression de données de test qui permette d'obtenir un taux de compression élevé tout en réduisant significativement la consommation de puissance dans le circuit lors du test. Nous avons choisi la technique de compression de données de test présentée dans [WANG05]. Cette technique permet d'obtenir des taux de compression de données allant jusqu'à 28x en se basant sur une analyse de la séquence de test et l'assignation des bits non spécifiés dans cette séquence. Nous avons modifié cette technique de façon à ce

que l'objectif ciblé soit, non seulement la compression de données, mais aussi la réduction de la consommation de puissance pendant le test. La partie suivante présente le principe général de cette technique de compression.

4.4 Solution proposée pour la compression de données et la réduction de puissance

Dans cette partie, nous allons tout d'abord présenter la technique de compression proposée dans [WANG05]. Ensuite, nous analysons la consommation de puissance après l'application des données décompressées par cette technique aux différentes chaînes de scan du circuit sous test. Enfin, nous présentons la modification de cette procédure de compression que nous avons proposé afin de combiner la réduction de puissance et la compression de données.

4.4.1 Compression de données par codage sélectif des "Scan Slices"

L'objectif de la technique présentée dans [WANG05] est de réduire le volume de données de test et le temps d'application pendant le test par scan. Cette technique cible le test des cœurs IP (Intellectual Property) car elle ne nécessite pas la connaissance de la structure interne du circuit sous test. De plus, elle utilise un décodeur générique qui peut être utilisé par les différents cœurs "IP" du circuit sous test.

A partir de la séquence de test générée par un outil ATPG sans assignation aléatoire des bits non spécifiés, chaque vecteur de cette séquence de test est reconstitué en plusieurs "Scan slices". Un scan slice est une série de données logiques appliquée aux différentes chaînes de scan interne du circuit. Le premier bit d'un scan slice est appliqué à la première chaîne de scan, le deuxième bit est appliqué à la deuxième chaîne de scan et ainsi de suite.

La Figure IV.5 présente un exemple de reconstitution d'un vecteur de test en scan slices générés pour le test d'un circuit contenant trois chaînes de scan, où chaque chaîne de scan est composée de quatre cellules.

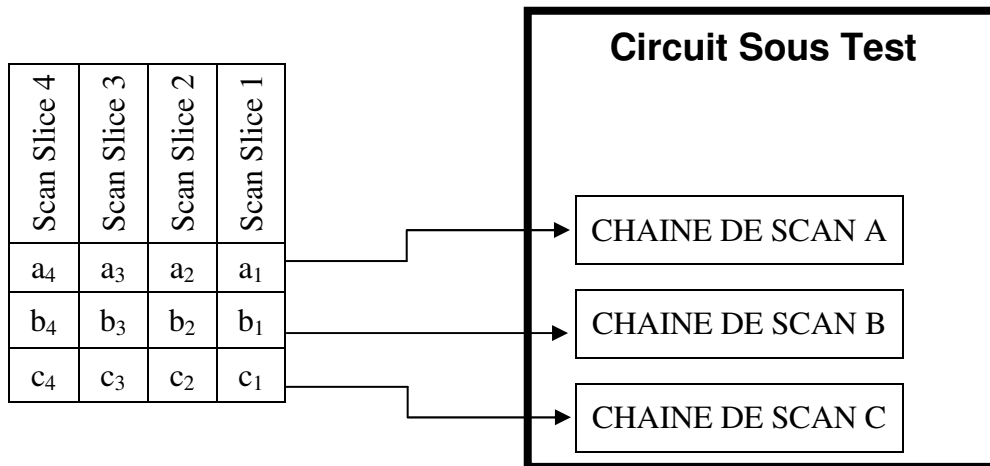
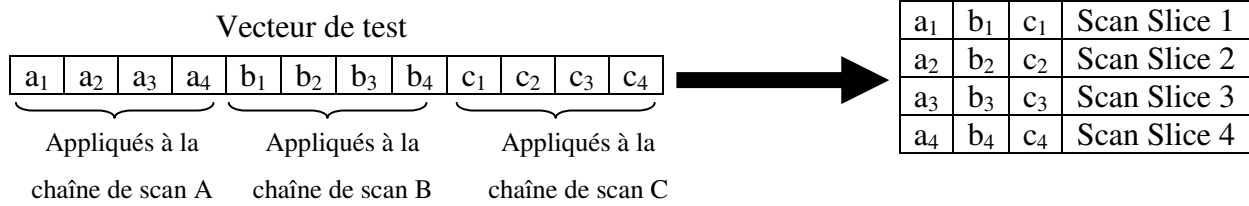


Figure IV.5 Exemple de reconstitution d'un vecteur de test en scan slices

La deuxième étape de la technique de compression est de coder chaque scan slice en une série de c bits appelée *code de slice* (ou slice-code) avec :

$$c = K + 2 \text{ et } K = \lceil \log_2(N + 1) \rceil$$

où N est le nombre de chaînes de scan dans le circuit sous test.

Les deux premiers bits d'un code-slice représentent le *code de contrôle* et les K bits suivants le *code de données* (Figure IV.6).

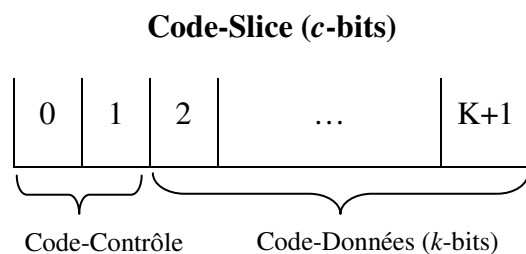


Figure IV.6 Représentation d'un Code-Slice

L'approche proposée code uniquement une partie des bits spécifiés dans un slice. Elle commence par la détermination du nombre de '0' et de '1' dans un slice donné. S'il y a plus

de '1' ('0') que de '0' ('1'), alors tous les bits à 'X' dans le slice seront affectés à '1' ('0') et seulement les '0' ('1') seront codés. Dans ce cas, les '0' ('1') sont appelés les *bits ciblés*. Les bits ciblés sont codés en codes-données à l'aide de deux modes différents : le mode un seul bit et le mode plusieurs bits.

Dans le *mode un seul bit*, chaque bit dans le slice est adressé de 0 à $N-1$ suivant sa position dans le slice. Un bit ciblé est représenté par un code-données qui prend la valeur de son adresse. Par exemple, pour coder le slice "XXX10000", les 'X' sont affectés à '0' et le seul bit à '1' à la position 3 est codé par "0011" qui fait référence à l'adresse 3. Cependant, comme il y a N chaînes de scan, un code-données qui a la valeur binaire de N signifie que le slice ne contient aucun bit ciblé (l'adressage commence à '0' et s'arrête à ' $N-1$ '). Dans le *mode un seul bit*, chaque slice est codé en un seul code-slice. Toutefois, s'il y a plusieurs bits ciblés dans le slice alors le mode plusieurs bits est utilisé afin d'optimiser le taux de compression obtenu.

Dans le *mode plusieurs bits*, un slice de N bits est divisé en $M = \lceil N / K \rceil$ groupes, où chaque groupe (sauf peut être le dernier) est constitué de K bits. Si un groupe contient plus de deux bits ciblés, alors le groupe considéré est copié en entier dans le code-données. Deux codes-données sont nécessaires pour coder un groupe. Le premier représente l'adresse du premier bit du groupe et le second contient les données du groupe. Considérons l'exemple où $N = 8$ et $K = 4$. Chaque slice est alors constitué de deux groupes de 4 bits. Pour coder le slice "X1110000", les trois premiers '1' du premier groupe "X111" sont codés. Le résultat du codage sera les deux codes-données "0000" et "X111" qui représentent, successivement, l'adresse du premier bit du premier groupe et les données de ce groupe. Le deuxième groupe "0000" n'est pas codé car il ne contient aucun bit ciblé (aucun bit à '1').

La Figure IV.7 présente le schéma général de cette technique de compression. Après le codage des slices en codes-slice, un décodeur intégré dans le circuit sous test permet de régénérer les slices originaux à partir des codes-slice. Un buffer inséré entre le décodeur et les différentes chaînes de scan permet de temporiser l'application des données de test aux chaînes de scan jusqu'au décodage complet de tous les codes-slice d'un slice donné.

