NOTE TO USERS

This reproduction is the best copy available.

UMI®

UNIVERSITÉ DE MONTRÉAL

DEFECT TOLERANCE METHODS AND THERMALLY INDUCED SKEW ANALYSIS FOR WAFER SCALE INTEGRATION

MENG LU

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE ET DE GÉNIE INFORMATIQUE

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION

DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES

(GÉNIE ÉLECTRIQUE)

APRIL 2003

© MENG LU, 2003

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-97964-4
Our file Notre référence
ISBN: 0-612-97964-4

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

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

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

DEFECT TOLERANCE METHODS AND THERMALLY INDUCED SKEW ANALYSIS FOR WAFER SCALE INTEGRATION

présenté par : MENG LU

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

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

M. KHOUAS Abdelhakim, Ph.D., président

M. SAVARIA Yvon, Ph.D., membre et directeur de recherche

Mme. WANG Chunyan, Ph.D., membre et codirectrice de recherche

M. THIBEAULT Claude, Ph.D., membre

ACKNOWLEDGEMENTS

I wish to thank my supervisor Professor Yvon Savaria for his encouragement, and support over the course of this work. Yvon was always there to give me the best guidance in theoretical, experimental and practical matters and led me to channel my work in the most productive manner. I would also like to acknowledge my co-supervisor Professor Chunyan Wang for her guidance on circuit level design and simulation, especially, for her effort on finalizing the thesis. I am grateful to Professor Claude Thibeault for numerous discussions on testing and fabrication issues, and for allowing me open access to his laboratory facilities.

Many thanks to Renaud Tiennot, Normand Leclerc, Bing Qiu and Ara Talaslian for their collaborations and efforts on delivering the defect tolerance demonstrator chip. Thanks to Marc-Andrew Canton and Sébastien Clavier for their help on testing the demonstrator chip. Thanks to Daniel Picard, David Claveau for their corrections on my French and English. I was very fortunate to have worked with several fantastic guys in the project, Boris Polianskikh, Stephane Nadeau,.Zhongfang Jin, Lakhsasi, Bougataya.,

I am grateful to Hyperchip Inc for their financial assistance. Many thanks to David Chamberlain and Karl Fecteau.

Finally, I acknowledge with heartfelt gratitude the love and support of my family.

RÉSUMÉ

L'évolution naturelle des technologies de l'information nécessite des méthodes d'intégration aux performances croissantes. Afin de répondre aux besoins des nouvelles technologies, des recherches sont effectuées pour tendre vers une intégration à l'échelle de la tranche. L'objectif de ce mémoire est d'étudier plusieurs facteurs qui influencent l'intégration à l'échelle de la tranche à l'aide d'un prototype. Il sera question, plus particulièrement, d'un commutateur Internet de génération future. Cette étude démontrera que l'intégration a des conséquences thermiques qui se répercutent sur l'intégrité des signaux. De plus, le niveau d'intégration visé rend nécessaire la tolérance des défectuosités et pose des défis accrus pour les tests.

Les biais de synchronisation induits par un gradient thermique sont caractérisés par une analyse mathématique ainsi que par des simulations. Le gradient de température induit un biais de synchronisation qui est approximé par une relation linéaire. Jusqu à 70% du biais de synchronisation induit par le gradient de température est causé par la variation des résistances des interconnexions. Ceci nous amène à estimer le biais de synchronisation maximal qui découle de ce gradient thermique. Il est prouvé que le biais de synchronisation induit par le gradient de température n'est pas significatif pour des interconnexions locales aussi bien que pour les bus globaux.

Ce mémoire présente une structure complète d'interconnexion, une stratégie innovatrice pour tolérance aux fautes, ainsi qu'une topologie régulière. Basé sur des recherches antérieures, nous proposons une configuration « pseudo-5-source » permettant de supporter douze (12) remplacements uniques requérant moins d'interconnexions qu'auparavant. Cette structure unique pour la tolérance aux pannes permet de tolérer des regroupements de connexions et cellules défectueuses. La structure peut supporter plusieurs modèles de remplacement à la fois. Le remplacement intra-bus des interconnexions permet l'utilisation d'un autre groupe au sein d'un bus. Cette option

permet de remplacer plusieurs types de cellules ou interconnexions défectueuses. Un ensemble de règles de configuration a été développé en fonction des limitations des connexions de la structure d'interconnexion. Nous analysons l'impact de l'algorithme de configuration sur la structure d'interconnexion. Une proposition d'algorithme de configuration global vorace sera discutée dans ce mémoire. La stratégie de cet algorithme est de laisser le plus d'éléments possibles dans l'assignation courante afin de maximiser la flexibilité des tâches subséquentes. Cette stratégie valorise l'exécution des calculs afin de configurer les défectuosités localement pour minimiser la complexité des remplacements. Nous avons proposé l'ébauche d'une stratégie de test, de configuration ainsi que de diagnostic de la structure. Afin de valider les recherches effectuées sur cette structure tolérante aux défectuosités, nous avons créé un prototype et nous lui avons fait subir une séquence de tests.

Nous proposons une chaîne de balayage pour exécuter les tests et la configuration à l'échelle de la tranche. La norme IEEE 1149.1 basé sur la tolérance aux défectuosités peut être implantée pour réaliser les essais voulus. Dans la structure proposée, les éléments de la chaîne du JTAG sur le chemin de données global, le contrôle, l'alimentation et l'horloge sont implémentés en triple. La robustesse de cette structure tolérante aux pannes ne permet pas à une seule défectuosité, soit d'une cellule ou d'une connexion, de rendre le système inopérant. Également, la structure de chaîne de balayage peut tolérer plusieurs types de défectuosités. La régularité du dessin de masques peut supporter une connexion inter-réticule.

ABSTRACT

As a natural evolution of conventional device-oriented VLSI chip technologies, the need for functional integration in high-performance systems (such as servers, communications equipment, high-end computers) has led to system-oriented wafer-level technologies. Beginning with the characterization of the thermal gradient induced skew, several aspects (i.e. signal integrity, fault tolerance and testing) of a Wafer Scale Integration (WSI) router prototype are investigated in this thesis.

The temperature gradient induced skew is studied by theoretical analysis and is characterized by simulation. A temperature gradient induces approximately linear skew, of which 70% is contributed by resistance variations of the interconnections. Temperature gradient induced skew is negligible for local interconnections as well as for global buses.

The central part of this thesis presents a fault tolerant scheme. Based on earlier work, the pseudo-5-source configuration structure which supports 12 basic replacement operations and that requires less interconnect overhead has been proposed. It is a very strong fault tolerant scheme, which can tolerate defects in cells as well as defects on long interconnections. The intra-bus bundle replacement supports the bundle shift along the second direction, which makes the repair of more clustered defect patterns feasible. Configuration algorithms and the procedures of test, diagnosis, and configuration are discussed. A set of configuration rules based on physical interconnect constraints are formalized. The impacts of this interconnection structure on the configuration algorithm are investigated. The Greedy Global Mapping Configuration Algorithm is proposed. The strategy of this algorithm is trying to leave as many elements as possible for the following assignments and making all the computation locally to minimize computational complexity. To verify the functionality of the fault tolerant structure, a prototype demonstrator chip is developed

The fault tolerant scan chain is proposed to perform wafer scale testing and configuration. The global IEEE 1149.1 based fault tolerant scan chain can be built on-wafer. In the proposed designs, the critical portion of data paths, power supply, and control are tripled. There is no single-point of defect that can make this global scan chain inoperative and multiple faults can be tolerated. The layout regularity is supported by the newly proposed inter-reticle stitching method and dummy off-wafer JTAGs.

CONDENSÉ EN FRANÇAIS

ANALYSE DES BIAIS DE SYNCHRONISATION INDUITS PAR DES GRADIENTS THERMIQUES ET D'UN ENSEMBLE DE MÉTHODES DE TOLÉRANCE AUX DÉFECTUOSITÉS POUR L'INTÉGRATION À L'ÉCHELLE DE LA TRANCHE

I. INTRODUCTION

L'évolution naturelle des technologies de l'information nécessite des méthodes d'intégration aux performances croissantes. Afin de répondre aux besoins des nouvelles technologies, des recherches sont effectuées pour tendre vers une intégration à l'échelle de la tranche. L'objectif de ce mémoire est d'étudier plusieurs facteurs qui influencent l'intégration à l'échelle de la tranche à l'aide d'un prototype. Il sera question, plus particulièrement, d'un commutateur Internet de génération future. Cette étude démontrera que l'intégration a des conséquences thermiques qui se répercutent sur l'intégrité des signaux. De plus, le niveau d'intégration visé rend nécessaire la tolérance des défectuosités et pose des défis accrus pour les tests.

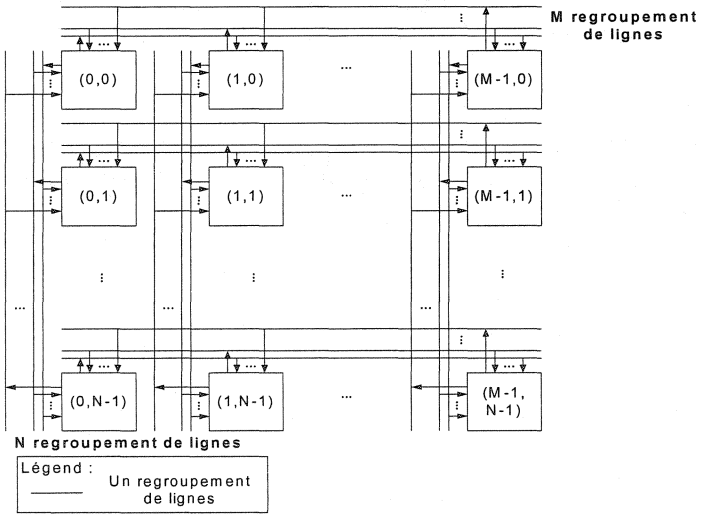


Figure 0.1 L'architecture ciblée de débit élevé. Une matrice de M x N

L'architecture ciblée, représentée par la figure 0.1, est une matrice de processeurs, dans laquelle une cellule communique avec d'autres cellules par l'entremise de bus orientées suivant une structure de rangées et de colonnes. Prenons par exemple la matrice M x N. Chaque cellule peut communiquer avec (M - 1) + (N - 1) autres cellules, (M-1) sur sa rangée et (N-1) sur sa colonne. Cette cellule émet sur une ligne d'un groupe horizontal et sur une ligne d'un groupe vertical, alors que les autres cellules dans la même rangée et dans la même colonne reçoivent les données émises sur ces interconnexions. Tous les liens entre les cellules sont unidirectionnels et contiennent des répéteurs pour régénérer les signaux. Cette structure de connexions unique, couplée à une intégration à l'échelle de la tranche, pose des problèmes nouveaux.

Dans la théorie classique de la réduction à l'échelle, la consommation d'énergie ne devrait pas causer de problème significatif. Par contre, dans la pratique, nous observons une tendance rapide à la hausse dans la consommation d'énergie. On sait que la température élevée et la dissipation inégale de puissance causent des points chauds et froids qui peuvent engendrer une diminution de la durée de vie des circuits et des changements paramétriques lors de l'exécution. Ceci provoque des gradients de température qui se répercutent comme des différences de temps de propagation des signaux des différents fils sur les bus qui introduisent des biais de synchronisation. Une déviation de phase excessive est problématique, car elle cause des problèmes de synchronisation pour les signaux qui doivent arriver plus ou moins simultanément dans un système informatique. Un but de ce mémoire est de caractériser la déviation induite par le gradient de la température.

Il est bien connu que le rendement de circuits sans redondance chute rapidement à mesure que la surface occupée accroît au-dessus d'un certain niveau. Diverses méthodes qui permettent de repousser les limites dues au rendement des circuits de grande taille sont basées sur la tolérance aux pannes et aux défectuosités. De plus, pour tolérer les pannes, il est souvent nécessaire de diagnostiquer la présence de pannes grâce à des tests. La

figure 0.2 illustre une stratégie commune, qui consiste à fractionner la fonctionnalité d'un système en modules et d'y ajouter des modules redondants. Ainsi, on peut contourner les défectuosités grâce aux modules redondants. Plusieurs techniques ont été développées pour configurer en présence de contraintes. Cependant, aucune de ces techniques n'est applicable à la structure ciblée. Une stratégie de configuration appropriée, une méthode de diagnostic, ainsi qu'une procédure de test doivent donc être développées.

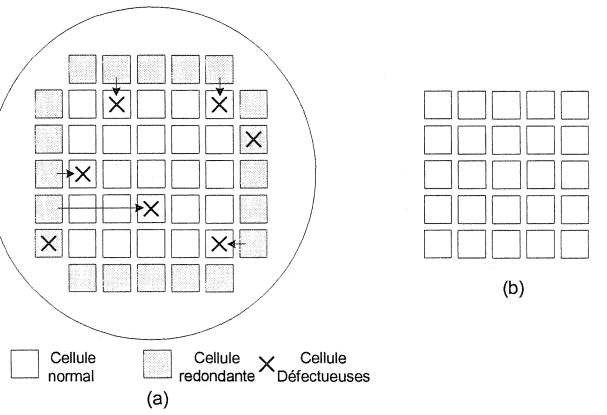


Figure 0.2 (a) Vue d'une matrice de processeurs sur une tranche
(b) Matrice logique ciblée

II. CARACTÉRISATION DU BIAIS DE SYNCHRONISATION INDUIT PAR UN GRADIENT DE TEMPÉRATURE

Les biais de synchronisation induits par un gradient thermique sont caractérisés par une analyse mathématique ainsi que des simulations. Ceci nous amène à estimer le biais de synchronisation maximal qui découle de ce gradient thermique.

1. Analyse mathématique

Les variations de température, sur une puce de grande taille, modulent les performances paramétriques comme les temps de commutation et les résistances des matériaux. En

supposant que la température est uniforme le long d'une ligne, le biais induit par un gradient de température transversal peut être exprimé comme suit :

$$\Delta D = \rho_0 \beta \left(C_L L + C_0 \frac{L^2}{2} \right) \Delta T \qquad (1)$$

Où:

ρ_0 : La résistance par unité de longueur à la température de référence (à savoir 0 °C);
β : Le coefficient en température de la résistance (1/°C);
L : La longueur d'interconnexion;
C_L : La capacité de charge;
C_0 : La capacité par unité de longueur;
ΔT : La différence de température.

Pour un circuit donné, $\rho_0 \beta \left(C_L L + C_0 \frac{L^2}{2} \right)$ est complètement déterminé. Dans ce cas, le biais de synchronisation induit varie linéairement avec le gradient de température.

2. Caractérisation par simulation

Des simulations ont été effectuées grâce au simulateur de circuit SPECTRE. La structure simulée est présentée à la figure 0.3. Elle comporte 9 lignes parallèles placées selon un pas de 3 μm. Le modèle électrique assume que les interconnexions utilisent la couche métal 5, que les fils ont une largeur de 0.76 μm et que le fil fait 3750 μm de long. Chaque fil est commandé par deux inverseurs configurés en parallèle. Les résultats se rapportent aux fils extérieurs ainsi qu'au fil au centre du bus et les interconnexions non stimulées sont connectées à un potentiel fixe qui correspond à un 1 logique.

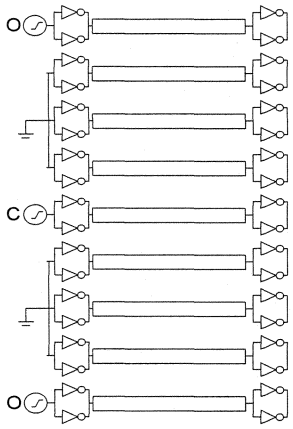
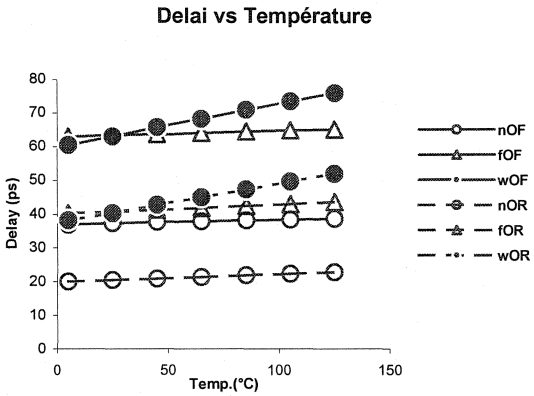


Figure 0.3 Le circuit de simulation



n------ Courbe sans extraction de résistance pour un long fil.
f------ Courbe illustrant une extraction de résistance pour laquelle le coefficient de température est de 0.
w ------ Courbe illustrant une extraction de résistance pour un long fil.
O ------ Fil aux extrémités du bus
R ------ Signal d'entrée excité par une transition montante.
F ------ Signal d'entrée excité par une transition descendante.

Figure 0.4 Résultats de simulation

Lors des simulations, le gradient de température induit un biais de synchronisation qui est approximé par une relation linéaire. Les résultats de simulation sont montrés à la figure 0.4. Les résultats démontrent que 70% du biais de synchronisation induit par le gradient de température est causé par la variation des résistances d'interconnexions, ce qui n'est pas un phénomène dominant.

3. Estimation du biais de synchronisation sur les interconnexions à l'échelle de la tranche

Nous supposons que le gradient thermique le plus prononcé sur la tranche est à la jonction séparant un groupe de cellules les plus chaudes et un groupe de cellules les plus froides, représenté sur la figure 0.5. Nous supposons également que les cellules les plus chaudes correspondent à une densité de puissance obtenue lorsque le circuit présente une activité maximale. Ceci ce traduit par des oscillateurs en anneau composés de 3 inverseurs (circuits les plus actifs). Par contre, les cellules les plus froides peuvent être représentées par des portes « NON-ET » dont leurs entrées sont fixes (seulement la puissance statique est absorbée).

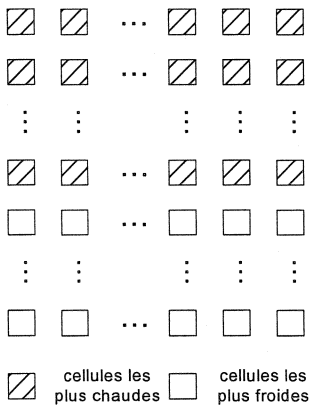


Figure 0.5 Situation pouvant produire un gradient de température maximal

D'après les résultats de simulations thermiques [1] réalisées par Lakhsasi et Bougataya, le gradient de température maximal, pour une structure de bus excédant une largeur de 0,2 mm, peut atteindre 1°C. D'après les simulations effectuées, une déviation de 7,6 psec peut être perçue lorsque le biais de synchronisation maximum est induit par un gradient de température de 1°C lors d'une configuration de bus de 20 centimètres de long, ce qui est négligeable.

III. STRUCTURE D'INTERCONNEXION RÉGULIÈRE ET STRATÉGIE DE CONFIGURATION

1. Structure d'interconnexion régulière de Pseudo-5-sources.

Dans ce mémoire, nous allons proposer une structure complète d'interconnexions, une stratégie innovatrice tolérante aux fautes ainsi qu'une topologie régulière. Ces recherches ont été basées sur des travaux réalisés par Thibeault et Norman. Les interconnexions régulières d'une cellule structurée en modules de 3 x 3 sont illustrées à la figure 0.6. Afin d'alléger cette figure, chacune des lignes représente un regroupement de fils. Les lignes en gras représentent les bus de données réguliers et les lignes légères représentent des liens de configuration. Les multiplexeurs sont utilisés pour aiguiller les liens. Le cœur de la logique, qui effectue les traitements, doit être positionné au centre afin de conserver une structure régulière. Étant donné l'avantage de la modularité d'une cellule, une structure d'interconnexion régulière de 3 x 3 cellules peut aisément être générée, voir la figure 0.7.

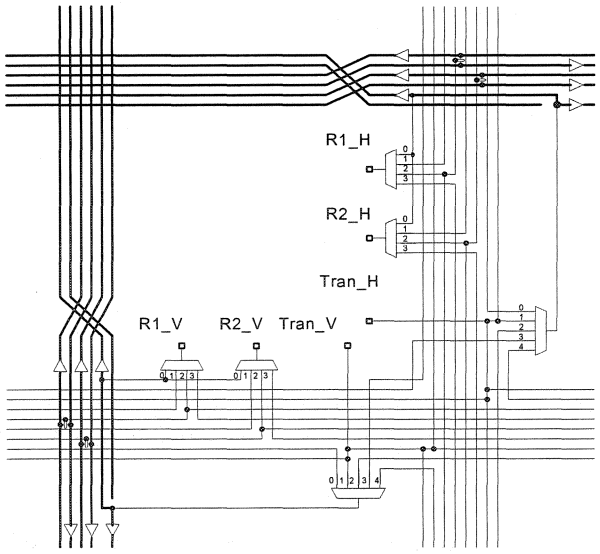


Figure 0.6. Représentation d'une cellule ayant une structure d'interconnexion Pseudo-5

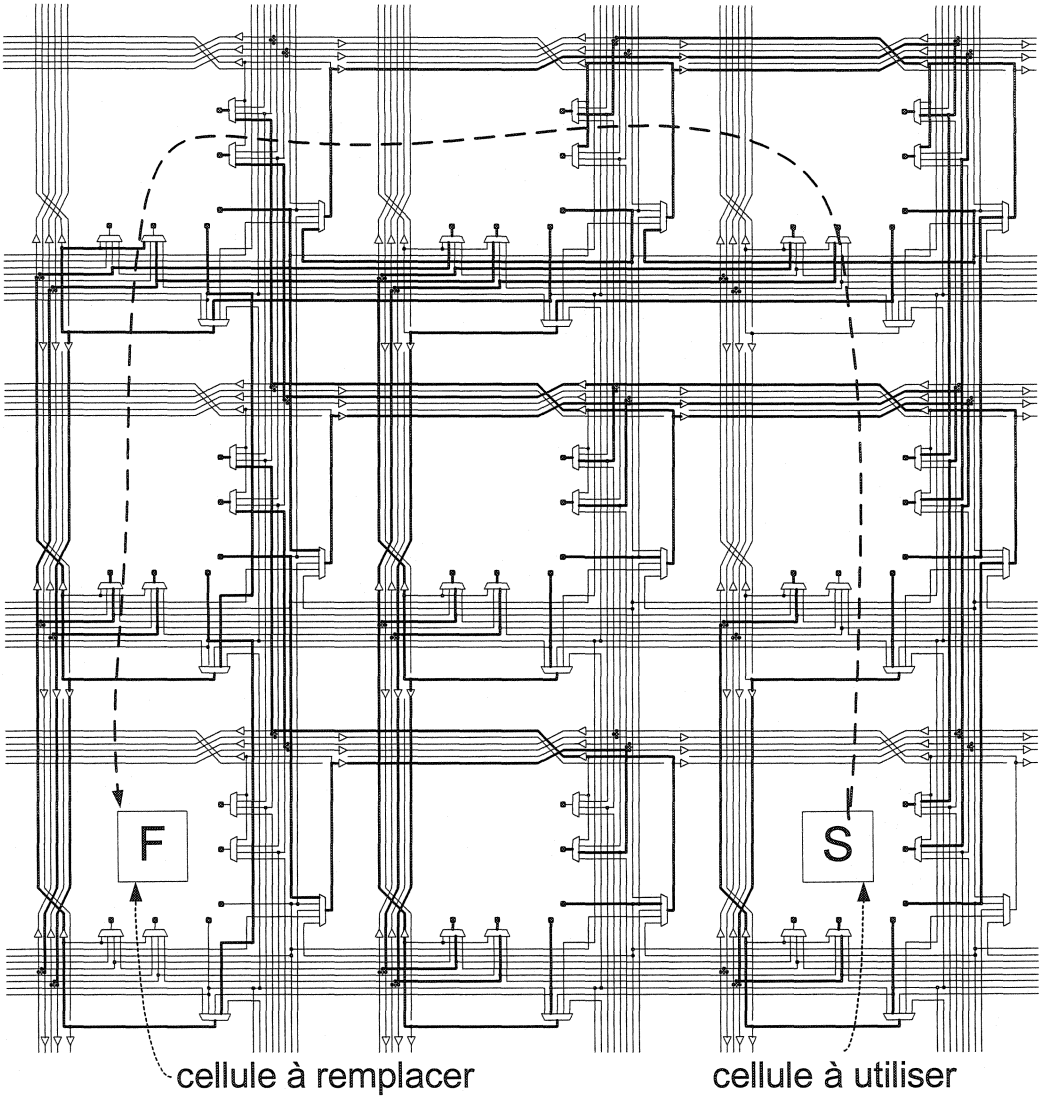


Figure 0.7 Un remplacement de chemin tourné par la combinaison de remplacements uniques

Cette structure unique est tolérante aux pannes. Elle permet de tolérer les regroupements de cellules défectueuses. Il existe trois (3) modèles de remplacements soutenus par cette structure. Le premier modèle de remplacement est lorsqu'une cellule apparaît défectueuse. Elle peut être directement remplacée par une de ses quatre voisines, soit celle de dessus, du bas, de gauche ou bien de droite. Le second modèle de remplacement est le regroupement « Inter-bus ». Lorsqu'un groupe de connexion est défectueux, il peut être directement remplacé par un bus voisin du même rang. Le dernier modèle de

remplacement est le regroupement « Intra-bus ». Lorsqu'un groupe de connexion est défectueux, il peut être directement remplacé par un des groupes voisins du même bus. Lorsqu'il y a plusieurs défectuosités, la structure de remplacement peut supporter plusieurs modèles de remplacement à la fois. Elles doivent cependant respecter certaines contraintes pour ne pas créer des conflits de remplacement entre les cellules et/ou connexions. La figure 0.7 démontre qu'un remplacement par un chemin de contournement est réalisé par une combinaison des remplacements de cellules et de regroupement de lignes. Le remplacement d'une cellule par un chemin de contournement implique que la cellule ne sera pas remplacée par le chemin le plus court entre la cellule défectueuse et la cellule de remplacement. Ainsi, à l'aide de ce système, nous pouvons rétablir le bon fonctionnement à l'aide des trois modèles lors de défectuosités de plusieurs natures, tout en profitant des avantages des trois (3) modèles énumérés précédemment. La figure 0.8 illustre plusieurs exemples de remplacement, pour lesquels les lignes pointillées en gras représentent des colonnes logiques.

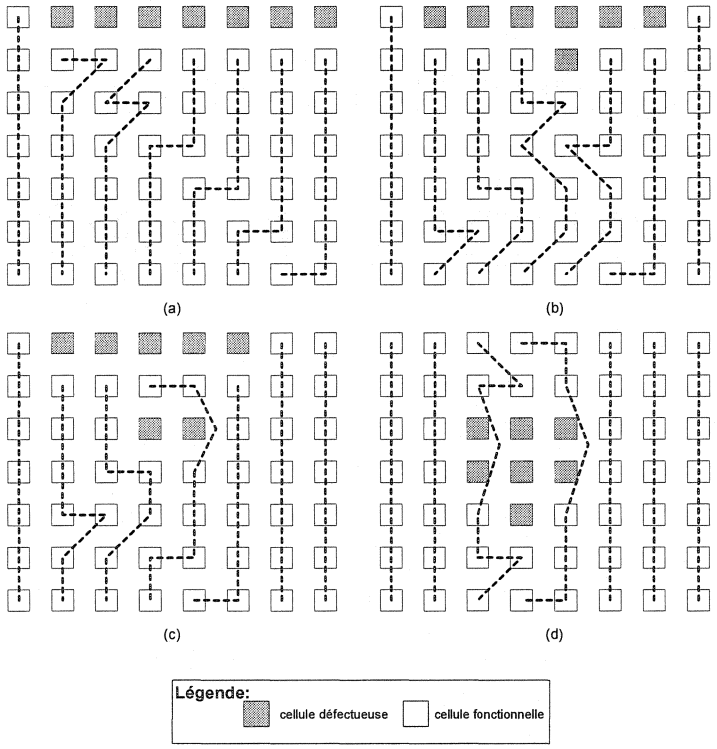


Figure 0.8 Restructuration des modèles de défectuosités groupés

2. Algorithme de configuration

Pour développer un algorithme de configuration approprié, il est essentiel de clarifier les limitations des connexions possibles de la structure d'interconnexion et de développer une stratégie heuristique de configuration. Ce développement permettra d'optimiser les remplacements des modules défectueux dans un système sujet à une configuration de pannes donnée.