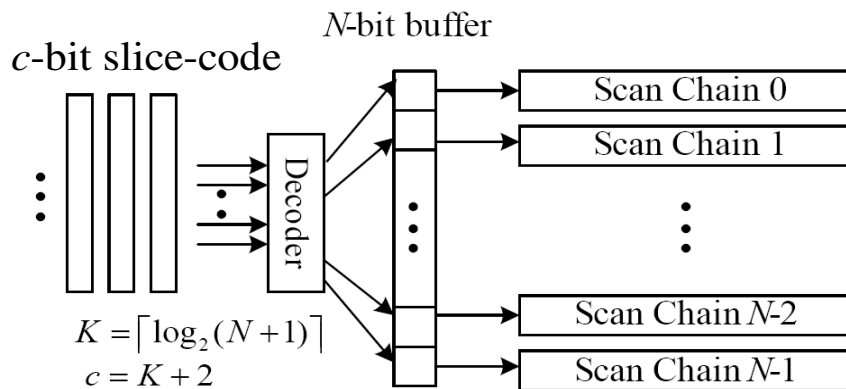


Figure IV.7 Schéma général de la technique de compression [WANG05]

Afin d'éviter une ambiguïté d'interprétation des codes-données lors du décodage des données compressées, les codes-contrôle "00", "01" et "10" sont utilisés pendant le mode un seul bit et le code-contrôle "11" pendant le mode plusieurs bits. Les codes-contrôle "00" et "01" indiquent le début du codage d'un nouveau slice.

Le Tableau IV.1 présente un exemple de codage par cette méthode. La première colonne indique les scan slices à coder. Les deuxième et troisième colonnes présentent le résultat du codage et la dernière colonne décrit la procédure de codage suivie. Par exemple, le premier slice "XX00 010X" contient un seul bit à '1'. Dans ce cas, le *mode un seul bit* est utilisé afin de coder ce slice. Le code de contrôle "00" indique que les bits à 'X' sont assignés à '0' et le code de données "0101" indique l'adresse 5 du bit ciblé (le bit à '1' dans le slice).

Scan Slices	Codage		Description
	Code de contrôle	Code de données	
XX00 010X	00	0101	Coder un slice, affecter les bits à 'X' à '0', mettre le bit 5 à '1'.
1110 0001	00	0111	Coder un nouveau slice, affecter les bits à 'X' à '0', mettre le bit 7 à '1'.
	11	0000	Commencer le mode plusieurs bits à partir de la position 0.
	11	1110	Les données sont "1110".
XXXX XX11	01	1000	Coder un nouveau slice, affecter les bits à 'X' à '1', aucun bit n'est mis à '0'

Tableau IV.1 Exemple de codage

Une propriété de cette technique de compression est que les slices compressés (codes-slice) peuvent être identiques. Dans ce cas, l'instruction "ATE pattern-repeat" du testeur peut être utilisée. Cette instruction permet de compacter V vecteurs de données compatibles en un

seul vecteur qui occupera une seule place dans la mémoire ATE et sera chargé V fois. Cela réduira le nombre total de données à sauvegarder dans l'ATE ainsi que le temps nécessaire à leur chargement.

Les auteurs dans [WANG05] ont appliqué cette technique à des circuits industriels contenant de 50K à 1,5 millions de portes logiques avec de 10K jusqu'à 110K cellules scan. Les séquences de test non compressées fournies par un outil ATPG contenaient seulement de 1% à 4% de bits spécifiés. Les résultats ont montré qu'un facteur de réduction du volume de données de test de l'ordre de 22x sans l'instruction "ATE pattern-repeat", et de 28x avec l'instruction "ATE pattern-repeat" peut être obtenu avec la présente technique. Le temps nécessaire à la génération des séquences compressées était de l'ordre de 2 minutes dans le cas des plus grands circuits. La taille du décodeur utilisé par cette technique pour régénérer la séquence originale à partir de la séquence compressée lors du test représente moins de 1% de la surface totale du circuit.

Cependant, cette technique ne considère pas l'aspect consommation de puissance pendant le test. En effet, durant l'affectation des bits non spécifiés des scan slices, la méthode de compression ne prend pas en compte le fait que les vecteurs de test résultants et qui seront appliqués aux chaînes de scan peuvent augmenter le nombre de cellules scan qui changent d'état à chaque front d'horloge. Cela peut favoriser une forte activité de commutation dans les chaînes de scan. Ce problème est détaillé dans la partie suivante.

4.4.2 Analyse de la consommation de puissance après la décompression des scan slices

Nous avons vu dans la partie précédente que la technique de compression présentée dans [WANG05] réduit d'une manière très efficace le volume de données de test. Par contre, elle ne prend pas en compte le problème de consommation de puissance lors du test. Dans cette partie nous expliquons le problème de consommation de puissance lors de la procédure de compression.

Pour évaluer l'activité de commutation que génère les différents scan slices dans les chaînes de scan, nous nous sommes appuyés sur la métrique proposée dans [SANK00] pour calculer le nombre de *transitions pondérées* (*Weighted Transitions WT*) dans les chaînes de scan après l'application de tous les scan slices. Cette métrique est détaillée dans l'Annexe A.

Pendant l'exécution de la procédure de compression présentée dans la partie précédente, l'affectation des bits à 'X' à des valeurs logiques '0' ou '1' peut générer une forte

activité de commutation dans les chaînes de scan du circuit sous test à chaque front d'horloge. Afin d'expliquer ce problème de forte activité de commutation, considérons l'exemple du Tableau IV.2 où $N = 8$ (nombre de chaînes de scan) et $K = 4$ (nombre de bits dans le code-données). Chaque slice est composé de 8 bits (deux groupes de 4 bits).

Scan Slices	Codage		Description
	Code de contrôle	Code de données	
XX00 010X	00	0101	Coder un slice, affecter les bits à 'X' à '0', mettre le bit 5 à '1'.
XXXX XX11	01	1000	Coder un nouveau slice, affecter les bits à 'X' à '1', aucun bit n'est mis à '0'.
X00X XXXX	00	1000	Coder un nouveau slice, affecter les bits à 'X' à '0', aucun bit n'est mis à '1'.
11XX 0XXX	01	0100	Coder un nouveau slice, affecter les bits à 'X' à '1', mettre le bit 4 à '0'.

} 4 codes-slice

Tableau IV.2 Exemple de codage

Les nouveaux slices obtenus après la décompression des quatre codes-slice du Tableau IV.2 et qui seront appliqués aux 8 chaînes de scan du circuit sont donnés dans le Tableau IV.3. La première colonne de ces scan slices sera appliquée à la première chaîne de scan, la deuxième colonne à la deuxième chaîne de scan et ainsi de suite. Les deux dernières lignes présentent le nombre de transitions pondérées (WT) dans chaque chaîne de scan et le nombre total de transitions pondérées (Total WT) générées aux entrées du circuit après le chargement de tous les slices. Le résultat de la compression est 4 codes-slice (Tableau IV.2) et le *WT Total* obtenu pour ces scan slices est égal à 44.

Scan Slices après décompression								Description
CS1	CS2	CS3	CS4	CS5	CS6	CS7	CS8	
0	0	0	0	0	1	0	0	Les 'X' ont été affectés à '0'
1	1	1	1	1	1	1	1	Les 'X' ont été affectés à '1'
0	0	0	0	0	0	0	0	Les 'X' ont été affectés à '0'
1	1	1	1	0	1	1	1	Les 'X' ont été affectés à '1'
6	6	6	6	5	3	6	6	WT
44								WT Total

Tableau IV.3 Résultat de la décompression

Maintenant, si tous les bits à 'X' de cette séquence de test sont affectés à '0' ('1') avant d'exécuter la procédure de compression (Tableaux IV.4 et IV.5), le nombre de codes-slice obtenu après la compression sera égal à 8 (9). Ce nombre est plus grand que celui obtenu avec la technique initiale (4 codes-slice), ce qui engendrera une diminution du taux de compression obtenu avec la technique initiale. Cependant, nous montrons dans la suite que l'affectation des bits à 'X' n'a qu'un faible impact sur le taux de compression dans le cas de circuits réels.

Scan Slices	Codes-slice	
0 0 0 0 0 1 0 0	00	0101
0 0 0 0 0 0 1 1	00	1000
	11	0100
	11	1100
0 0 0 0 0 0 0 0	00	1000
1 1 0 0 0 0 0 0	00	1000
	11	0000
	11	1100
15	WT Total	

} 8 codes-slice

Tableau IV.4 Tous les bits à 'X' sont affectés à '0'

Scan Slices	Codes-slice	
1 1 0 0 0 1 0 1	01	1000
	11	0000
	11	1100
	11	0101
1 1 1 1 1 1 1 1	01	1000
1 0 0 1 1 1 1 1	01	1000
	11	0000
	11	1001
1 1 1 1 0 1 1 1	01	0100
19	WT Total	

} 9 codes-slice

Tableau IV.5 Tous les bits à 'X' sont affectés à '1'

En ce qui concerne la consommation de puissance, nous remarquons à partir du Tableau IV.4 (Tableau IV.5) que le *WT Total* obtenu lorsque tous les bits à 'X' sont affectés à '0' ('1') est égal à 15 (19). Ce *WT Total* est largement inférieur à celui obtenu sans la modification de la séquence de test qui est égal à 44. Cela se traduira par une forte réduction de l'activité de commutation dans les chaînes de scan.

A partir de cette observation, nous avons proposé de modifier la technique de compression présentée dans [WANG05] afin de combiner la compression de données et la réduction de la consommation de puissance lors du test. La modification proposée est présentée dans la partie suivante.

4.4.3 Modification proposée

La modification apportée à la technique de compression initiale consiste à affecter les bits non spécifiés d'une séquence de test à des valeurs logiques avant d'exécuter la procédure de compression. Cette assignation se fait de manière à réduire l'activité de commutation dans les chaînes de scan lors du test [BADE06b].

La modification proposée consiste à affecter tous les bits non spécifiés de la séquence de test par les trois techniques d'assignation 0, 1 et MT-Filling présentées dans le chapitre précédent. La technique 0-Filling assigne tous les bits non spécifiés de la séquence de test à '0'. La technique 1-Filling assigne tous les bits non spécifiés de la séquence de test à '1'. La technique MT-Filling assigne chaque bit non spécifié à la valeur du dernier bit spécifié rencontré en fonctionnant de haut en bas de chaque colonne des scan slices. Le Tableau IV.6 présente un exemple d'application de la technique MT-Filling. Ce tableau montre les slices de départ sans aucune affectation des bits à 'X' et les slices modifiés par l'application de la technique d'assignation MT-Filling.

Scan Slices de départ								Scan Slices après l'application de MT-Filling							
CS1	CS2	CS3	CS4	CS5	CS6	CS7	CS8	CS1	CS2	CS3	CS4	CS5	CS6	CS7	CS8
0	1	1	0	1	0	0	1	0	1	1	0	1	0	0	1
X	X	X	1	X	0	1	X	0	1	1	1	1	0	1	1
X	1	1	X	X	X	X	0	0	1	1	1	1	0	1	0
X	X	X	X	X	X	1	X	0	1	1	1	1	0	1	0

Tableau IV.6 Exemple de modification des scan slices avec la technique MT-Filling

Après l'application d'une de ces trois techniques d'assignation, la procédure de compression est exécutée afin d'évaluer le nouveau taux de compression de données et la réduction de *WT Total* pendant le test. Nous avons appliqué la technique modifiée à plusieurs séquences générées pour le test de différents circuits académiques et industriels. Les résultats obtenus sont présentés dans la partie suivante.

4.4.4 Résultats expérimentaux

Dans une première étude, nous avons appliqué la technique de compression initiale *TCI* et la technique de compression modifiée *TCM* à différentes séquences générées pour le test des circuits appartenant aux benchmarks ISCAS'89 et ITC'99. Les caractéristiques des circuits étudiés, les caractéristiques de la station de travail et les outils utilisés sont identiques à ceux présentés dans la partie 1.5.1. Les caractéristiques des séquences de test sont identiques à celles présentées dans la partie 3.3.2. Nous avons calculé le taux de compression obtenu avec les deux techniques de compression *TCI* et *TCM*. Ensuite, nous avons évalué la réduction de consommation de puissance pendant le test. Dans une deuxième étude, nous avons évalué le taux de compression ainsi que la réduction de puissance obtenus avec les deux techniques *TCI* et *TCM* pour plusieurs circuits industriels.

4.4.4.1 Réduction du volume de données de test

Dans le Tableau IV.7, pour chaque circuit nous présentons les taux de compression de données obtenus avec les deux techniques *TCI* et *TCM*. La première colonne indique le nom du circuit. La deuxième colonne présente le nombre de chaînes de scan dans le circuit et le nombre de canaux ATE nécessaires (canaux du testeur). Nous avons considéré différents nombres de chaînes de scan afin de montrer son influence sur le taux de compression obtenu. Les colonnes de 3 à 12 présentent le taux de compression $\gamma_v = T_D / T_E$ obtenu avec *TCI* et *TCM* en utilisant les différentes techniques d'assignation de bits non spécifiés. Ces résultats sont obtenus avec et sans l'utilisation de l'instruction ATE pattern-repeat du testeur.

Ces résultats montrent que γ_v obtenu avec *TCI* (en moyenne 3,29x et 2,57x avec et sans ATE pattern-repeat) est souvent plus important que celui obtenu avec *TCM*. Ce résultat était prévisible car, d'une part, tous les bits non spécifiés sont affectés à des valeurs logiques avant même l'exécution de la procédure de compression. D'autre part, cette assignation se fait d'une manière à optimiser la consommation de puissance (assignation colonne par colonne dans les scan slices) sans considération de la compression de données (assignation ligne par ligne dans les scan slices). De ce fait, le résultat de cette assignation peut conduire à des scan slices difficiles à compresser. L'importance de ces résultats c'est qu'ils montrent aussi que même si γ_v obtenu avec *TCI* est plus grand, γ_v obtenu avec *TCM* reste dans le même ordre de grandeur que celui obtenu avec *TCI* (2,95x, 3,12x et 2,58x avec ATE pattern-repeat après l'application des techniques 0, 1 et MT-Filling respectivement).

Circuit	# Chaînes de scan / # Canaux ATE	γ_v sans ATE pattern-repeat				γ_v avec ATE pattern-repeat			
		TCI	0-Filling	1-Filling	MT-Filling	TCI	0-Filling	1-Filling	MT-Filling
b04s	7/5	1,28	0,98	1,05	0,9	1,69	1,25	1,35	1,1
	15/6	1,57	1,11	1,26	1,18	1,67	1,18	1,33	1,26
	31/7	1,88	1,37	1,66	1,62	1,93	1,52	1,73	1,7
b12	7/5	1,32	1,11	1,05	0,78	2,11	1,6	1,52	0,99
	15/6	1,76	1,35	1,25	1	1,97	1,46	1,36	1,08
	31/7	1,49	1,25	1,23	1,13	1,56	1,32	1,32	1,2
b14s	15/6	1,86	1,37	1,09	0,91	2,34	1,62	1,21	0,98
	31/7	1,79	1,38	1,27	1,07	1,93	1,58	1,4	1,17
	63/8	1,74	1,47	1,49	1,2	2,09	1,91	1,92	1,62
b17s	127/9	3,25	2,06	3,01	2,17	3,87	3,05	3,87	2,74
	255/10	2,46	1,82	2,44	2,18	3,52	3,29	4,07	3,45
	511/11	2,43	1,98	2,57	2,33	4,37	4,57	5,15	4,81
s5378	15/6	1,79	1,25	1,5	1,02	2,29	1,48	1,86	1,15
	31/7	1,92	1,26	1,69	1,31	2,05	1,34	1,83	1,39
	63/8	1,75	1,24	1,65	1,33	1,86	1,48	1,87	1,55
s13207	63/8	4,08	3,03	3,25	2,47	4,98	4,04	3,92	2,91
	127/9	4,18	2,93	3,25	2,72	5,07	4,42	4,19	3,59
	255/10	3,35	2,49	2,72	2,46	5,26	4,96	5,13	4,74
s38417	127/9	4,12	3,8	4,03	3,41	5,5	5,71	5,93	4,43
	255/10	4,69	4,16	4,64	4,19	5,9	6,39	6,61	5,39
	511/11	5,34	4,94	5,47	5,18	7,03	7,73	7,86	6,9
Moyenne		2,57	2,02	2,27	1,93	3,29	2,95	3,12	2,58

Tableau IV.7 Résultats de compression

4.4.4.2 Réduction de consommation de puissance

Nous avons calculé la réduction du nombre de commutations WT pendant le chargement et le déchargement des données de test dans les chaînes de scan. Le Tableau IV.8 présente les résultats obtenus. Nous avons réalisé deux études. La première consistait à comparer WT obtenu avec la séquence générée par TCM à celui obtenu avec la séquence générée par TCI (colonnes 3 à 5). La deuxième étude consistait à comparer WT obtenu avec TCI et TCM à celui obtenu avec la séquence standard (affectation aléatoire des bits à 'X') générée par l'outil TetraMax (colonnes 6 à 9).