Règles de configuration

Un ensemble de règles de configuration sera développé en fonction des limitations des connexions de la structure d'interconnexion. Cet ensemble de règles est indépendant des algorithmes de configuration, dans la mesure où il capture toutes les contraintes pertinentes pour la configuration.

Algorithme de configuration vorace global

Nous avons vu précédemment la nécessité d'un algorithme permettant de réaliser la configuration des cellules et des lignes défectueuses. Une proposition d'algorithme de configuration vorace global sera discutée dans ce mémoire. La stratégie de cet algorithme est de privilégier les solutions qui laissent le plus d'éléments disponibles dans l'assignation courante afin de maximiser la flexibilité des tâches de configuration subséquentes. Cette stratégie valorise l'exécution des calculs afin de configurer les défectuosités localement pour minimiser la complexité des remplacements.

La stratégie de l'optimisation heuristique adoptée cherche des solutions qui ont de grandes chances de fonctionner du premier coup. Lors de la procédure d'allocation globale d'une matrice physique à une matrice logique ciblée, ces allocations sont choisies de manière à maximiser le taux de survie des allocations suivantes. La figure 0.9 (a)

montre une manière de spécifier une matrice logique. La matrice logique est placée dans le coin inférieur droit. Les nombres dans la cellule définissent la séquence qui sera adoptée pour l'allocation des cellules. La figure 0.9 (b) définit la priorité d'allocation des cellules physiques à une cellule logique selon une approche vorace. La priorité d'allocation des groupes de lignes est définie de manière semblable à l'allocation de cellules logiques. Les ressources physiques (des cellules et des groupes) qui sont localisées près du coin supérieur gauche sont toujours utilisées en premier comme cellule de remplacement. Le nombre maximum de ressources physiques dépend des allocations précédentes.

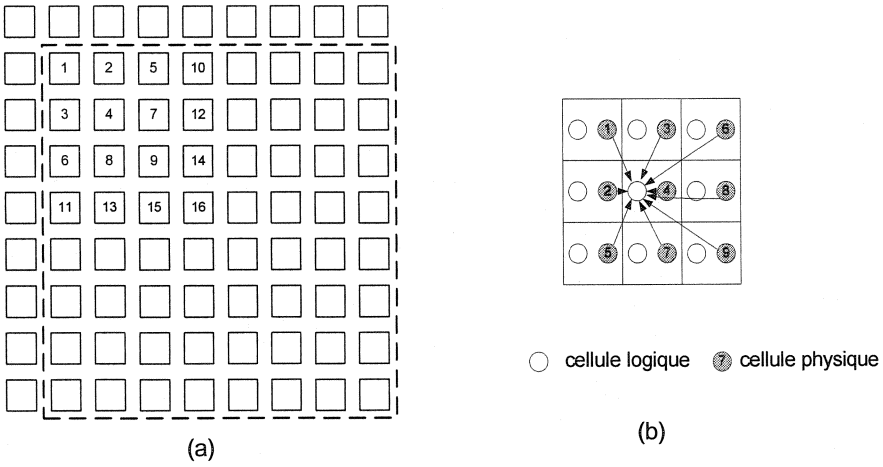


Figure 0.9 (a) Spécification d'une matrice logique possible
(b)Priorité d'allocation des cellules

Selon l'organigramme présenté à la figure 0.10, la configuration est accomplie avec succès si elle respecte trois conditions. La première est que toutes les cellules logiques ont été assignées à une cellule physique. La seconde condition est que toutes les cellules actives ont été assignées à deux groupes du bus. La dernière condition est que toutes les allocations ont passé avec succès la vérification des règles de configuration. La configuration échoue s'il n'y a plus d'alternative disponible pour les éléments défectueux qui ne sont pas encore assignés.

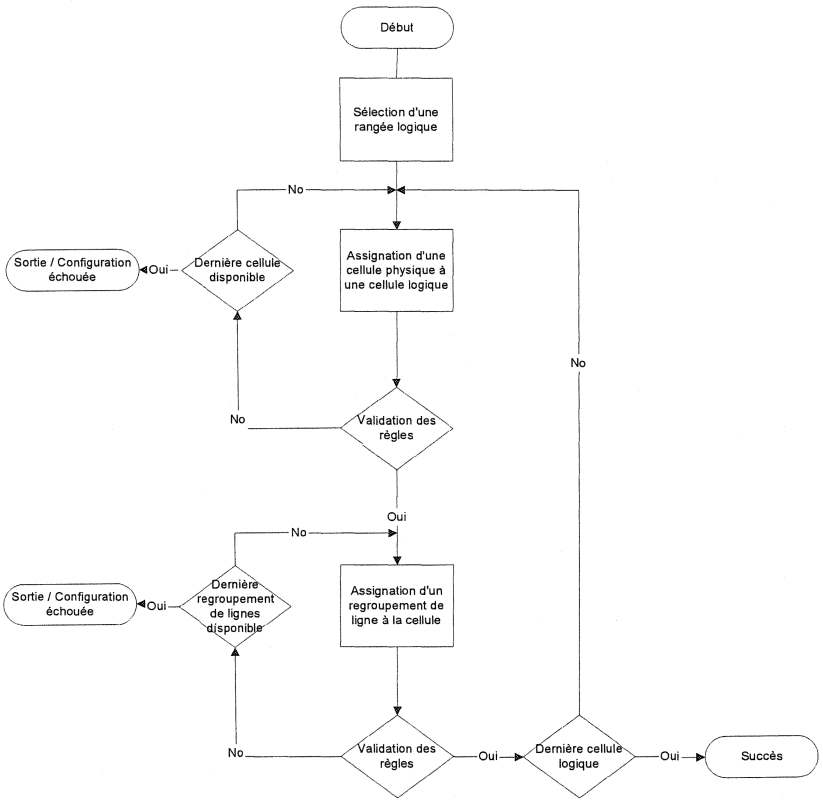


Figure 0.10 Organigramme de la procédure de configuration vorace

L'avantage de cet algorithme est sa faculté de produire rapidement des solutions. Cependant, il ne garantit pas la génération des solutions même si la matrice est configurable. Dans le cas où cet algorithme serait intégré avec un algorithme semblable à PODEM [36], tels que les options de branchements, les limites et la réintégration à la bonne direction, nous croyons qu'il serait possible de faire en sorte qu'il produit une solution de configuration, si la matrice est configurable.

IV. Chaîne de balayage (JTAG) tolérance aux pannes

Comme présenté sur la figure 0.11, une structure basée sur un chaîne de balayage permet d'effectuer les tests et la configuration. Une telle chaîne peut être commandée par un contrôleur JTAG [50], qui fonctionne comme une interface standard au monde extérieur.

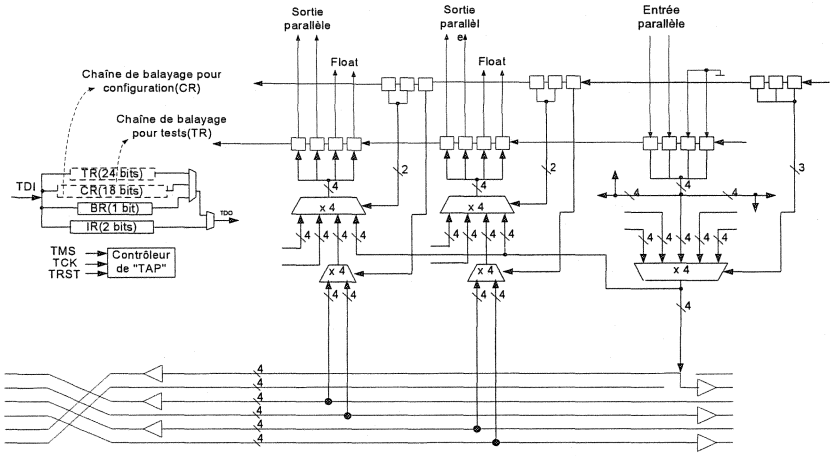


Figure 0.11 Configuration du contrôle et structures de tests

Il est démontré qu'une longue chaîne intolérante aux défectuosités possède un faible rendement, tandis que qu'une chaîne tolérante aux pannes n'a pas d'effet significatif sur le rendement [47]. On propose une mise en oeuvre de mécanismes de tolérance aux pannes compatible avec la norme IEEE1149,1. Comme représenté à la figure 0.12, les éléments du JTAG sur le chemin de donnés global, le contrôle, l'alimentation et l'horloge sont implémentés en triple. Les voteurs peuvent masquer un groupe défectueux de la structure JTAG pour chaque étape. Pour chacune des cellules, les éléments JTAG possèdent leur propre alimentation ainsi qu'un domaine d'horloge distinct. Les trois (3) groupes fonctionnels de la structure JTAG associés à chaque cellule sont distribués sur trois (3) cellules voisines. Cette structure peut tolérer une défectuosité par étage logique. Une panne dans l'horloge ou l'alimentation d'une cellule ne peut pas neutraliser le chemin de données global, puisqu'elle cause seulement une défectuosité dans un étage. Ainsi, une défectuosité du groupe local peut empêcher le contrôle et la configuration d'une cellule, par contre, cela n'est pas critique pour le rendement de la tranche.

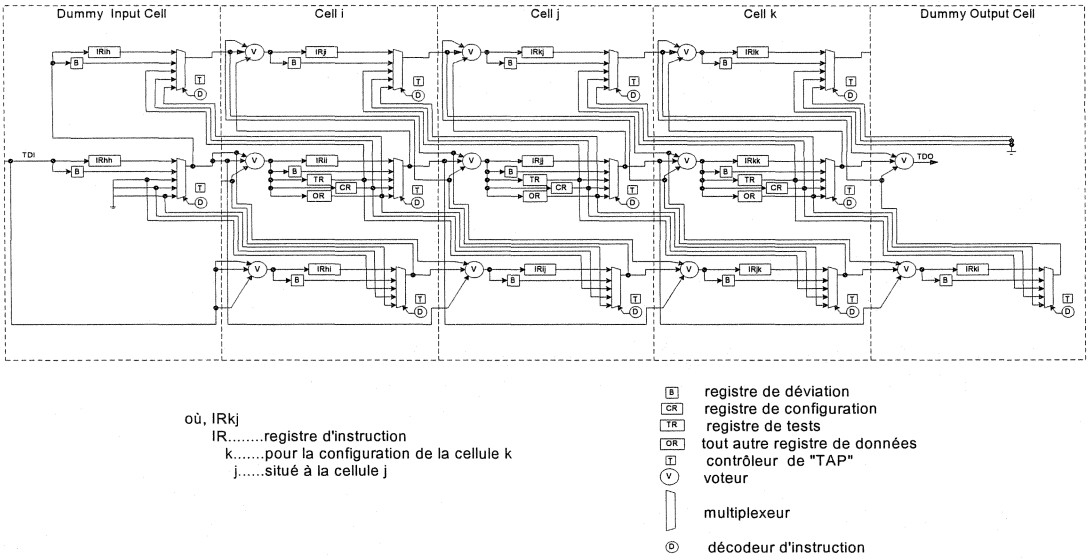


Figure 0.12 Chaîne de balayage tolérance aux pannes à l'échelle de la tranche

V. CONCLUSION ET RÉALISATIONS FUTURES

Dans cette recherche, il a été prouvé que le biais de synchronisation induit par le gradient de température n'est pas significatif pour des interconnexions locales aussi bien que pour les bus globaux. Basé sur quelques travaux antérieurs, nous avons proposé une structure de configuration « Pseudo-5-sources » qui permet trois (3) types de remplacement en utilisant moins d'interconnexions. Avec la flexibilité de ses remplacements, la réparation de plusieurs modèles de défectuosité devient réalisable à l'aide de cette nouvelle structure. Il faut toutefois respecter un ensemble de règles de configuration basées sur les contraintes physiques d'interconnexion pour effectuer des remplacements. Nous avons également proposé un algorithme d'allocation de remplacement global vorace. La configuration et le test se font grâce a une chaîne de balayage. Nous avons proposé une façon de l'implémenter conforme à la norme IEEE1149.1. La structure proposée peut tolérer une panne par étage.

Beaucoup de travail serait requis pour parfaire un prototype complet. La procédure de configuration pourrait être automatisée et améliorée pour être en mesure de garantir une solution lorsqu'elle existe.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iv
RÉSUMÉ v
ABSTRACT vii
CONDENSÉ EN FRANÇAIS ix
TABLE OF CONTENTS xxiv
LIST OF FIGURES xxvii
LIST OF TABLES xxxiii
LIST OF ABBREVIATIONS AND SYMBOLS xxxv
LIST OF ANNEXS xxxviii
CHAPTER 1 1
INTRODUCTION 1
1.1 Motivation 1
1.2 Objectives of Research 4
1.3 Thesis Organization 5
CHAPTER 2 6
CHARACTERIZATION OF SKEW INDUCED BY 6
TEMPERATURE GRADIENT 6
2.1 Skew over Long Interconnections 6
2.1.1 Propagating Delay Modeling 7
2.1.2 Interconnection Scaling Trend and Global Wire Optimization 10
2.1.3 Skew over Very Long Buses 11
2.2 Mathematical Analysis of Skew Induced by a Temperature Gradient 12
2.3 Temperature Gradient Estimation 15
2.4 Simulations and Analysis 17
2.4.1 Simulated Circuit 17
2.4.2 Parameters of the Simulation 18
2.4.3 Simulation Results and Analysis 18

2.4.4 Skew Estimation on Wafer Scale Interconnections 20
2.5 Demo Chip to Characterize Temperature Gradient Induced Skew 21
2.6 Summary 21
CHAPTER 3 22
A REGULAR INTERCONNECT STRUCTURE AND 22
CONFIGURATION STRATEGY 22
3.1 Introduction 22
3.2 Layout Repeatability Solution of the Bus Structure 23
3.3 Fault Tolerant Structure 24
3.3.1 Summary of Thibeault's Implementation 25
3.3.2 Pseudo-5 Connections 30
3.3.3 Comparison and Conclusion of Types of Interconnections 38
3.3.4 Example of a 3 x 3 Physical Interconnection Array Structure 43
3.4 Replacement Analysis of Pseudo-5–source Connection Structure 44
3.4.1 Basic Replacement Operations 45
3.4.2 Notations 45
3.4.3 Tabular Presentation of The Configuration Structure 51
3.4.4 Replacement Solutions by Combination of Operations 53
3.4.5 Serviceability of Configuration Wires 53
3.4.6 Ability and Limitation of Replacements 54
3.4.7 Inter-operation Conflicts 55
3.5 Configuration Rules 56
3.5.1 Receiver Connection Recovering From Simplified Abstraction 56
3.5.2 Connectivity Based Configuration Rules 64
3.6 Reparability Analysis 71
3.6.1 Tolerable Patterns for 1 Column Redundancy 72
3.6.2 General Reparability Analysis 73
3.7 Configuration Algorithms 78
3.7.1 Existing Approaches 78

3.7.2 Investigation of the Approaches for the Pseudo-5-sourse Structure 79
3.7.3 A Greedy Global Mapping Configuration Procedure 82
3.8 Test Diagnosis and Configuration Processes 94
3.8.1 Diagnostic Test 96
3.8.2 Diagnostic Model 97
3.9 Prototype Demo Chip 98
3.9.1 Summary of Specification 98
3.9.2 Design Flow and Design Verification 100
3.9.3 Test 104
3.10 Summary 105
CHAPTER 4 107
DEFECT AND FAULT TOLERANT SCAN CHAIN 107
4.1 Introduction 107
4.2 Overview of IEEE 1149.1 108
4.2.1 Basic Architecture 109
4.2.2 Boundary-scan Chains 110
4.3 Tests and Configuration Structure in One Cell 110
4.4 Fault Tolerant Scan Chain (FTSC) 111
4.4.1 On-wafer Fault Tolerant Scan Chain 112
4.4.2 Off-wafer Fault Tolerant Scan Chain 117
4.5 Proposed Demonstrator Circuit to Verify the On-chip Fault Tolerant Scan Chain 119
4.6 Summary 121
CHAPTER 5 122
CONCLUSION AND FUTURE WORKS 122
5.1 Conclusion 122
5.2 Future works 123
References 125

LIST OF FIGURES

Figure 1.1 (a) WSI processor array (b) One cell in the array..... 2
Figure 1.2 High throughput target architecture — an M x N matrix array 3
Figure 2.1 Cross-section of an interconnection structure with major wire capacitances indicated..... 8
Figure 2.2 (a) Typical on-chip driver-net-receiver connections [12] (b) Equivalent RC model [12] 9
Figure 2.3 Cross section of wiring hierarchy 11
Figure 2.4 Temperature profile along a interconnect line 14
Figure 2.5 Situation expected to generate maximum temperature gradient..... 16
Figure 2.6 Circuit used for the simulation..... 17
Figure 2.7 Delay caused by temperature variations 20
Figure 3.1 Solution of interleaved lines to satisfy layout repeatability constraints in a bi-directional bus 23
Figure 3.2 Two lines per bundle version of the regular bus structure..... 24
Figure 3.3 Horizontal bus transmitter multiplexer connections of Thibeault's implementation..... 26
Figure 3.4 Connections of receiver multiplexers 27
Figure 3.5 One-dimensional bypass 28
Figure 3.6 Inter-bus bundle replacement..... 29
Figure 3.7 A failed turned-path replacement..... 30
Figure 3.8 5-source connections of transmitter multiplexes 31
Figure 3.9 Receiver multiplexer connections of type C connections..... 31
Figure 3.10 Two-dimensional bypassing supported by type C connections..... 32
Figure 3.11 Intra-bus bundle replacement supported by type C connections 33
Figure 3.12 Bundle of a defective cell used by a neighbouring cell for type B connections..... 34
Figure 3.13 Intra-bus bundle replacement supported by type D connections 35

Figure 3.14 Turned path replacement supported by the self-monitoring 36
Figure 3.15 Self-monitoring links of Pseudo-5 receiver connection 37
Figure 3.16 Regular layout version of self-monitoring links 37
Figure 3.17 Example of a 3 x 3 array 44
Figure 3.18 Notation for configuration multiplexers and their inputs 46
Figure 3.19 Diagram (b) is the simplified abstraction of (a) 48
Figure 3.20 Simplified abstractions of cell replacements 49
Figure 3.21 Simplified abstractions of inter-bus bundle replacements 50
Figure 3.22 Simplified abstractions of intra-bus bundle replacements 51
Figure 3.23 Full connections of a configuration 57
Figure 3.24 A simplified abstraction of Figure 3.23 57
Figure 3.25 Flow chart of horizontal bundle transformation 59
Figure 3.26 Example of Pointer m1 60
Figure 3.27 Example of Pointer m2 60
Figure 3.28 Pseudo code of the transformation algorithm 61
Figure 3.28 (continue) Pseudo code of the transformation algorithm 62
Figure 3.28 (continue) Pseudo code of the transformation algorithm 63
Figure 3.29 Cell Replacement Offset 64
Figure 3.30 Transmitter Connectivity rule 65
Figure 3.31 Target cells are from one physical row 66
Figure 3.32 Target cells are from two neighboring rows 66
Figure 3.33 Target cells belong to three neighboring rows 67
Figure 3.34 Constraints in column 67
Figure 3.35 Intra-bus bundle borrowing caused connectivity loss 69
Figure 3.36 Source Exclusive Receiver Rule 70
Figure 3.37 Block Integrity rule 71
Figure 3.38 Several tolerable patterns of 1 column redundancy 72
Figure 3.38 (continue) Several tolerable patterns of 1 column redundancy 73
Figure 3.39 Several intolerable patterns of 1 column redundancy 75

Figure 3.39 (continue) Several intolerable patterns of 1 column redundancy 76
Figure 3.40 Simplest intolerable pattern for 1 column and 1 row redundancy 77
Figure 3.41 (a) Unrepairable cell (b) Unusable cell 77
Figure 3.42 Physical array after specifying redundancy and its corresponding primary logical array. 80
Figure 3.43 Specifying a possible logical array 83
Figure 3.44 Additional possible logical arrays 84
Figure 3.45 Domain of assignments and mapping priority 84
Figure 3.46 Parallel and perpendicular replacement 85
Figure 3.47 Pep and Pap vs. logical index and bundle connection 86
Figure 3.48 Flow chat of the greedy configuration procedure 87
Figure 3.49 Clustered fault pattern repaired by the procedure 88
Figure 3.50 Existing domains and constrained domains 89
Figure 3.51 Data Structure 90
Figure 3.52 Cell Assignment Check (assign P(i,j) to L(k,l)) 90
Figure 3.52 (continue 1) Cell Assignment Check (assign P(i,j) to L(k,l)) 91
Figure 3.52 (continue 2) Cell Assignment Check (assign P(i,j) to L(k,l)) 92
Figure 3.53 Bundle Assignment Check (assign VB(i,j) HB(k,l) to P(m,n) L(o,p)) .. 92
Figure 3.53 (continue) Bundle Assignment Check (assign VB(i,j) HB(k,l) to P(m,n) L(o,p)) 93
Figure 3.54 Test, diagnosis, and configuration procedure 95
Figure 3.55 Configuration circuits partitioning 98
Figure 3.56 Configuration control and test structure 99
Figure 3.57 chained JTAGs 100
Figure 3.58 Demo chip design flow 101
Figure 3.59 Simulation result of logic integrity test 101
Figure 3.60 Simulation result of connectivity tests 102
Figure 3.61 Layout of one cell 103
Figure 3.62 Layout of the demo chip 103

Figure 3.63 Sequence of a test 105
Figure 4.1 JTAG elements of a Boundary-Scan IC 109
Figure 4.2 Simple chain of Boundary-Scan ICs 110
Figure 4.3 Configuration control and test structure 111
Figure 4.4 On wafer fault-tolerant scan chain 112
Figure 4.5 Example to show how multiple faults are tolerated 114
Figure 4.6 Stitching method to generate global scan chain 116
Figure 4.7 Two-line version of the inter-reticle stitching method 117
Figure 4.8 Global scan chain connected on FPGA 118
Figure 4.9 Demonstrator for on-chip fault tolerant JTAG testing 120
Figure A.1 Default configuration 131
Figure A.2 Cell replacement (up shift) R_C_U[(1,0),1] 132
Figure A.3 Cell replacement (down shift) R_C_D[(1,2),1] 133
Figure A.4 Cell replacement (left shift) R_C_L[(0,1),1] 134
Figure A.5 Cell replacement (right shift) R_C_R[(2,1),1] 135
Figure A.6 Cell replacement (left shift) R_C_L[(0,1),2] 136
Figure A.7 Cell replacement (left shift) R_C_L[(0,0),2] 137
Figure A.8 Cell replacement (up Shift) R_C_U[(2,0),2] 138
Figure A.9 Vertical Inter-bus Bundle Replacement (left shift) R_B_V_L[(1,2),1]. 139
Figure A.10 Vertical Inter-bus Bundle Replacement (right shift) R_B_V_R[(2,2),1] 140
Figure A.11 Vertical Inter-bus Bundle Replacement (right shift) R_B_V_R[(2,2),2] 141
Figure A.12 Vertical Inter-bus Bundle Replacement (left shift) R_B_V_L[(0,2),2] 142
Figure A.13 Horizontal Inter-bus Bundle Replacement (up shift) R_B_H_U[(1,1),1] 143
Figure A.14 Horizontal Inter-bus Bundle Replacement (down shift) R_B_H_D[(1,2),1] 144

Figure A.15 Horizontal Inter-bus Bundle Replacement (left shift) R_B_H_U[(1,0),2] 145
Figure A.16 Horizontal Inter-bus Bundle Replacement (down shift) R_B_H_D[(1,2),2] 146
Figure A.17 Vertical Intra-bus Bundle Replacement (down shift) R_B_V_D[(0,2),1] 147
Figure A.18 Vertical Intra-bus Bundle Replacement (down shift) R_B_V_D[(0,2),2] 148
Figure A.19 Vertical Intra-bus Bundle Replacement (up shift) R_B_V_U[(0,0),1] .. 149
Figure A.20 Vertical Intra-bus Bundle Replacement (up shift) R_B_V_U[(0,0),2] .. 150
Figure A.21 Horizontal Intra-bus Bundle Replacement (left shift) R_B_H_L[(0,1),1] 151
Figure A.22 Horizontal Intra-bus Bundle Replacement (left shift) R_B_H_L[(0,1),2] 152
Figure A.23 Horizontal Intra-bus Bundle Replacement (right shift) R_B_H_R[(2,1),1] 153
Figure A.24 Horizontal Intra-bus Bundle Replacement (right shift) R_B_H_R[(2,1),2] 154
Figure A.25 Failed cell replacements with L-turned path 155
Figure A.26 Cell replacements with L-turned path 156
Figure A.27 Cell replacements with two parallel L_turned paths 157
Figure A.28 Two cell replacements with crossed paths 158
Figure A.29 Cell replacement perpendicular with Inter-bus Bundle Replacement ... 159
Figure A.30 Inter-bus Bundle Replacement continue with Intra-bus Bundle Replacement 160
Figure A.31 Intra-bus Bundle Replacement parallel with Inter-bus Bundle Replacement 161
Figure A.32 Cell replacements with U-turned path 162
Figure A.33 Spare cell is the contributor of the spare bundle 163

Figure A.34 Cell replacement parallel with Intra-bus Bundel Replacement 164
Figure A.35 Horizontal Intra-bus Bundle Replacement cross vertical Intra-bus Bundle replacement 165
Figure A.36 Inter-bus Bundle Replacement continue with cell replacement..... 166
Figure A.37 Inter-bus Bundle Replacement parallel with cell replacement 167
Figure A.38 Bundle replcement with with L-turned path 168
Figure A.39 Inter-bus Bundle Replacement combine with Intra-bus Bundle Replacement 169
Figure A.40 Two Inter-bus Bundle Replacements in parallel..... 170
Figure A.41 Diagonal cell replacement (simple) 171
Figure A.42 Diagonal cell replacement (Receiver Source Exlusive Rule violation). 172
Figure A.43 Diagonal cell replacement (no violation)..... 173
Figure A.44 Violation of Receiver Connectivity Rule (Rule.3)..... 174
Figure B.1 The situations to be considered and the corresponding symbols 176

LIST OF TABLES

Table 2.1 Power density estimation of a 3-inverter ring oscillator 16
Table 2.2 Power density estimation of a static NAND 16
Table 2.3 Delay of a stage composed of inverters and a long wire 18
Table 2.4 Calculated skew induced by a 20C° temperature gradient 19
Table 3.1 Bits to control for type A connections 38
Table 3.2 Configuration lines for type A connections 39
Table 3.3 Bits to control for type B connections 39
Table 3.4 Configuration lines for type B connections 40
Table 3.5 Bits to control for type C connections 40
Table 3.6 Configuration lines for type C connections 41
Table 3.7 Bits to control for type D connections 41
Table 3.8 Configuration lines for type D connections 42
Table 3.9 Comparison on configuration connection structures 42
Table 3.10 Physical indexes of the structure 46
Table 3.11 Cell Data Structure 47
Table 3.12 Bundle Data Structure 47
Table 3.13 Special indexes 48
Table 3.14 Cell replacement operations 49
Table 3.15 Inter-bus bundle replacement operations 50
Table 3.16 Intra-bus bundle replacement operations 51
Table 3.17 Tabular presentation of the default configuration 52
Table 3.18 Tabular presentation after a R_C_L[(0,1),1] operation 53
Table 3.19 Serviceability Analysis of All Configuration Wires 54
Table 3.20 Comparison of control and wiring overhead 69
Table 3.21 Vertical domain for row assignment 89
Table 3.22 Horizontal domain for column assignment 90
Table 3.23 Test vectors 96

Table 3.24 A test result example 97
Table 3.25 Wire Capture Locations 104
Table B.1 Inter configuration operation constraints 177

LIST OF ABBREVIATIONS AND SYMBOLS

ABBREVIATIONS

AR Aspect Ratio

BIST Built-In-Self-Test

BSR Boundary Scan Register

CMC Canadian Microelectronics Corporation

CMOS Complementary Metal Oxyde Semiconductor

CR Configuration Register

DPM Dynamic Power Management

FPGA Field Programmable Gate Array

FTSC Fault Tolerant Scan Chain

HDL Hardware Description Language

IC Integrated Circuits

IEEE Institute of Electrical and Electronics Engineers

IPCC Independent Power and Clock Control

ISC In-System Configuration

JTAG Joint Test Action Group

MOSFET Metal Oxide Silicon Field Effect Transistor

nTRST Test Reset

PaP Parallel Placement

PCB Printed Circuit Board

PE Processing Elements

PeP Perpendicular Placement

PODEM Path-Oriented Decision Making

SoC System-on-chip

TAP Test Access Port

TCK Test Clock

TDI Test Data In

TDO Test Data OutTMS Test Mode Select

TR Test Register

TSMC Taiwan Semiconductor Manufacturing Corporation

VHDL Very High Speed Integrated Circuit Hardware Description Language

VLSI Very Large Scale Integration

WSI Wafer Scale Integration

SYMBOLS

R Resistance

C Capacitance

L Inductance

Id Source current

VTH Threshold voltage

Tμ Pseudo empirical coefficients

Tnom Nominal temperature

ρ Unit length resistance at reference temperature

β Temperature coefficient of resistance

CL Capacitance load

C0 Unit length capacitance

$\rho 0$ Unit length resistance at reference temperature (namely 0 °C)

T(x) Temperature profile along the interconnect line

M Number of columns of a target logicalarray

N Number of rows of a target logical array

m Number of redundant columns of a physical array

n Number of rows of a physical array

P(i,j) Physical index

L(i,j) Logical index

LIST OF ANNEXS

Annex A Replacement Examples of A 3 x3 Array.....133
Annex B Inter Configuration Operation Constraints.....178

CHAPTER 1

INTRODUCTION

1.1 Motivation

From the perspective of conventional system design, integrated circuits (IC's) can be regarded as "discrete" devices interconnected according to system design objectives imposed at the circuit board level and higher levels in the system implementation hierarchy. As a natural evolution of conventional device-oriented VLSI chip technologies, the need for functional integration in high-performance systems (such as servers, communications equipment, high-end computers) has led to system-oriented wafer-level technologies.