Circuit	# Chaînes de scan / # Canaux ATE	Réduction de WT comparée à TCI (%)			Réduction de WT comparée à une séquence standard (%)			
		TCM			TCI	TCM		
		0-Filling	1-Filling	MT-Filling		0-Filling	1-Filling	MT-Filling
b04s	7/5	11,51	5,17	9,62	55,00	60,18	57,33	59,33
	15/6	9,36	1,05	3,37	50,48	55,11	51,00	52,15
b12	7/5	1,41	0,01	2,21	67,23	67,70	67,24	67,96
	15/6	0,90	0,00	6,52	64,85	65,17	64,85	67,15
b14s	15/6	10,00	1,03	2,82	75,80	78,22	76,05	76,49
	31/7	13,44	1,99	8,85	68,81	73,00	69,43	71,57
b17s	31/7	-17,94	5,52	12,14	90,74	89,08	91,25	91,86
	63/8	-15,91	5,20	10,04	90,77	89,30	91,25	91,70
	127/9	-12,47	8,85	12,07	89,77	88,49	90,67	91,00
s5378	7/5	6,02	27,75	26,35	52,82	55,66	65,91	65,25
	15/6	9,77	26,75	22,78	47,15	52,31	61,29	59,19
	31/7	9,72	17,78	-15,73	52,09	56,74	60,61	44,55
s13207	31/7	9,36	-2,19	-46,90	80,24	82,09	79,81	70,97
	63/8	18,97	-4,45	-4,77	76,73	81,14	75,69	75,62
	127/9	26,32	0,74	2,49	74,67	81,33	74,85	75,30
s38417	63/8	27,38	7,98	-4,98	69,97	78,19	72,37	68,48
	127/9	37,19	8,53	-9,06	72,23	82,55	74,60	69,71
	255/10	41,66	8,53	-5,97	70,68	82,90	73,18	68,93

Tableau IV.8 Réduction de consommation de puissance

Ces résultats montrent l'efficacité de la modification apportée à *TCI* puisque la réduction obtenue sur *WT* peut atteindre jusqu'à 41,66% avec 0-Filling (27,75% et 26,35 pour 1 et MT-Filling respectivement) par rapport à la solution *TCI* (sans considération explicite de la consommation de puissance). Pour la majorité des circuits, la technique 0-Filling apporte le meilleur résultat par rapport à 1-Filling ou MT-Filling en terme de réduction de puissance. D'autre part, la faible réduction obtenue pour quelques circuits peut être expliquée par le pourcentage élevé de réduction de *WT* obtenu avec la séquence générée par *TCI* comparée à *WT* obtenu avec la séquence standard.

En d'autres termes, quand *TCI* atteint une bonne réduction de consommation de puissance, elle ne laisse pas la place à une réduction supplémentaire avec *TCM*. Considérons le circuit b17s, la meilleure réduction de *WT* obtenue avec *TCM* est de l'ordre de 12% avec la technique d'assignation MT-Filling comparée à *TCI*. Ce faible résultat est dû à la forte réduction obtenue avec *TCI* (de l'ordre de 90%) comparée à la solution standard. D'autres raisons peuvent expliquer cette faible réduction, comme l'influence des commutations présentes dans les chaînes de scan lors du déchargement des réponses du circuit sur *WT Total*.

En effet, les résultats présentés dans le Tableau IV.8 prennent en compte l'activité de commutation due au chargement des vecteurs de test et au déchargement des réponses du circuit dans les chaînes de scan. La *TCM* modifie la séquence de test initiale en remplaçant les bits à 'X' par des valeurs logiques dans le but de réduire *WT* pendant le chargement des vecteurs de test. Cependant, les réponses du circuit à ces vecteurs de test complètement spécifiés sont uniques et dépendent de la fonction du circuit (nous ne pouvons pas contrôler le résultat de la réponse du circuit avec *TCM* afin de réduire *WT* pendant le déchargement de ces réponses).

4.4.4.3 Résultats sur circuits industriels

Nous avons évalué le taux de compression de données ainsi que la réduction de *WT* pour différents circuits industriels dans le cadre d'une collaboration avec le Professeur Krishnendu CHAKRABARTY de l'Université de Duke aux Etats-Unis.

Le Tableau IV.9 présente les caractéristiques principales des circuits expérimentés et leurs séquences de test. Pour le circuit 3, les séquences de test fournies sont différentes. La différence est dans le nombre maximum de bits spécifiés autorisés par vecteur de test. Ce nombre a été fixé à l'aide d'une option dans l'outil ATPG. Par exemple, pour le circuit 3_2000, chaque vecteur de test ne peut contenir plus de 2000 bits spécifiés.

Circuit	# Portes	# Cellules scan	Couverture de fautes (%)	Bits spécifiés (%)
Circuit 1	302714	43414	99,42	1,58
Circuit 2	404860	26970	98,85	1,31
Circuit 3_2000	121470	9628	98,66	1,96
Circuit 3_1000			98,51	2,04
Circuit 3_500			98,45	1,83
Circuit 3_200			98,27	1,32
Circuit 4	1,21 M	22205	99,16	1,76
Circuit 5	1,41 M	108863	95,07	0,92

Tableau IV.9 Caractéristiques des circuits expérimentés

Le tableau IV.10 regroupe les résultats en termes des taux de compression obtenus avec *TCI* et *TCM*. Ces résultats montrent que γ_v obtenu avec *TCI* pour la compression des séquences générées pour le test des circuits industriels (plus de 20x avec ou sans l'instruction ATE pattern-repeat) est plus grand que ce celui obtenu pour la compression des séquences générées pour le test des circuits académiques ISCAS'89 et ITC'99 (7,86x au maximum). Ce

résultat est directement lié au nombre de bits spécifiés dans les séquences de test (1-4% pour les circuits industriels et 5-35% pour les circuits académiques). Plus le nombre de bits spécifiés dans la séquence de test est petit, meilleur est le taux de compression. Le pourcentage élevé de bits spécifiés dans les séquences générées pour le test des circuits ISCAS'89 et ITC'99 est dû principalement à la grande profondeur logique de ces circuits, tandis que les circuits industriels ont une faible profondeur logique [WANG05].

Circuit	# Chaînes de scan / # Canaux ATE	γ_v sans ATE pattern-repeat				γ_v avec ATE pattern-repeat			
		TCI	0-Filling	1-Filling	MT-Filling	TCI	0-Filling	1-Filling	MT-Filling
Circuit 1	255/10	13,08	11,29	11,42	2,05	20,32	14,79	14,90	2,08
	511/11	14,71	12,74	12,81	2,67	19,56	14,30	14,34	2,70
	1023/12	15,02	13,26	13,26	3,39	18,92	13,83	13,82	3,42
Circuit 2	255/10	16,27	13,02	12,84	1,22	21,73	15,77	15,53	1,23
	511/11	17,86	14,02	13,93	1,66	20,44	14,84	14,79	1,68
	1023/12	16,74	13,71	13,74	2,36	18,41	13,93	13,97	2,38
Circuit 3_200	255/10	17,28	12,77	12,65	1,66	21,41	14,62	14,51	1,68
	511/11	17,34	13,20	13,11	2,36	18,93	13,75	13,70	2,38
	1023/12	15,89	12,85	12,75	3,40	16,95	13,02	12,92	3,43
Circuit 3_500	255/10	11,69	9,30	9,35	1,89	14,48	10,63	10,74	1,92
	511/11	11,60	9,50	9,60	2,37	12,98	9,90	10,05	2,39
	1023/12	10,93	9,31	9,39	3,03	12,20	9,44	9,52	3,06
Circuit 3_1000	255/10	10,10	8,36	8,48	2,30	13,00	9,62	9,83	2,33
	511/11	10,22	8,62	8,79	2,95	12,15	9,02	9,26	2,97
	1023/12	9,87	8,58	8,73	3,64	11,94	8,72	8,87	3,67
Circuit 3_2000	255/10	10,22	8,69	8,86	2,74	14,81	10,35	10,69	2,80
	511/11	10,75	9,24	9,49	3,63	14,56	9,85	10,22	3,67
	1023/12	10,81	9,47	9,74	4,62	14,65	9,70	9,99	4,65
Circuit 4	255/10	13,70	10,22	10,08	1,03	15,49	11,04	10,86	1,04
	511/11	12,87	10,16	10,02	1,24	13,78	10,44	10,28	1,25
	1023/12	11,51	9,74	9,61	1,59	12,28	9,84	9,69	1,60
Circuit 5	511/11	22,96	17,77	17,20	1,11	26,43	19,62	18,88	1,12
	1023/12	21,68	17,69	17,02	1,27	22,84	18,25	17,53	1,28
Moyenne		14,05	11,46	11,43	2,36	16,88	12,40	12,39	2,38

Tableau IV.10 Résultats de compression

Nous remarquons aussi que le γ_v obtenu avec *TCM* dans le cas de l'application des deux techniques d'assignation 0 et 1-Filling reste élevé comme cela est montré sur la dernière ligne du Tableau IV.10 (plus de 11x avec ou sans ATE pattern-repeat). Cependant, γ_v obtenu avec *TCI* est fortement dégradé lors de l'application de la technique d'assignation MT-Filling (2,37x en moyenne avec ou sans l'option ATE pattern-repeat). Ceci est dû à la nature de la

technique MT-Filling. Comme les bits non spécifiés sont assignés de manière verticale (colonne par colonne) dans chaque chaîne de scan avec *TCM*, les scan slices résultants (lignes) peuvent présenter une suite de valeurs logiques difficiles à compresser dans des codes de slice, ce qui se traduit par un faible γ_v . Ce problème n'apparaît pas lors de l'application des techniques d'assignation 0 ou 1-Filling car, dans ce cas, tous les bits non spécifiés sont assignés à '0' ou à '1'. Puisque le nombre de bits spécifiés dans les séquences de test est très petit (1-4%), les scan slices résultants présentent un grand nombre de '0' ou de '1' qui sont faciles à compresser.