Wafer scale integration (WSI) has been investigated since the early commercialization of integrated circuits [1,2]. The fact that such a long history of interest in WSI has not led to widespread commercial wafer scale products has suggested that the approach is fatally flawed and of little interest. However, it is more appropriate to regard the failures of WSI as failures in competing with the aggressively expanding silicon VLSI technology which evolved with conventional packaging and system architecture approaches. It was predicted that WSI would emerge as a natural evolution of conventional device-oriented VLSI chip technologies to system-oriented wafer level technologies [3].

As integration technology evolves, new possibilities and challenges arise. With chip-based switch architectures reaching their limits, scalable system-on-chip (SoC) solutions are new approaches to satisfy the continual increasing demand on interconnect bandwidth and circuits density. WSI is identified by Hyperchip Inc [4] as a technology to support its future generation Internet routing products. Other wafer scale SoC applications include

Internet servers, broadband communications and high-end computers. All of these applications have a continuous need for functional integration in high-performance systems. This has led to the use of SoC (in spite of its high cost) to provide a performance needed.

It is well known that yield of circuits without redundancy drops rapidly as the surface area occupied increases beyond a certain level. To relax the limits imposed by yield and cost on IC dimensions, defect avoidance, fault tolerance and testing are major implementation issues for wafer scale integration. For the transition from the VLSI chip systems, WSI systems will require a mapping of the full range of system design issues onto the design of a monolithic circuit. A common strategy is to partition system functions into blocks and to add redundant blocks with which, defect avoidance can be achieved by replacing defective blocks with redundant blocks. As spare sharing is essential to maintain a reasonable ratio of redundancy, WSI is widely regarded as technically feasible for reconfigurable systems with enough regularity. A typical category is regard as processor array as shown in Figure 1.1, for which a large number of processing elements (PEs) incorporate and provide tremendous capacity of computation and bandwidth.

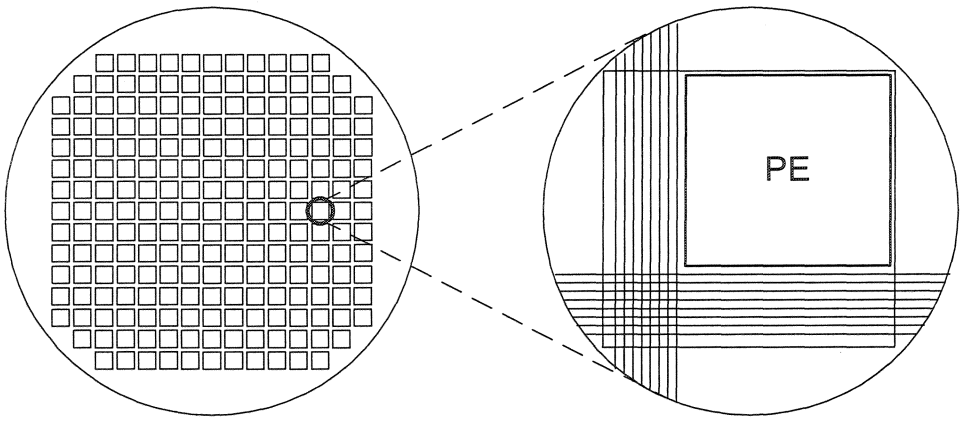


Figure 1.1 (a) WSI processor array (b) One cell in the array

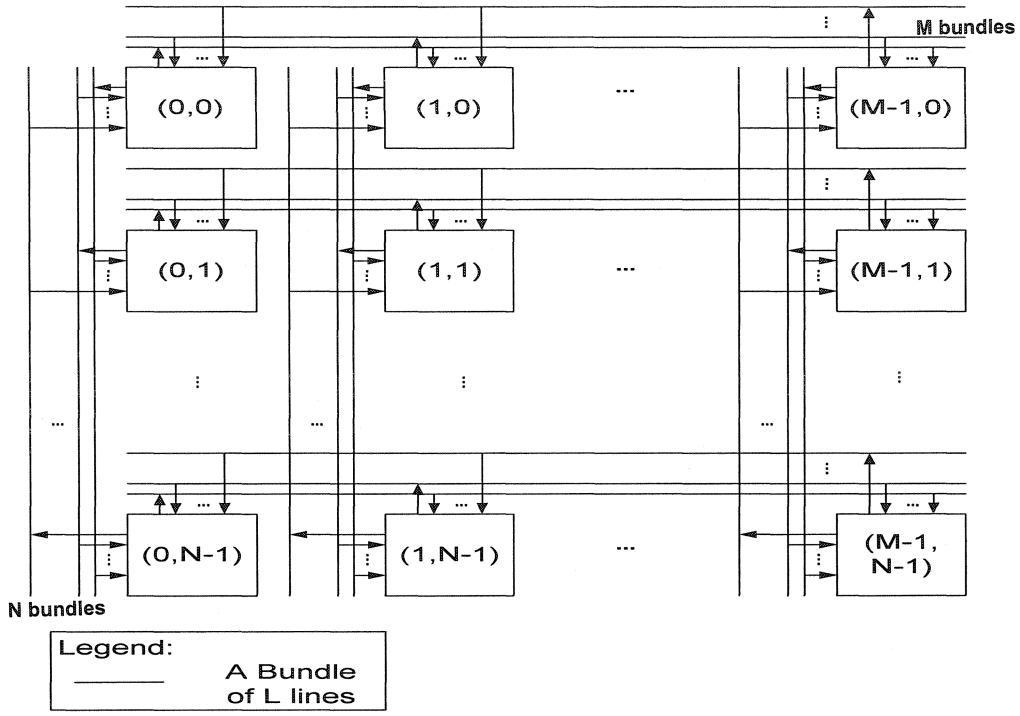


Figure 1.2 High throughput target architecture — an M x N matrix array

The target architecture of interest in this thesis is a processor array applied to Internet switch routers. In the target architecture shown in Figure 1.2, a cell talks to other cells in a row/column through a bundle (L lines) of wires. Assuming that it is an M x N array, each cell can communicate to (M-1) + (N-1) other cells, (M-1) on its row and (N-1) on its column. Each cell talks to a horizontal bundle and a vertical bundle, while other cells in the same row listen to the horizontal bundle and the other cells in the same column listen to the vertical bundle. All links connecting these cells are unidirectional and contain repeaters to regenerate signals.

Technology trends and this unique interconnect structure impose new constraints on WSI system design. In classical scaling theory, the problem of power consumption does not look like that significant, but in actual practice we see an alarming rapid upward trend in power consumption. From one perspective, the main reason is that the scaling down of

power supply voltage does not match the scaling of other factors [5]. It is known that high temperature and unequally dissipated power cause hot and cool spots, which may cause both the decrease of circuit lifetime and more run-time failures. How to sink the heat generated by several kilowatts of power and compensate for the impact of thermal effects are new challenges. The target structure contains very long high dense inter-connections, on which signal integrity is a critical issue. This dense interconnection structure also presents unusual constraints not taken into account in conventional reconfiguration techniques. Appropriate strategies for configuration, diagnosis and test need to be developed.

.

1.2 Objectives of Research

One goal of this thesis is to characterize the skew induced by a temperature gradient. A mathematical model of thermal gradient induced skew will be derived and validated with simulations as well as practical measurements. Maximum thermal gradient induced skew on wafer scale interconnection buses should be estimated. If the skew is critical, a corresponding compensation scheme should be developed.

Another objective is to develop a configuration strategy for the high throughput structure. Several constraints applied by this unique structure and the manufacturability should be taken into account. First, the configuration structure should support the layout regularity of one cell and a group of cells. Secondly, the configuration structure should support enough restructuring flexibility that will directly contribute to yield improvements. Thirdly, the structure should consume adequate amounts of interconnection wires. In this work, the feasibility of configuration algorithms, test, and diagnosis should be investigated.

The last objective of this thesis is to develop configuration control and interconnection testing circuits. JTAG [6] was proposed to conduct configuration control and long

interconnection testing in the early stage of this project. To compensate the yield loss on the global link of JTAGs, a conceptual design of fault tolerance scan chain should be developed.

1.3 Thesis Organization

The thesis consists of five chapters. Following this chapter, chapter 2 describes how skew is induced by temperature gradient. The maximum skew over the long interconnection buses induced by temperature gradient is estimated. Global interconnection strategies are also reviewed.

A fault-tolerant structure dedicated to the high throughput architecture is presented in Chapter 3. The ability of this structure to tolerate defects is analysed. A set of configuration rules is formalized. The configuration algorithm, test and diagnosis on this structure are discussed. A proof of concept demo chip which contains the interconnect structure and JTAG chain is implemented.

In chapter 4, conceptual designs of fault-tolerant scan chains are proposed. One is an on-wafer fault-tolerant scan chain and an alternative is an off-wafer fault-tolerant scan chain. A pattern to generate a global scan chain by regular reticle is presented. Conclusions from this project and future work are presented in the last part of the thesis.

CHAPTER 2

CHARACTERIZATION OF SKEW INDUCED BY TEMPERATURE GRADIENT

The increasing demand for more complex VLSI circuits with higher performance is leading to higher power dissipation and increased thermal problems. Heat generation and parametric variations are rapidly becoming one of the most challenging problems in high-performance chip design due to ever-increasing device counts and clock speeds [7]. At the circuit level, temperature variations in the substrate and interconnect lines have important implications for circuit performance and reliability [8]. In high performance ICs, the chip temperature can reach up to 120° C [9]. Additionally it has been shown that temperature variations can cause a significant change in the interconnect resistance, thereby altering the propagation delay of different paths, and in some extreme cases, cause timing violations [10]. One goal of this thesis is to characterize how the temperature gradient induces skew over very long interconnections and to develop a strategy to compensate for the skew if it is significant.

2.1 Skew over Long Interconnections

In a data bus, electrical signals propagating on different wires take a different amount of time to reach the other end of the bus. The time difference between the "fastest" wire and the "slowest" wire within a bus is defined as the bus's maximum (worst-case) delay skew. Excessive delay skew is problematic because it results in timing problems for signals intended for virtually simultaneous arrival in a computer system.

2.1.1 Propagating Delay Modeling

As has long been predicted, interconnect wiring delays rather than transistor logic delays are the major contributors to global path delays on VLSI circuits fabricated in current deep submicron CMOS processes [11].

2.1.1.1 Parameters Affecting Wire Delay

The three parameters of the wires that determine the characteristics of propagation delays are the resistance, the capacitance and the inductance.

a) Resistance

All wires have resistance, representing the ability of the wire to carry a charge flow. Aluminum wires have a resistivity of 3.3μΩ-cm, while thin-film copper wires have a resistance of 2.2μΩ-cm. Interconnections between metal layers (plugs or vias) for aluminum wires are made of tungsten, and tend to be fairly resistive; in a 0.25-um process, a M1-M2 via resistance can be about 5Ω, while vias from M5 down to the substrate can add to more than 20Ω.

b) Capacitance

All wires constitute capacitances, representing amount of charge that must be added or removed to change a certain electric potential on the wire. Figure 2.1 and equation (2.1) show how a capacitance can be modeled by four parallel-plate capacitors for the top, bottom, right, and left sides, plus a constant term for fringing capacitance. The thickness of the wires in present technologies is usually greater than their width, and will get even thicker to scale down the resistance. At minimum pitch, their side-to-side capacitance is a significant and growing portion of the total.

$$C = C_{LL} + C_{LR} + C_{VO} + C_{VD} + C_{fringing} \qquad (2.1)$$

where, C_{LL} , C_{LR}, C_{VO} and C_{VD} are defined in Figure 2.1.

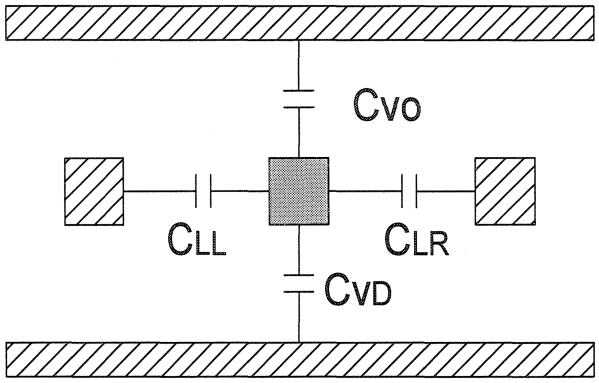


Figure 2.1 Cross-section of an interconnection structure with major wire capacitances indicated

c) Inductance

No handy closed form model exists for on-chip wire inductance, as they do for resistance or capacitance. We usually consider the inductance with respect to a loop, and a changing magnetic flux through it induced by a current charge in the loop. The technology scaling trends not only result in a significant increase in conductor resistance, but also give rise to a moderate increase in inductance (at low frequency), for the case of a wide bus above a plane.