Circuit	# Chaînes de scan / # Canaux ATE	Réduction de WT comparée à TCI (%) TCM		
		0-Filling	1-Filling	MT-Filling
Circuit 1	255/10	96,77	96,70	97,90
	511/11	97,24	97,25	98,28
	1023/12	97,57	97,56	98,54
Circuit 2	255/10	97,74	97,74	98,77
	511/11	98,08	98,03	99,02
	1023/12	98,12	98,06	99,08
Circuit 3_200	255/10	97,51	97,45	99,06
	511/11	97,71	97,61	99,21
	1023/12	97,70	97,62	99,27
Circuit 3_500	255/10	96,33	96,31	98,48
	511/11	96,75	96,68	98,78
	1023/12	96,79	96,73	98,90
Circuit 3_1000	255/10	95,79	95,80	98,13
	511/11	96,34	96,30	98,52
	1023/12	96,47	96,45	98,71
Circuit 3_2000	255/10	95,53	95,60	97,85
	511/11	96,34	96,35	98,39
	1023/12	96,62	96,64	98,64
Circuit 4	255/10	92,17	91,93	96,60
	511/11	96,72	96,74	98,76
	1023/12	96,83	96,84	98,96
Circuit 5	511/11	1,14	-1,04	55,69
	1023/12	75,60	75,03	89,79

Tableau IV.11 Réduction de consommation de puissance

Le Tableau IV.11 présente les résultats de comparaison de *WT* obtenu avec la séquence générée par *TCM* à celui obtenu avec la séquence générée par *TCI* (colonnes 3 à 5). Etant donné que la description des circuits expérimentés n'était pas disponible, nous n'avons pas considéré le déchargement des réponses. Pour cette raison, la réduction de *WT* présentée

dans le Tableau IV.11 a été évaluée uniquement pendant la phase de chargement des vecteurs de test.

En plus des taux de compression élevés obtenus, la *TCM* est très efficace en terme de réduction de consommation de puissance dans le circuit pendant le test par scan. En effet, les résultats obtenus pour tous les circuits expérimentés présentent une moyenne de 90% de réduction de l'activité de commutation. Pour expliquer ce résultat, il faut noter que la *TCI* assigne les bits non spécifiés scan slice par scan slice (ligne par ligne). Cette assignation garantit un bon taux de compression. Cependant, d'une part, cette assignation ne permet pas de réduire le nombre de transitions sur les vecteurs de test (colonnes dans les scan slices). D'autre part, les techniques d'assignation 0, 1 et MT-Filling affectent les bits non spécifiés de façon à limiter le nombre de transitions sur les vecteurs de test (ce qui se traduit par une faible activité de commutation).

4.5 Conclusion

Dans ce chapitre, nous avons présenté une technique de compression de données pour les circuits munis de chaînes de scan [WANG05]. Cette technique permet de réduire jusqu'à 26x le volume de données de test. Cette technique de compression présente de nombreux avantages. Elle ne nécessite pas la connaissance de la structure interne du circuit sous test. Elle utilise un décodeur générique indépendant de la structure du circuit et de la séquence de test. Ce décodeur peut être utilisé par les différents cœurs "IP" du circuit sous test et sa taille ne représente que 1% de la surface totale du circuit.

L'inconvénient de cette technique est qu'elle ne considère pas le problème de consommation de puissance dans le circuit lors de la procédure de compression. En effet, durant l'affectation des bits non spécifiés d'une séquence de test par la procédure de compression, les vecteurs de test résultants peuvent générer une forte activité de commutation dans les chaînes de scan du circuit. Pour cette raison, nous avons modifié cette technique afin de réduire, simultanément, le volume de données et la consommation de puissance lors du test. La modification apportée consiste à affecter les bits non spécifiés de la séquence de test à compresser à des valeurs logiques dans le but de réduire le nombre de transitions sur chaque vecteur de cette séquence de test.

Comparée à la technique de compression initiale, la technique proposée permet de réduire d'une manière très efficace (plus de 95%) la consommation de puissance pendant le

test par scan, tout en garantissant un taux élevé de compression de données de test (plus de 19x).

Conclusion Générale

Conclusion Générale

La conception en vue du test, et notamment le test par scan, peut entraîner des problèmes de consommation de puissance trop élevée lors du test. Les conséquences de cette surconsommation sont une perte de fiabilité, le risque de destruction du circuit, l'apparition d'informations logiques erronées, etc. Dans le cadre de cette thèse, nous avons proposé trois techniques permettant de réduire la puissance dissipée lors du test par scan, et plus particulièrement la réduction de la puissance de pic pendant le cycle de test.

La première technique proposée permet de réduire l'activité de commutation dans le circuit sous test par un ordonnancement des cellules scan. Cet ordonnancement est basé sur une analyse des données de test et nécessite la vérification de deux contraintes. D'une part, il faut réduire la puissance de pic pendant le cycle de test sur toute la séquence de test. D'autre part, cette réduction doit se faire tout en respectant une contrainte de limite sur la puissance de pic (qui doit être fixée par le concepteur du circuit). Les résultats obtenus sont intéressants, car ils montrent qu'une réduction significative de la puissance de pic (jusqu'à 50%) peut être obtenue pour tous les vecteurs composant la séquence de test, tout en respectant la limite sur la puissance de pic. Cependant, l'inconvénient de cette technique est, d'une part, la faible réduction de puissance de pic pendant les cycles de décalage, et d'autre part, les problèmes de routage résultant de l'ordonnancement des cellules scan.

La deuxième technique proposée permet de répondre à ces problèmes. Cette technique propose d'assigner les bits non spécifiés des séquences de test afin de réduire la puissance de pic pendant le cycle de test et pendant les cycles de décalage. Cette assignation est réalisée en se basant sur des informations structurelles du circuit sous test. Contrairement à la solution de réordonnancement des cellules de la chaîne de scan, cette technique apporte une réduction significative de la puissance de pic pendant le cycle de test (avec plus de 80%), en réduisant aussi la puissance de pic pendant les cycles de décalage. Par ailleurs, elle permet d'éliminer le

problème du routage des cellules de la chaîne de scan. Un des inconvénients de cette technique est la dégradation du taux de couverture de fautes. Toutefois, cette perte de couverture peut être limitée en établissant un compromis entre la réduction de puissance et la couverture de fautes. Un autre inconvénient de cette technique est qu'elle ne permet pas d'avoir un bon taux de compression de données de test.

La dernière technique proposée dans le cadre de cette thèse permet de réduire simultanément la consommation de puissance et le volume de données de test. Cette solution est basée sur une modification de la méthode de compression proposée dans [WANG05]. La modification apportée consiste à affecter de façon spécifique les bits non spécifiés de la séquence de test à compresser dans le but de réduire le nombre de transitions sur chaque vecteur de cette séquence de test. Comparée à la technique de compression initiale, la technique proposée permet de réduire de manière très efficace la consommation de puissance pendant le test par scan (jusqu'à 99%) tout en garantissant un taux élevé de compression de données de test (19x). Outre les résultats obtenus, cette technique présente de nombreux avantages. Elle ne nécessite pas la connaissance de la structure interne du circuit sous test et peut donc s'appliquer dans un contexte d'utilisation de cœurs IP. De plus, cette technique utilise un décodeur générique indépendant de la structure du circuit et de la séquence de test qui représente une très faible surface rajoutée.

Une perspective à ces travaux de thèse serait d'adapter toutes ces techniques au test des systèmes sur puce (Systems-On-Chip). En effet, dans les SOC, de nombreux éléments servant à l'acheminement des données de test, tels que les cellules de test dans les wrappers, les cellules scan à l'intérieur des cœurs, ou les éléments de connexion au niveau des mécanismes d'accès (Test Access Mechanism), nécessitent une prise en compte de leur activité lors du test. Cette adaptation devrait ouvrir la voie à d'autres études sur la réduction de consommation de puissance pendant le test des SOC. Une autre perspective est de cibler la réduction de consommation de puissance pendant le test pour un type ciblé de cœurs logiques ou de circuits intégrés, tel que les mémoires, les DSP, les convertisseurs...