2.1.1.2 RC Model

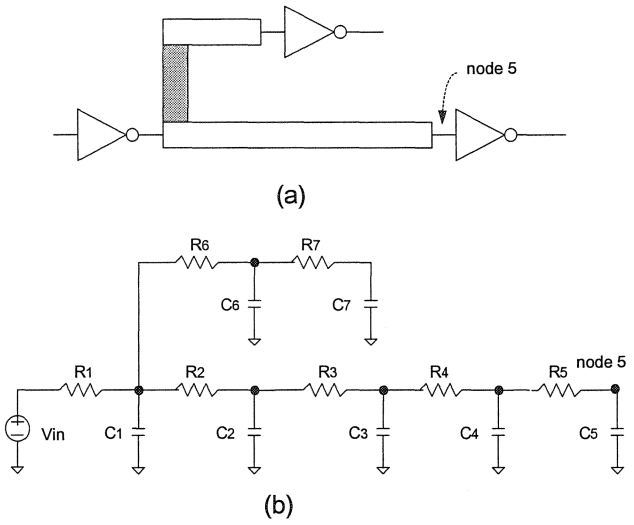


Figure 2.2 (a) Typical on-chip driver-net-receiver connections [12]
(b) Equivalent RC model [12]

For the circuit of Figure 2.2, where the driver is modeled by a resistor driven by a voltage source, the Elmore delay [13] at node 5 can be written as follows:

$$T_{d5} = \sum_{i \in P(5)} R_i C_{di}$$
$$= R_1(C_1 + C_2 + C_3 + C_4 + C_5 + C_6 + C_7) + R_2(C_2 + C_3 + C_4 + C_5)$$
$$+ R_3(C_3 + C_4 + C_5) + R_4(C_4 + C_5) + R_5 C_5 \qquad (2.2)$$

In the above equation, P(5) represents the path from node 5 to the root of the tree, while C_{di} represents the sum of the capacitors that are downstream of each resistor R_i that lies along this path.

2.1.1.3 RLC Model

As edge rates continue to improve, we arrive at un unfortunate situation in which the lumped RC modeling of the past becomes inaccurate, and complicated RLC(f) matrices and lossy transmission line models become necessary in same cases. Ismail and Friedman's analysis shows that RC models can create errors of up to 30% in the total propagation delay of a repeater system as compared to the optimal delay if inductance is considered [14].

2.1.2 Interconnection Scaling Trend and Global Wire Optimization

CMOS technology evolution in the past twenty years has followed the path of device scaling for achieving density, speed, and power improvements. MOSFET transconductance per device width increases with shorter channel length, that is, with reduced source-to-drain spacing. The intrinsic capacitance of a short-channel MOSFET is also lower, which makes it easier to switch. Interconnect capacitance and resistance have negligible effects on the delay of local circuits. RC delay of local wires will not limit circuit speed significantly even though it cannot be reduced through scaling.

The increase of the line resistance is one of the main reasons for the increased wiring delay. Due to the rising need for higher densities on-chip, wiring pitches are dropping rapidly at about the same rate as gate length. As resistance is inversely proportional to the cross-sectional area of the wire, one of the methods to reduce resistance is to slowly scale line thickness, resulting in taller, narrower wires. These high aspect ratio (AR) lines have a detrimental side effect in that they result in a large amount of coupling capacitance. With AR>1, lines tend to have more capacitance to their neighboring wires than to upper and lower wiring layers. In addition, spacing between wires is shrinking quickly in an attempt to maintain high packing densities, further increasing coupling capacitance. For

instance, line-to-line capacitance between wires on the same level makes up over 70% of the total wiring capacitance at lower levels even at 0.25-um processes [15].

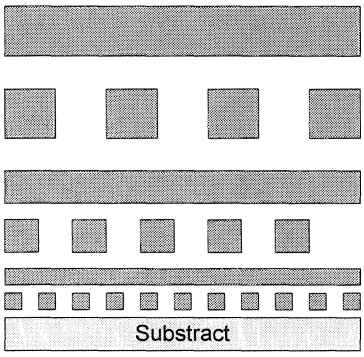


Figure 2.3 Cross section of wiring hierarchy

As shown in Figure 2.3, the general trend is to use fine wire pitch and thin wires in the lower wire levels, to meet integration density requirements and to minimize capacitance, respectively, in combination with wide wire pitch and thick wires in the upper wire levels , to maximize speed [16].

Global nets tend to be very long and, therefore, exhibit delays that may span several clock cycles. Furthermore, these nets are usually routed on the upper 'low resistance' metal layers of the chip, which tend to have high relative coupling. One can also use repeaters to reduce the dependence of RC delay on wire length from a quadratic relationship to a linear one [17].

2.1.3 Skew over Very Long Buses

Skew over long buses may be induced by parameter variation and electromagnetic interference.

2.1.3.1 Parameter Variation

Parameter variations are deviations from intended or designed values for a structure or circuit parameter concerned. The electrical performance of microprocessors or other integrated circuits are impacted by two sources of variation. Environmental factors arise during the operation of a circuit, and include variations in power supply, switching activity, and temperature of the chip or across the chip. Physical factors during manufacture result in structural device and interconnect variations that are essentially permanent. These variations arise due to processing and masking limitations. This thesis focuses on the skew induced by temperature gradients.

2.1.3.2 Electromagnetic Interference

When subject to electromagnetic interference, the delay induced by a signal propagating on a long wire changes as a function of the electrical activity in the wires environment. For instance, simultaneous switching of signals on neighbor wires can accelerate or slow down signal propagations. Inductance effects can also induce delay variations. The signal propagating in one line is affected by the current direction and variation of its neighboring lines. The magnitude of the inductively coupled noise is strongly dependent on the distribution of capacitive load and the length of the interconnections.

2.2 Mathematical Analysis of Skew Induced by a Temperature Gradient

Temperature variations over a chip can result in different buffer speeds and different resistances of the wires.

The temperature's effect on inverter speed can be observed from the temperature's effect on source current I_d and threshold voltage V_{TH}. A good submicron current model is presented in [18]:

$$I_d = \frac{\mu_0}{R} C_{ox} \frac{W}{L}\left[V_{GS} - V_{T0} - \frac{1+\delta}{2} V_{DS}\right] V_{DS} \qquad V_{GS} > V_{T0} \quad V_{DS} < V_{GS} - V_{T0} \qquad (2.3)$$

Where, $\mu 0$ represents carrier mobility, R represents reduction factor, C_{ox} represents oxide capacitance, W and L represent the channel width and length of the transistor respectively, V_{GS} represents gate-to-source voltage, V_{T0} represents threshold voltage for source-to-substrate voltage equal to 0, δ represents non-uniform distribution of charge along the channel, V_{DS} represents drain-to-source voltage.

Temperature sensitivity can be included in considering mobility and threshold voltage variation such as [19][20]:

$$\mu_0(T) = \mu_0(T_{nom})\left(\frac{T_{nom}}{T}\right)^{T^{\mu}} \qquad (2.4)$$

$$V_{T0} = V_{T0}(T_{nom}) - \delta(T - T_{nom}) \qquad (2.5)$$

where, T_{nom} is approximately 300°K, δ represents pseudo empirical coefficients, which is related to the lattice and impurity scattering effects, and the impact of temperature variations on the intrinsic carrier concentration respectively. δ must be calibrated on the various processes and have values averaged at $-1.72 \cdot 10^{-3} V / ^{o} C$ for p-channel devices and $1.52 \cdot 10^{-3} V / ^{o} C$ for n-channel devices.

It is shown in (2.6) [21] that the resistance of an interconnect increases linearly with the temperature.

$$r = \rho (1 + \beta \cdot \Delta t) \qquad (2.6)$$

where, r is the unit length resistance and ρ is the unit length resistance at reference temperature and β is its temperature coefficient. It is also assumed that the unit length capacitance dose not change with temperature variations along the interconnect length.

Consider an interconnection with length L and uniform width W that is driven by a driver of output resistance Rd and terminated at a load with capacitance C_L. The distributed RC Elmore delay can be written as follows [21]:

$$D = D_0 + (c_0 L + C_L)\rho_0 \beta \int_0^L T(x)dx - c_0 \rho_0 \beta \int_0^L x.T(x)dx \qquad (2.7)$$

where:

$$D_0 = R_d (C_L + c_0 L) + \left(c_0 \rho_0 \frac{L^2}{2} + \rho_0 L C_L \right)$$

C_0 is the unit length capacitance, ρ_0 is the unit length resistance at reference temperature (namely 0 °C), β is the temperature coefficient of resistance (1/°C) and D_0 is the Elmore delay of the interconnect corresponding to the unit length resistance at 0 °C. $T(x)$ is the temperature profile along a interconnect line, as shown in Figure 2.4.

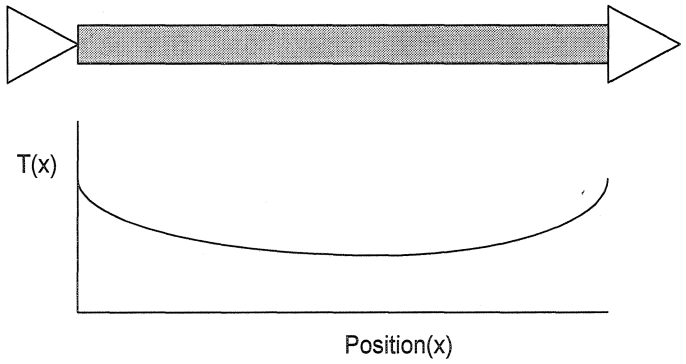


Figure 2.4 Temperature profile along a interconnect line

Assuming that temperature is constant along one line, the skew induced by a temperature gradient, ΔT, between this one and another line can be derived from (2.7) as:

$$\Delta D = \rho_0 \beta \left(C_L L + C_0 \frac{L^2}{2} \right) \Delta T \qquad (2.8)$$

For certain circuits, $\rho_0 \beta \left(C_L L + C_0 \frac{L^2}{2} \right)$ is considered fixed. In such a case, it is obvious that temperature gradient induces linear skew.

2.3 Temperature Gradient Estimation

It has been shown that temperature gradients on silicon substrates can occur due to different activities and/or different sleep modes of various functional blocks in high-performance microprocessor chips [22]. Dynamic power management (DPM)[23] and functional-block clock gating can be major sources of such thermal gradients over the substrate.

Let us assume that a wafer is composed of 20 x 20 cells that are placed on a pitch of 4mm. Each cell is with a respective area of 4mm x 4mm. The biggest temperature gradient may be generated when a group of hottest cells meet with a group of coolest cells, as shown in Figure 2.5.

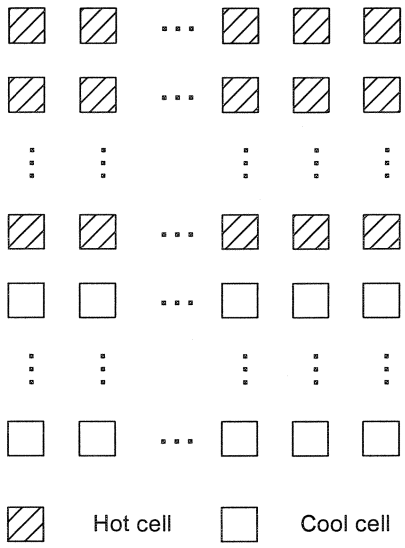


Figure 2.5 Situation expected to generate maximum temperature gradient

One of the most active circuits we can image is a 3-inverter ring oscillator. As shown in Table 2.1, we estimate its power density is 6.32 $\mu W / \mu M^2$.

Table 2.1 Power density estimation of a 3-inverter ring oscillator

Average power consumption	861.4 μW
Peak power consumption	1312 μW
Surface area	136.32 μM^2
Average power density	6.32 $\mu W / \mu M^2$
Peak power density	9.62 $\mu W / \mu M^2$

The coolest cell power density is simulated as a NAND gate with fixed input and floating output, which only consumes static power.

Table 2.2 Power density estimation of a static NAND

Average power consumption	16.4 pW
Surface area	47.57 μM^2
Average power density	0.345 $pW / \mu M^2$

Thermal simulations [24] done by Lakhsasi and Bougataya show that the maximum temperature gradient generated by the above hottest cells and coolest cells is 37 °C over 8 mm. Assuming there are 20 buses in a 4 mm wide channel, the temperature gradient over one bus is approximately 1°C.

2.4 Simulations and Analysis

To characterize skew induced by a temperature gradient, Spectre simulations are conducted.

2.4.1 Simulated Circuit

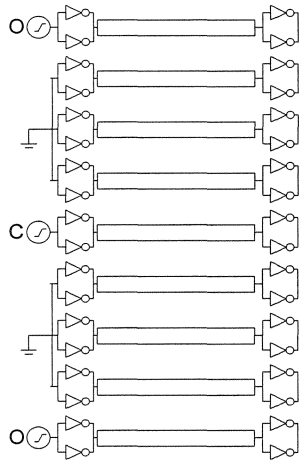


Figure 2.6 Circuit used for the simulation

The simulated circuit shown in Figure 2.6 is a segment of the bus structure in one of our WSI demonstrator chips. This chip was designed and manufactured using a CMOS 0.18 μm technology, in which the features of WSI long interconnections are taken into consideration. The Spectre model is extracted from the layout of the chip. This model corresponds to 9 wires laid out in parallel with pitches of 3 μm on the metal 5 layer. The dimension of the wires is 0.76 μm (Width) x 3750 μm (Length). Each long wire is driven by 2 inverters in parallel. The two outmost wires and the center wire are selected to propagate the signal. The other wires are grounded in order to minimize the effect of electromagnetic interferences.