Références Bibliographiques

Références Bibliographiques

- [ALTE02] J. Altet et A. Rubio, "Thermal testing of integrated circuits", Kluwer Academic Publishers, 2002.
- [ALYA03] A. A. Al-Yamani et E. J. McCluskey, "Built-in reseeding for serial BIST", IEEE VLSI Test Symposium, pp. 63-68, 2003.
- [ANDO80] H. Ando, "Testing VLSI with random access scan", IEEE Computer Conference, pp. 50-52, 1980.
- [BAIK05] D.H. Baik et K.K. Saluja, "Progressive Random Access Scan: A Simultaneous Solution to Test Power, Test Data Volume and Test Time", IEEE International Test Conference, pp. 15.2, 2005.
- [BONH01] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault et S. Pravossoudovitch, "A gated clock scheme for low power scan testing of logic Ics or embedded cores", IEEE Asian Test Symposium, pp. 253-258, 2001.
- [BONH02] Y. Bonhomme, P. Girard, C. Landrault et S. Pravossoudovitch, "Power Driven Chaining of Flip-flops in Scan Architectures", IEEE International Test Conference, pp. 796-803, 2002.
- [BONH03] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault et S. Pravossoudovitch, "Efficient scan chain design for power minimization during scan testing under routing constraint", IEEE International Test Conference, pp. 488-493, 2003.
- [BRGL89] F. Brglez and F. Fujiwara, "Combinational Profiles of Sequential Benchmark Circuits", IEEE International Symposium on circuits and systems, pp. 1929-1934, 1989.
- [BUSH00] M.L. Bushnell et V.D. Agrawal, "Essentials of Electronic Testing", Kluwer Academic Publishers, 2000.
- [BUTL04] K.M. Butler, J. Saxena, T. Fryars, G. Hetherington, A. Jain et J. Lewis, "Minimizing Power Consumption in Scan Testing: Pattern Generation and DFT Techniques", IEEE International Test Conference, pp. 355-364, 2004.
- [CADE00] "Silicon Ensemble", Cadence Design System Inc., 2000.
- [CAST96] T. Castell, C. Cayrol, M. Cayrol et D. Le Berre, "Using the davis and putnam procedure for an efficient computation of preferred models", European Conference on Artificial Intelligence, pp. 350-354, 1996.

- [CHAN01] A. Chandra et K. Chakrabarty, "Combining Low-Power Scan Testing and test Data Compression for System-on-a-Chip", Design Automation Conference, pp. 166-171, 2001.
- [CIRI87] M.A. Cirit, "Estimating dynamic power consumption of CMOS circuits", IEEE International Conference on VLSI Design, pp. 534-537, 1987.
- [CROU99] A. Crouch, "Design-For-test for digital IC's and embedded core systems", Prentice Hall, 1999.
- [CSHI04] C. Shi et R. Kapur, "How Power Aware Test Improves Reliability and Yield", IEEDesign.com, September 15, 2004.
- [DABH98] V. Dabholkar, S. Chakravarty, I. Pomeranz et S. Reddy, "techniques for minimizing power dissipation in scan and combinational circuits during test application", IEEE Transaction on Computer Design of Integrated Circuits and Systems, Vol. 17, pp. 1325-1333, décembre 1998.
- [DAST03] G. Dastida, M.E Ng et N.A Touba, "An efficient test vector compression scheme using selective Huffman coding", IEEE Transaction on Computer Design of Integrated Circuits and Systems", Vol. 22, N° 6, Juin 2003.
- [DAVI99] S. Davidson et panelists, "ITC99 Benchmarks circuits – preliminary results", IEEE International Test Conference, pp. 1125, 1999.
- [EICH91] E. B. Eichelberger, E. Lindbloom, J.A. Waicukauski, et T.W. Williams, "Structured Logic Testing", Englewood Cliffs, New Jersey: Prentice-Hall, 1991.
- [ELMA01] A. El-Maleh, S. Al Zahir et E. Khan, "Ageometric primitive based compression scheme for testing system-on-a-chip", IEEE VLSI Test Symposium, pp. 54-59, 2001.
- [FUJI85] H. Fujiwara, "FAN: A Fanout-Oriented Test Pattern Generation Algorithm", in Proceeding of the International Symposium on Circuits and Systems, pp.671-674, 1985.
- [GARE78] M R. Garey et D. S. Johnson, "Computers and Intractability, A Guide to the Theory of NP-Completeness", W.H. Freeman & Company, San Francisco, 1978.
- [GERS99] S. Gerstendorfer et H.J. Wunderlich, "Minimized Power Consumption for Scan-based BIST", IEEE International Test Conference, pp. 77-84, 1999.
- [GERS00] S. Gerstendorfer et H.J. Wunderlich, "Minimized Power Consumption for Scan-based BIST", Journal of Electronic Testing: Theory and Applications, Vol. 16, No. 3, pp. 203 - 212, 2000.
- [GIRA02] P. Girard, "Survey of Low-Power Testing of VLSI Circuits", IEEE Design & Test of Computers, Vol. 19, no. 3, pp. 82-92, 2002.
- [GOEL81] P. Goel, "An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits", IEEE Trans. on Computers, vol. C-30, no. 3, pp. 215-222, 1981.
- [GOLO66] S.W. Golomb, "Run-Length Encodings", IEEE Transaction on Information Theory, Volume 12, pp. 399-401, 1966.
- [HAMZ99] I. Hamzaoglu et J. Patel, "Reducing Test Application Time for Full Scan Embedded Cores", IEEE International Symposium on Fault Tolerant Computing, pp. 206-267, 1999.

- [HIRA03] T. Hiraide, K. O. Boateng, H. Konishi, K. Itaya, M. Emori, H. Yamanaka et T. Mochiyama, "BIST-Aided scan test – a new method for test cost reduction", IEEE VLSI Test Symposium, pp. 359-364, 2003.
- [HUAN99] T. C. Huang et K. J. Lee, "An Input Control Technique for Power Reduction in Scan Circuits During Test Application", IEEE Asian Test Symposium, pp. 315-320, 1999.
- [JOHN96] D.S. Johnson et L.A. McGeoch, "The Traveling Salesman Problem : A Case Study in Local Optimization", in Local Search algorithms in Combinatorial Optimization, E.H.L. Aarts and J.K. Lenstra, eds. John Wiley and Sons, 1996.
- [KLEI05] M. Klein, "Static Power and the Importance of Realistic Junction Temperature Analysis", Xilinx Inc., 2005.
- [MUDL05] A.S. Mudlapur, V.D. Agrawal et A.D. Singh, "A random access scan architecture to reduce hardware overhead", IEEE International Test Conference, pp. 15.1, 2005.
- [NARA94] S. Narayanan et M.A. Breuer, "Reconfigurable scan chains: A novel approach to reduce test application time", International Conference on Computer-Aided Design, pp. 624-630, 1994.
- [NGUY03] D. Nguyen, A. Davare, M. Orshansky, D. Chinnery, B. Thompson et K. Keutzer, "Minimization of Dynamic and Static Power Through Joint Assignment of Threshold Voltages and Sizing Optimization", International Symposium on Low Power Electronics and Design, pp.158-163, 2003.
- [POEH03] F. Poehl, M. Beck, R. Arnold, P. Muhmenthaler, N. Tamarapalli, M. Kassab, N. Mukherjee et J. Rajski "Industrial Experience with Adoption of EDT for Low-Cost Test Without Concessions", IEEE International Test Conference, pp. 1211-1220, 2003.
- [POIR04] R. Poirier, "Compression de données pour le test des circuits intégrés", Thèse de Doctorat, Université de Montpellier II, 2004.
- [RAJS04] J. Rajski, J. Tyszer, M. Kassab et N. Mukherjee, "Embedded deterministic test", IEEE Trans. on CAD, vol. 23, pp. 776-792, May 2004.
- [ROTH66] J. P. Roth, "Diagnosis of Automata Failures: A calculus and a Method", IBM Journal of Research and Development, vol. 10, no. 4, pp. 278-291, 1966.
- [SAMA03] S. Samaranayake, E. Gizdarski, N. Sitchinava, F. Neuveux, R. Kapur et T.W. Williams, "A reconfigurable shared scan-in architecture", IEEE VLSI Test Symposium, pp. 14-19, 2003.
- [SANK00] R. Sankaralingam, R. R. Oruganti et N. A. Touba, "Static Compaction Techniques to Control Scan Vector Power Dissipation", IEEE VLSI Test Symposium, pp35-40, 2000.
- [SANK02] R. Sankaralingam et N. A. Touba, "Controlling Peak Power during Scan Testing", IEEE VLSI Test Symposium, pp. 153-159, 2002.
- [SAXE03] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreepakash et M. Hachinger, "A Case Study of IR-Drop in Structured At-Speed Testing", IEEE International Test Conference, pp. 1098-1104, 2003.
- [SERV01] G. Serval, "Effets Parasites dus aux interconnexions", Thèse de Doctorat, Université de Montpellier II, 2001.

- [SHIN97] Yi-Shing Chang, Sandeep K. Gupta et Melvin A. Breuer, "Analysis of Ground Bounce in Deep Sub-Micron Circuits", IEEE VLSI Test Symposium, pp. 110-116, 1997.
- [SIAS05] SemiConductor Industry Association (SIA), "International Technology Roadmap for Semiconductors (ITRS)", 2005.
- [SINA02] O. Sinanoglu et A. Orailoglu, "A novel scan architecture for power-efficient, rapid test", International Conference on Computer-Aided Design, pp. 299-303, 2002.
- [SPEA96] W. M. Spears, "Simulated annealing for hard satisfiability problems", Discrete Mathematics and Theoretical Computer Science, pp. 533-558, 1996.
- [SYNO00] PowerMill®, Version 5.4, Synopsys Inc., 2000.
- [SYNO01] TetraMAX™, Version 2001.08, Synopsys Inc., 2001.
- [WANG05] Z. Wang et K. Chakrabarty, "Test Data Compression for IP Embedded Cores Using Selective Encoding of Scan Slices", IEEE International Test Conference, pp. 24.3, 2005.
- [WANG97] S. Wang et S. K. Gupta, "ATPG for heat dissipation minimization during scan testing", IEEE Design Automation Conference, pp. 614-619, 1997.
- [WANG99] S. Wang et S.K. Gupta, "LT-RTPG: A New Test-Per-Scan BIST TPG for Low Heat Dissipation", IEEE International Test Conference, pp. 85-94, 1999.
- [WHET00] L. Whetsel, "Adapting Scan Architectures for Low Power Operation", IEEE International Test Conference, pp. 863-872, 2000.
- [WILL73] M. J. Y. Williams et J. B. Angell, "Enhancing Testability of Large-Scale Integrated Circuits via Test Points and Additional Logic", IEEE Trans. On Computers, vol. C-22, no. 1, pp. 46-60, 1973.
- [WOHL03] P. Wohl, J. A. Waicukauski, S. Patel et M. B. Amin, "Efficient compression and application of deterministic patterns in a logic BIST architecture", IEEE Design Automation Conference, pp.566-569, 2003.
- [WUND85] H.J. Wunderlich, "PROTEST: A Tool for Probabilistic Testability Analysis", ACM/IEEE Design Automation Conference, pp. 639-644, 1985.
- [WURT04] A. Wurtenberger, C.S. Tautermann et S. Hellebrand, "Data compression for multiple scan chains using dictionaries with corrections", IEEE European Test Symposium, pp. 451-459, 2004.
- [XWEN05] X. Wen, Y. Yamashita, S. Kajihara, L. T. Wang, K. K. Saluja et K. Kinoshita, "On Low-Capture-Power Test Generation for Scan Testing", IEEE VLSI Test Symposium, pp. 265-270, 2005.
- [YOSH03] T. Yoshida et M. Watati, "A New Approach for Low Power Scan Testing", IEEE International Test Conference, pp. 480-487, 2003.
- [ZORI93] Y. Zorian, "A Distributed BIST Control Scheme for Complex VLSI Devices", IEEE VLSI Test Symposium, pp. 4-9, 1993.

Références relatives à l'étude

Ouvrages

[BADE06a] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel et C. Landrault, "Scan Cell Reordering for Peak Power Reduction during Scan Test Cycles", chapitre dans IFIP Series, VLSI-SOC'05 Conference, Springer, 2006, à paraître.

Conférences internationales avec actes et comités de lecture

[BADE06b] N. Badereddine, Z. Wang, P. Girard, K. Chakrabarty, A. Virazel, S. Pravossoudovitch et C. Landrault, "Power-Aware Test Data Compression for Embedded IP Cores", IEEE Asian Test Symposium, 2006, à paraître.

[BADE06c] N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel et H-J. Wunderlich, "Structural-Based Power-Aware Assignment of Don't Cares for Peak Power Reduction during Scan Testing", IFIP International Conference on Very Large Scale Integration, 2006, à paraître.

[BADE06d] N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel et H-J. Wunderlich, "Minimizing Peak Power Consumption during Scan Testing: Test Pattern Modification with X Filling Heuristics", IEEE Design and Test of Integrated Systems, 2006, à paraître.

[BADE06e]¹ N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel et H-J. Wunderlich, "Minimizing Peak Power Consumption during Scan Testing: Structural Technique for Don't Care Bits Assignment" IEEE Ph.D. Research in Microelectronics and Electronics, pp. 65-68, 2006.

[BADE05a]² N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault et A. Virazel, "Peak Power Consumption During Scan Testing: Issue, Analysis and Heuristic Solution", IEEE Workshop on Design and Diagnostics of Electronic Circuits and Systems, pp. 151-159, 2005.

¹ Bronze Certificate (20-30% best papers).

² Best Paper Award.

[BADE05b] N. Badereddine, P. Girard, A. Virazel, S. Pravossoudovitch et C. Landrault, "Why Do We Need to Reduce Peak Power Consumption During Scan Capture?", International Workshop on Power and Timing Modeling, Optimization and Simulation, pp. 540-549, 2005.

[BADE05c] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel et C. Landrault, "Power-Aware Scan Testing for Peak Power Reduction", IFIP International Conference on Very Large Scale Integration, pp. 441-446, 2005.

Conférences sans actes ou avec actes à diffusion restreinte

[BADE06f] N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel et H-J. Wunderlich, "Structural-Based Power-Aware Assignment of Don't Cares for Peak Power Reduction During Scan Testing", IEEE European Test Symposium, 2006.

[BADE06g] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel et C. Landrault, "Réduction de la Consommation de Puissance de Pic Pendant le Test Série", Journée des Doctorants de l'école doctorale Information, Structures et Systèmes, 2006.

[BADE06h] N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel et H-J. Wunderlich, "Technique Structurale d'Affectations des Bits Non Spécifiés en Vue d'une Réduction de la Puissance de Pic Pendant le Test Série", Journées Nationales du Réseau Doctoral en Microélectronique, 2006.

[BADE06i] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel, C. Landrault et H-J. Wunderlich, "Peak Power Reduction During Scan testing", South European Test Seminar, 2006.

[BADE05d] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel et C. Landrault, "Analyse et Réduction de la Puissance de Pic durant le Test Série", Journées Nationales du Réseau Doctoral en Microélectronique, pp. 502-504, 2005.

[BADE05e] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel et C. Landrault, "Peak Power Consumption During Scan testing", South European Test Seminar, 2005.

[BADE04] N. Badereddine, P. Girard, S. Pravossoudovitch, A. Virazel et C. Landrault, "Low Power testing For Soc Cores", South European Test Seminar, 2004.

Listes des figures, tableaux et organigrammes

Listes des figures, tableaux et organigrammes

Figure I.1 Description d'un circuit séquentiel muni d'une chaîne de scan	24
Figure I.2 Charge et décharge de la capacité de sortie d'un inverseur	26
Figure I.3 Mise en évidence de l'activité de commutation lors du test par scan	29
Figure I.4 Modification apportée à la bascule scan	32
Figure I.5 Segmentation de la chaîne de scan	33
Figure I.6 Illustration de la technique par vecteur d'entrée précalculé	34
Figure I.7 Courant consommé pendant le test par scan	36
Figure II.1 Exemple des commutations dans la chaîne de scan pendant le cycle de test	46
Figure II.2 Exemple d'ordonnancements des cellules scan	47
Figure II.3 Exemple d'optimisation globale	49
Figure II.4 Exemple de construction d'un graphe pondéré	50
Figure II.5 Exemple d'une optimisation globale sous contrainte	54
Figure II.6 Recherche du minimum global avec l'algorithme de recuit simulé	55
Figure II.7 Exemple de réduction du nombre de transitions par vecteur de test avec l'algorithme de recuit simulé pour le circuit s713	62
Figure III.1 Pourcentage de bits spécifiés dans les vecteurs de test	69
Figure III.2 Procédure d'évaluation des techniques 0, 1 et MT-Filling	72
Figure III.3 Exemple de portes connectées aux cellules de la chaîne de scan	76
Figure III.4 Procédure du déroulement de la technique SB-Filling	79
Figure IV.1 Architecture générique de compression de données de test	93
Figure IV.2 Compression horizontale et verticale	94
Figure IV.3 Exemple de réordonnement des vecteurs de test	95
Figure IV.4 Illinois Scan Architecture	96
Figure IV.5 Exemple de reconstitution d'un vecteur de test en scan slices	100
Figure IV.6 Représentation d'un Code-Slice	100
Figure IV.7 Schéma général de la technique de compression [WANG05]	102
Tableau I.1 Caractéristiques des circuits expérimentés	39
Tableau I.2 Consommation de puissance de pic pendant le test par scan	40
Tableau II.1 Résultats comparatifs entre l'algorithme glouton et une solution industrielle	53
Tableau II.2 Résultats comparatifs entre l'algorithme de recuit simulé et une solution industrielle	59
Tableau II.3 Réduction de puissance de pic pendant les cycles de décalage	60