2.4.2 Parameters of the Simulation

Simulations with 0 wiring resistance, fixed wiring resistance (with 0 temperature coefficients) and temperature-varied wiring resistance are carried out in order to see the effect of resistance variation. The 0-resistance load of the long interconnections is extracted from the layout without its resistor layer. The temperature-varied resistance load of the long interconnections is extracted from the layout with resistor layers. The fixed resistance load of the long interconnections is created from a Spectre netlist by fixing the temperature coefficients to 0. Delay over one stage of inverter on the outmost wires and the center wire is simulated in order to see the effect of the difference of coupling parasitic capacitance. The input signal has a fixed rising and falling time of 65ps. Temperature increments form 5 °C to 125 °C by a step of 20 °C.

2.4.3 Simulation Results and Analysis

The delay of a stage composed of two inverters and a long wire for various cases of interest is reported in Table 2.3. In this table:

n is for the cases without extracted resistance of the long wire,

f is for the cases with extracted resistance where the temperature coefficient is 0,

w is for the cases with the extracted resistance of the long wire,

O is for the outmost wire of the bus,

C is for the center wire of the bus,

R is with rising edge as input signal,

F is with falling edge as input signal.

nOF, for example, is the delay of a outmost line with a falling edge of a signal as input without considering wiring resistance .

Table 2.3 Delay of a stage composed of inverters and a long wire

Temp (°C)	nOF (ps)	nCF (ps)	fOF (ps)	fCF (ps)	wOF (ps)	wCF (ps)	nOR (ps)	nCR (ps)	fOR (ps)	fCR (ps)	wOR (ps)	wCR (ps)
5	36.91	37.75	62.94	64.89	60.47	62.34	20.14	20.73	40.30	42.03	38.25	39.80
25	37.27	38.10	63.38	65.33	63.00	65.02	20.56	21.14	40.71	42.41	40.36	42.08
45	37.63	38.39	63.69	65.67	65.80	67.79	20.98	21.58	41.31	43.00	42.80	44.58
65	37.88	38.74	64.21	66.14	68.30	70.50	21.44	22.02	41.79	43.57	45.03	47.00
85	38.14	38.96	64.52	66.46	70.95	73.14	21.93	22.51	42.39	44.17	47.31	49.36
105	38.37	39.25	64.82	66.74	73.50	75.75	22.39	22.99	42.87	44.63	49.68	51.77
125	38.62	39.53	65.07	66.91	75.95	78.44	22.85	23.47	43.56	45.29	51.93	54.22

The delay on the center wire is a little bigger than the delay on the outmost wires due to a stronger coupling capacitance between the wire and its neighbors. The skew between neighboring wires can be conveniently ignored at this point, since it can almost be canceled by some techniques [25] developed by this team. The temperature gradient induced skew of interest in this thesis is the skew between the two outmost wires under a temperature gradient. Since the situations (the drivers and the wiring loads) of the two outmost wire are the same, the delay difference caused by a temperature difference on one of the outmost wires is equivalent to the skew observed on the two wires due to a temperature gradient of the same magnitude as the temperature difference. The equivalent skew on the two outmost wires induced by 20°C temperature gradient, i.e. the calculated delay difference caused by 20°C temperature increase from Table 2.3 is shown in Table 2.4. The average skew and the relative variation compared to the total delay are shown in the last two rows. Figure 2.7 graphically shows the results of the outmost wires in Table 2.3.

Table 2.4 Calculated skew induced by a 20C° temperature gradient

Temp (°C)	nOF (ps)	fOF (ps)	wOF (ps)	nOR (ps)	fOR (ps)	wOR (ps)
5	0.36	0.44	2.53	0.42	0.41	2.11
25	0.36	0.31	2.80	0.42	0.60	2.44
45	0.25	0.52	2.50	0.46	0.47	2.23
65	0.26	0.32	2.66	0.49	0.61	2.28
85	0.23	0.29	2.55	0.46	0.47	2.37
105	0.25	0.25	2.44	0.46	0.69	2.25
Aver.	0.28	0.35	2.58	0.45	0.54	2.28
Rel(%)	0.77	0.56	4.27	2.24	1.35	5.96

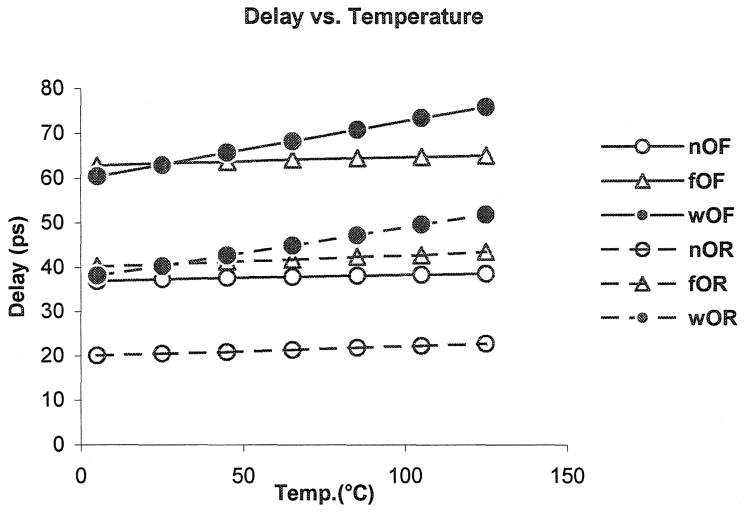


Figure 2.7 Delay caused by temperature variations

Our simulation results show that the skew induced by a temperature gradient is approximately linear with its magnitude. 70% of the skew induced by a temperature gradient is contributed by the resistance variation of the interconnections. Thus, the resistance variation is the dominant factor.

2.4.4 Skew Estimation on Wafer Scale Interconnections

From each inverter stage, the maximum skew induced by an overall 20 °C temperature gradient is 2.8 ps. Since the skew induced by a temperature gradient is approximately linear with the magnitude of the gradient, if the temperature difference over one bus is about 1 °C, we thus expect that a maximum of 2.8 / 20 = 0.14 ps skew could be introduced on a bus segment. For a 20 cm long bus, the maximum skew induced by a 1°C temperature gradient should be no more than 20 / 0.37 * 0.14 = 7.6 ps.

We conclude that the temperature gradient induced skew is very small for local interconnections as well as for global buses, which is negligible; however, it may be a significant factor for clock distribution networks. Due to the high current driven through

the clock wires, clock nets usually exhibit the highest Joule heating among signal nets, and since they span a large area over the die, the probability that they will experience some thermal gradients is much higher than for shorter signal nets [21].

2.5 Demo Chip to Characterize Temperature Gradient Induced Skew

A demo chip with heater cells, temperature sensor cells and skew measurement circuits has been designed by Fecteau. This chip could be used to perform on-chip characterization of temperature gradient and how it induces skew. No experimental results are available at the time of completing this thesis

2.6 Summary

The wiring resistance and lateral coupling capacitance are main contributors to RC delay on long interconnections for deep submicron CMOS processes. There is approximately linear skew induced by a temperature gradient on long buses. 70% of the skew induced by a temperature gradient is contributed by resistance variations of the interconnections, which is the dominant phenomenon. Temperature gradient induced skew is negligible for local interconnections as well as for global buses; however, it may be a significant factor for clock distribution networks.

CHAPTER 3

A REGULAR INTERCONNECT STRUCTURE AND CONFIGURATION STRATEGY

3.1 Introduction

Recent progress in computer technology has enabled the construction of massively parallel computers with large numbers of processing elements (PEs), interconnected with high-throughput networks. Large PE arrays must provide fault tolerance in order to maintain the dependability of systems in use and to attain a high yield (defect tolerance) of integrated microelectronic systems [26]. Faults (defects) occur not only in PEs, but also in interconnection links. Faults in the connections between PEs and links sometimes affect both of them [28]. The grade of fault tolerance where faulty PEs and faulty links can be tolerated in a static manner is called strong fault tolerance [29]. Several strong fault-tolerance schemes using switching networks [29][30], replacement circuits (directly connected types) [31], or graph-based bypass connections [32] have been proposed for reconfiguring complete primary arrays.

The rich interconnection structure shown in Figure 1.1 imposes new constraints. Via talker owned long interconnections, a cell talks to a group of cells in a row and in a column, rather than only to its neighbouring cells. In addition, very long high-speed interconnections occupy an area comparable to that occupied by logical cells. The defects in long interconnections should be tolerated as well as those in logical cells. There is no currently available fault-tolerance strategy that has addressed this type of sophisticated structure.

Both cost and technology limit the size of reticles used to produce integrated circuits. Wafers are manufactured by stepping the reticle to cover them. WSI manufacturing methods (such as Inter-reticle Stitching), design cost and fault tolerance all demand the regular layout of logical cells and interconnections.

Based on some earlier work, a comprehensive interconnection scheme with a strong fault tolerant structure and a regular layout is proposed in this thesis.

3.2 Layout Repeatability Solution of the Bus Structure

A bi-directional bus structure is disclosed in [42]. As shown in Figure 3.1, it uses an solution of interleaved lines to satisfy the layout repeatability constraint. The number of lines is doubled to produce a regular bus layout in each cell. Half of the buffers are inactive (do not propagate a signal) as shown in bold lines. Specified levels (logical '1' or '0') should be applied to the inputs (circles at the far end of structures in Figure 3.1 and Figure 3.2) of each repeater located at the beginning of an inactive line in order to stabilize inactive interconnections.

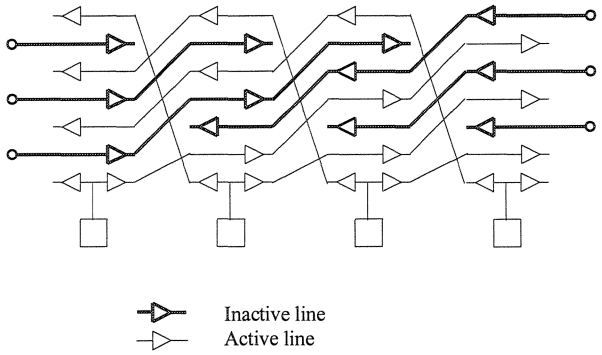


Figure 3.1 Solution of interleaved lines to satisfy layout repeatability constraints in a bi-directional bus

One contribution of this thesis is to prove that this scheme with interleaved lines is a generic one. In Figure 3.2, the structure of Figure 3.1 is developed to the case where each bundle contains 2 lines. It is obvious that this structure can be easily generalized to the cases where one bundle contains more than 2 lines.

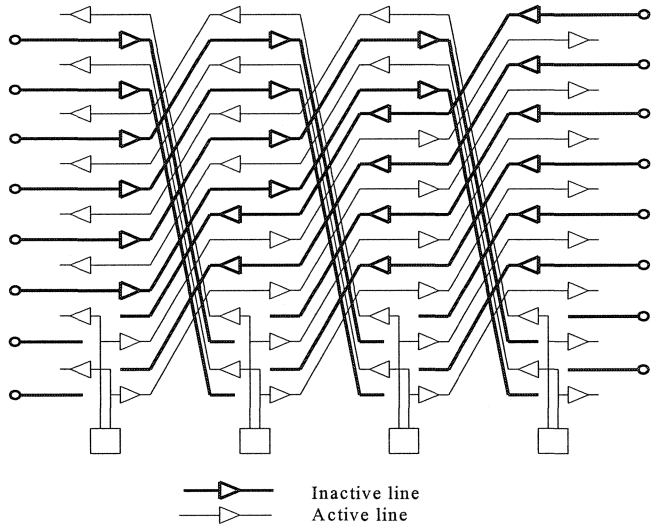


Figure 3.2 Two lines per bundle version of the regular bus structure

3.3 Fault Tolerant Structure

Many configuration schemes aim to bring good elements together from a physical array with redundancy to build a fully functioning and connected logical target array. The desired array architecture for an intended application is called the logical array, and the indexes used to represent the location of the PEs are called logical indexes [27]. The logical array is indexed from [0,0] to [M-1, N-1] for the M x N logical array shown in Figure 1.2. The physical array is a reconfigurable array, which has the ability to tolerate manufacturing defects. It usually consists of spare PEs and a switching network for

restructuring [27]. The indexes used to represent the location of PEs, including spares on the physical array, are called physical indexes [27]. The physical array is indexed from [0,0] to [M-1+*m*, N-1+*n*] for an (M + m) x (N + n) physical array, where *m* is the number of spare rows, and *n* is the number of spare columns. The logical array can be derived from the healthy PE's on the physical array if reconfiguration is possible [27].

Thibeault proposed an implementation [40] of Norman's direct replacement fault-tolerance concept [41] two years ago. The fault tolerant structure proposed in this thesis is a refined structure based on Thibeault's implementation.

3.3.1 Summary of Thibeault's Implementation

The architecture of interest is composed of bundles, which represent groups of lines and processing elements that perform a function of interest.

Replaceable Units

Bundles as well as processing elements (PEs) are defined as replaceable units.

Number of Transmitters and Receivers

For an (M+*m*) x (N+*n*) physical array, each cell contains M+*m*-1 receivers and one emitter for its row, as well as N+*n*-1 receivers and one emitter for its column.

Connection of Transmitter Multiplexers

Figure 3.3 shows the horizontal bus transmitter connections of a 3 x 3 array. The bold lines represent global data buses and the other interconnect lines generate bypasses. The transmitter multiplexers are connected to 3 sources (the emitter of its own cell and the emitters of its upper and lower neighbouring cells). Consequently, each bundle can be connected to its natural owner PE or borrowed by any one of its upper and lower neighbouring PEs. For example, cell C11 can talk to bundle HB11 via connection r11o,

to bundle HB10 (the same ranking bundle in upper neighbouring bus H0) via connection r10d or to bundle HB12 (the same ranking bundle in lower neighbouring bus H2) via connection r12u. Similarly, the inputs of transmitter multiplexers of vertical buses are connected to its natural transmitter and the transmitters of its left and right neighbouring cells.