Tableau II.4 Résultats comparatifs des taux de couverture de fautes de transition non robuste	61
Tableau III.1 Caractéristiques des séquences de test partiellement spécifiées	73
Tableau III.2 Réduction de puissance de pic pendant le cycle de test	74
Tableau III.3 Équations de satisfiabilité pour les portes AND/NAND – OR/NOR avec une ou deux entrées connectées aux cellules de la chaîne de scan	77
Tableau III.4 Nombre d'équations à satisfaire	80
Tableau III.5 Réduction de puissance de pic pendant le cycle de test	81
Tableau III.6 Réduction de puissance de pic pendant les cycles de décalage	82
Tableau III.7 Résultats comparatifs de la longueur des séquences de test	83
Tableau III.8 Résultats comparatifs du taux de couverture de fautes de collage	84
Tableau III.9 Résultats comparatifs du taux de couverture de fautes de transition non robuste	85
Tableau IV.1 Exemple de codage	102
Tableau IV.2 Exemple de codage	104
Tableau IV.3 Résultat de la décompression	104
Tableau IV.4 Tous les bits à 'X' sont affectés à '0'	105
Tableau IV.5 Tous les bits à 'X' sont affectés à '1'	105
Tableau IV.6 Exemple de modification des scan slices avec la technique MT-Filling	106
Tableau IV.7 Résultats de compression	108
Tableau IV.8 Réduction de consommation de puissance	109
Tableau IV.9 Caractéristiques des circuits expérimentés	110
Tableau IV.10 Résultats de compression	111
Tableau IV.11 Réduction de consommation de puissance	112
Organigramme II.1 Algorithme Glouton	51
Organigramme II.2 Procédure du déroulement de l'algorithme de recuit simulé	57

Annexe A

Estimation de la consommation de puissance par la métrique de transitions pondérées [SANK00]

Annexe A : Estimation de la consommation de puissance par la métrique de transitions pondérées [SANK00]

La métrique de transitions pondérées (*Weighted Transitions WT*) permet d'évaluer l'activité de commutation dans la chaîne de scan due à une transition contenue dans le vecteur de test de manière rapide et efficace. En effet, une transition n'engendre pas le même nombre de commutations dans la chaîne de scan selon sa position dans le vecteur de test. Pour estimer l'influence d'un vecteur de test et de la réponse associée sur l'activité de commutation dans la chaîne de scan lors de leur chargement/déchargement, l'auteur de [SANK00] a introduit une technique basée sur une pondération des transitions présentes dans les vecteurs de test et les réponses du circuit.

Le nombre WT_i de commutations dans la chaîne de scan dues à une transition i dans le vecteur de test V peut être exprimé par la relation suivante :

$$WT_i = (N - Post_i)$$

Où N est la taille de la chaîne de scan et $Post_i$ la position de la transition i dans V . Le nombre WT_V de commutations dans la chaîne de scan dues aux transitions dans V peut alors être exprimé par la relation suivante :

$$WT_V = \sum_i (N - Post_i)$$

Appliquons ceci au vecteur "10001". Dans ce cas, WT_V qui correspond au chargement de ce vecteur de test est égal à la somme des WT de chaque transition, c'est-à-dire $(5 - 1) + (5 - 4)$ (où 5 est égal à la taille de la chaîne de scan, 1 est la position de la première transition et 4 la position de la deuxième transition). Donc WT_V est égal à 5 pour ce vecteur de test.

Nous avons vu dans la partie 1.5 que l'activité de commutation dans la chaîne de scan est due à la fois au chargement du vecteur de test mais aussi au déchargement de la réponse du circuit. Il est donc nécessaire d'évaluer l'influence du déchargement de la réponse du circuit sur l'activité de commutation dans la chaîne de scan. Le principe est le même que pour l'évaluation de l'influence du chargement du vecteur de test sur l'activité de commutation dans la chaîne de scan, le seul changement concerne la position des transitions (ordre inversé par rapport au chargement). Par exemple, considérons la réponse "01001". Cette réponse comporte trois transitions. Dans ce cas, WT_V correspondant au déchargement de cette réponse est égal à 8, c'est-à-dire $(5 - 4) + (5 - 2) + (5 - 1)$ (où 5 est la longueur de la chaîne de scan, 4, 2 et 1 sont les positions des trois transitions).

Pour un calcul exact du WT total, il faut considérer la transition qui peut exister entre le premier bit du vecteur de test et le dernier bit de la réponse précédente. Cette transition génère N commutations dans la chaîne de scan lors du chargement du vecteur de test. Considérons le vecteur de test "10001" et la réponse précédente "01001" qui a été capturée dans la chaîne de scan. Le dernier bit de la réponse est '0' tandis que le premier bit du vecteur de test suivant est '1'. Dans ce cas $WT_{Total} = WT_{Total} + N$. La Figure A.1 présente un exemple de calcul de WT_{Total} suite au déchargement de la réponse "01001" et au chargement du vecteur de test "10001" dans une chaîne de scan avec 5 cellules.

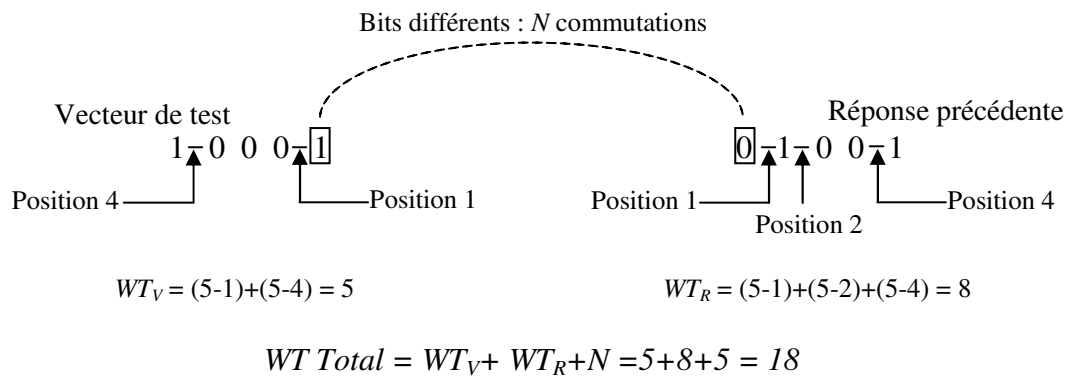


Figure A.1 Exemple de calcul de WT Total

REDUCTION DE PUISSANCE DURANT LE TEST PAR SCAN DES CIRCUITS INTEGRES

Thèse de Doctorat – Nabil BADEREDDINE

Cette thèse s'inscrit dans le cadre de la réduction de la consommation de puissance durant le test par scan des circuits intégrés. Le test par scan est une technique de conception en vue du test qui est largement utilisée, mais qui pose quelques problèmes. Elle nécessite en effet un nombre important de cycles d'horloge pour permettre le chargement, l'application, et le déchargement des données de test. Ces opérations engendrent une activité de commutation dans le circuit largement plus importante que celle rencontrée lors du fonctionnement normal. Cette forte activité lors du test peut avoir des conséquences graves sur le circuit telles qu'une baisse de sa fiabilité ou sa destruction pure et simple. L'objectif de cette thèse est de proposer des techniques permettant de réduire cette suractivité, en particulier durant la période comprise entre l'application d'un vecteur de test et la récupération de la réponse du circuit.

Mots clés : Test, SCAN, Test Faible Consommation, Consommation de Puissance de Pic, Compression de Données.

POWER REDUCTION DURING SCAN TESTING

This thesis relates to power minimization during scan testing. The Scan technique is considered as the most often used DfT (Design for Test) technique. During test, scan-based architectures require a large number of operations to load, apply, and unload test data. All these operations produce a switching activity which is much higher than that during functional mode. For this purpose, we propose several solutions to minimize the power consumption during scan testing, and particularly during the period of time comprised between the application of a test vector and the capture of the circuit response. These solutions allow safe and no destructive testing of the circuit under test.

Keywords: Test, DFT, SCAN, Low Power Testing, Peak Power Consumption, Data Compression.

DISCIPLINE GENIE INFORMATIQUE, AUTOMATIQUE ET TRAITEMENT DU SIGNAL

Université de Montpellier II : Sciences et Technique du Languedoc

LIRMM : Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier

161, rue Ada 34392 Montpellier Cedex 5.