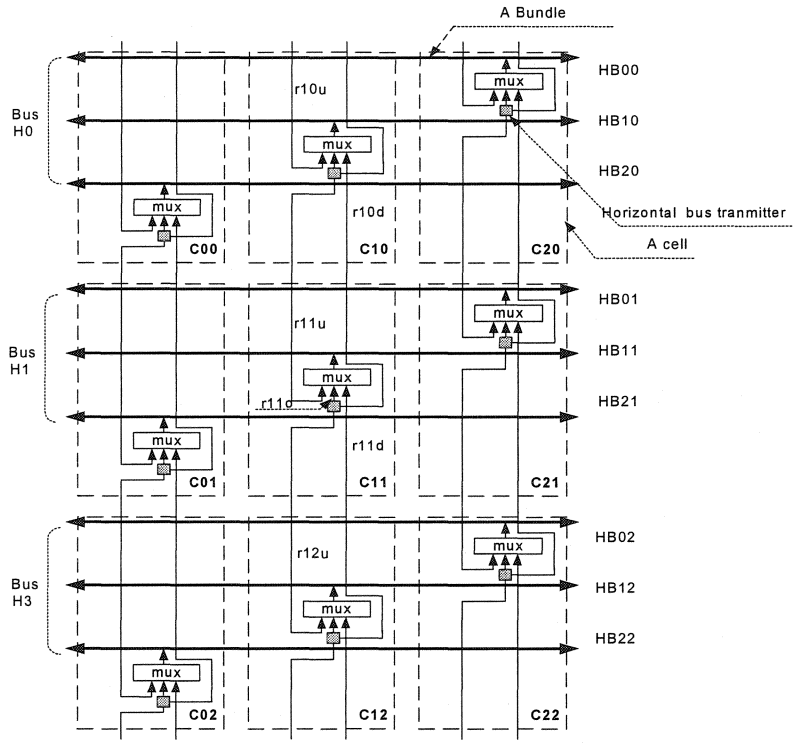


Figure 3.3 Horizontal bus transmitter multiplexer connections of Thibeault's implementation

Connections of Receiver Multiplexers

Receiving multiplexers are connected to 3 sources (the bundle of the receiver, the two bundles with the same rank of the neighbouring buses). Figure 3.4 shows the receiving connections of horizontal buses of a 3x3 array, as an example. A receiver can receive data from its naturally owned bundle or the bundles with the same rank of the neighbouring buses. Similarly, the receiving multiplexers of vertical buses receive from their natural bus and their left and right neighbouring buses.

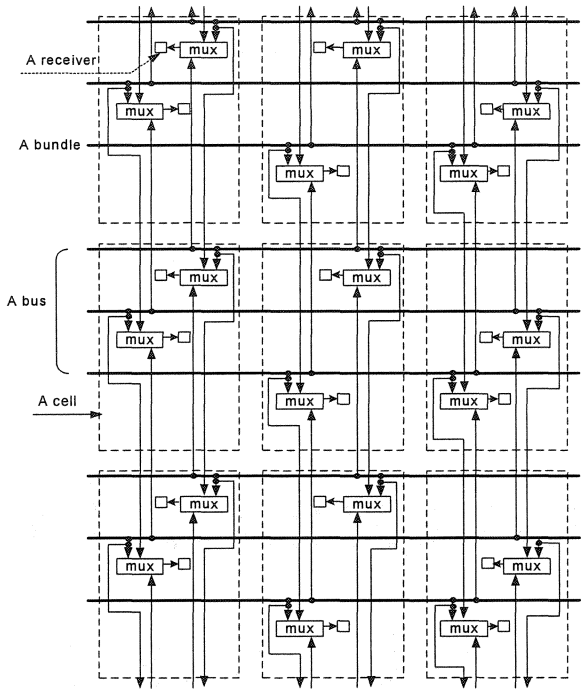


Figure 3.4 Connections of receiver multiplexers

The transmitting connections (as shown in Figure 3.3) and the receiving connections (as shown in Figure 3.4) coexist in a cell. The reason to separate them into two figures is just to simplify the drawings.

Control of Multiplexers

The configuration multiplexers in each cell are assumed to be controlled by a data register of a JTAG (IEEE 1149.1) controller.

Supported Fault Tolerance and Limitations

Each cell not on the edge of the array can be replaced by one of its eight neighbors. In Figure 3.5, the long bold lines are global buses, the short bold lines are active taps to global buses, the thin lines are the primary bypass links for each cell to talk to or listen to its naturally owned buses. The inactive alternate bypass links, which are not currently selected as an active connection, and the links that enable a cell to talk to another cell's

transmitting multiplexer are not shown for simplicity. The active alternate bypass links, which are currently selected as the active bypass connection are drawn as bold dotted lines. As shown by the bold dotted lines, Cell S replaces cell F by redirecting the cell S's vertical bus connections from bus V0 (by default, cell S is connected to V0) to bus V1 (by default, cell F is connected toV1). After the replacement, cell S can communicate with cell C10 and cell C12 with vertical bus V1 and communicate with cell C21 still with horizontal bus H1.

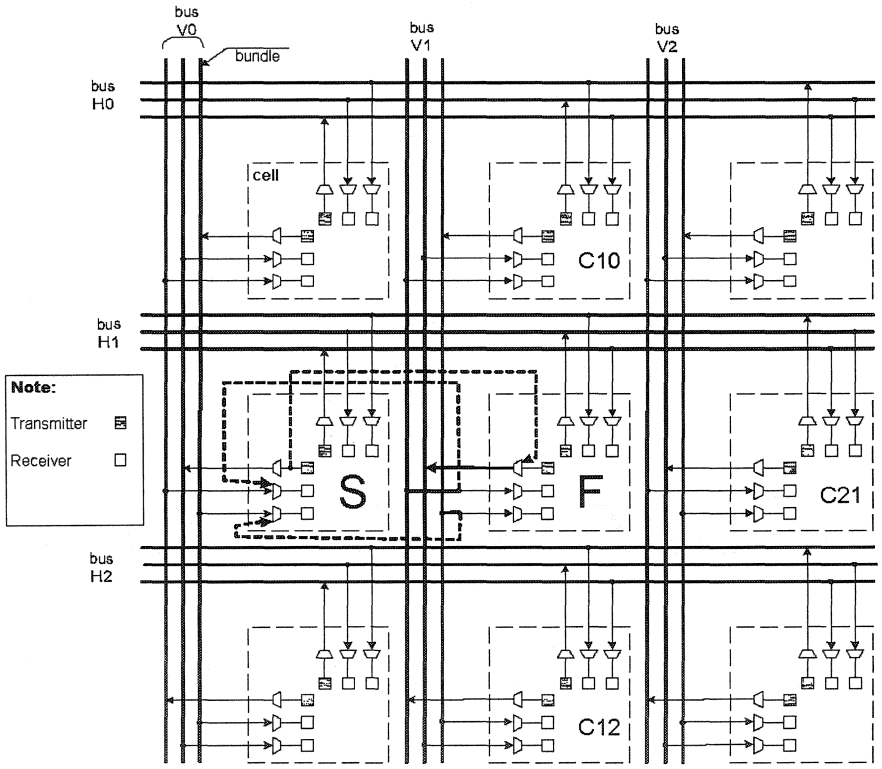


Figure 3.5 One-dimensional bypass

Each bundle can be replaced with one of the bundles with the same ranks in the two neighboring buses. The rank of a bundle indexes the position of the bundle in a data bus. In Figure 3.6, bundle HB00 is replaced by bundle HB01. After replacement, the horizontal transmitter of cell C00 communicates with bundle HB01 owned by cell C01 via connection s00, cell C10 receives from bundle HB01 via connection s102 and cell C20 receives from bundle HB01 via connection s201.

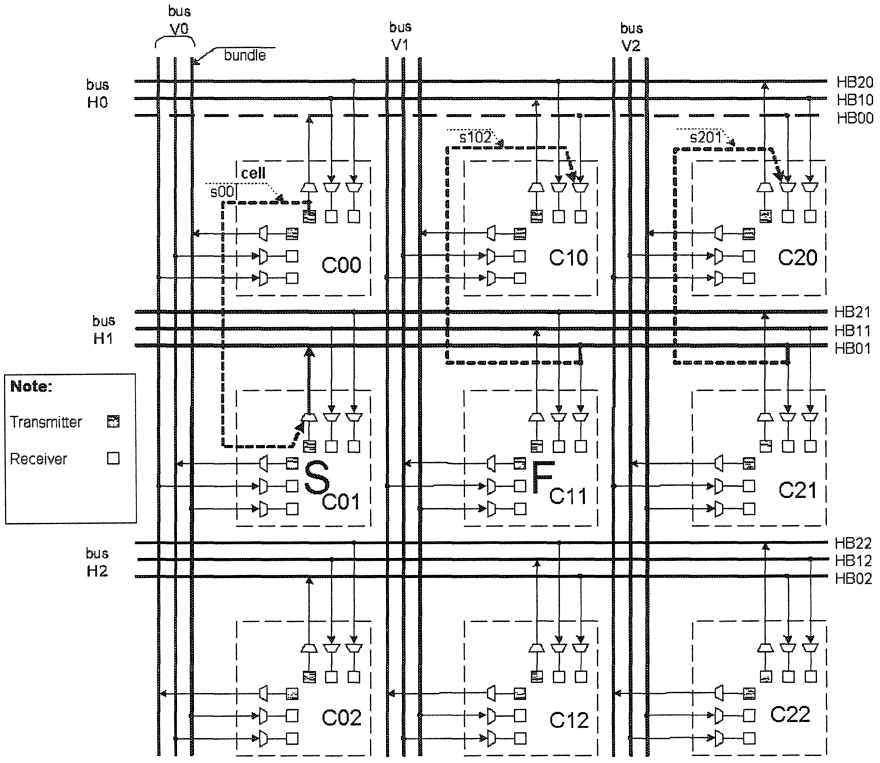


Figure 3.6 Inter-bus bundle replacement

There are two significant limitations of this structure. The first is that it does not support turned-path cell replacement. In Figure 3.7, we intend to replace cell F with cell S. Cell C11 can replace cell F by connecting its horizontal bus connections to bus H0 and cell S can replace cell C11 by connecting its vertical connections to bus V1. After the two operations, cell S and cell C11 are supposed to be able to communicate with each other over vertical bus interconnections. However, we cannot make these connections since there are no available wires. As a consequence, the turned-path replacement cannot succeed. Without turned path cell replacement, large clustered defect pattern cannot be tolerated. The second limitation is that bundle replacements are restricted to the bundles with the same rank in the neighboring buses. Thus spare bundles within the same bus cannot replace a neighbor bundle.

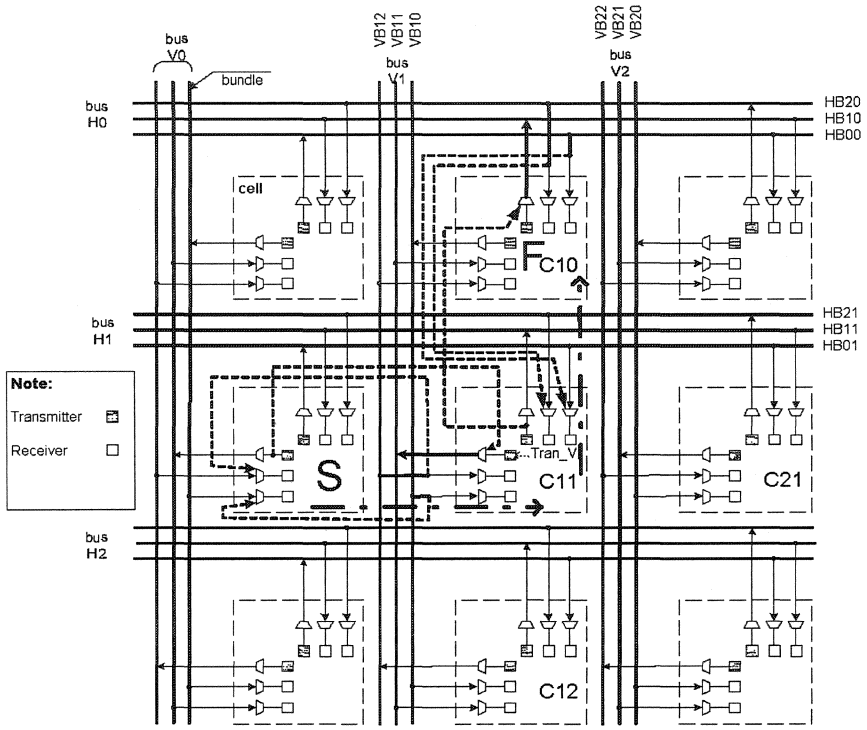


Figure 3.7 A failed turned-path replacement

3.3.2 Pseudo-5 Connections

Explorations

There is a trade off between the number of configuration connections and the flexibility of the configuration method. Finding out an effective configuration is the objective of this exploration. 3 types (A, B and C) of interconnections will be compared first.

Thibeault's implementation, for which both transmitters and receivers are connected to 3 sources, is named type A connections. For type B connections, the receiving multiplexers are connected to 3 sources that are the same as type A, while the transmitting multiplexers are connected to 5 sources (its naturally associated transmitter and the transmitters of its 4 neighbors) as shown in Figure 3.8. For type C connections, both transmitting and receiving multiplexers are connected to 5 sources (naturally associated

transmitter/receiver and the transmitters/receivers of its 4 neighbors). The receiver multiplexer connections of type C connections are shown in Figure 3.9.

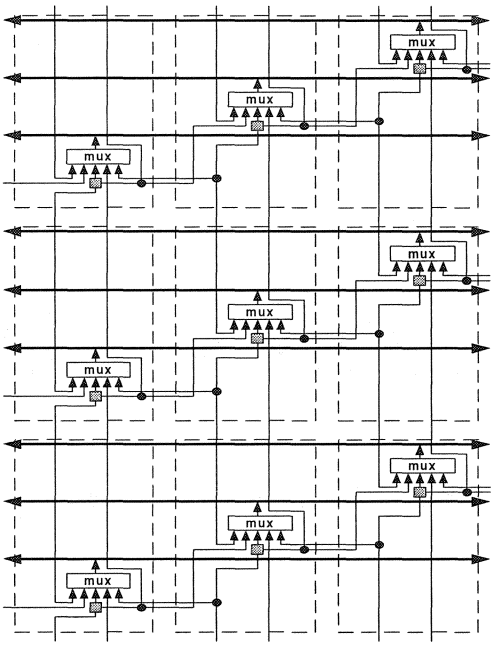


Figure 3.8 5-source connections of transmitter multiplexers

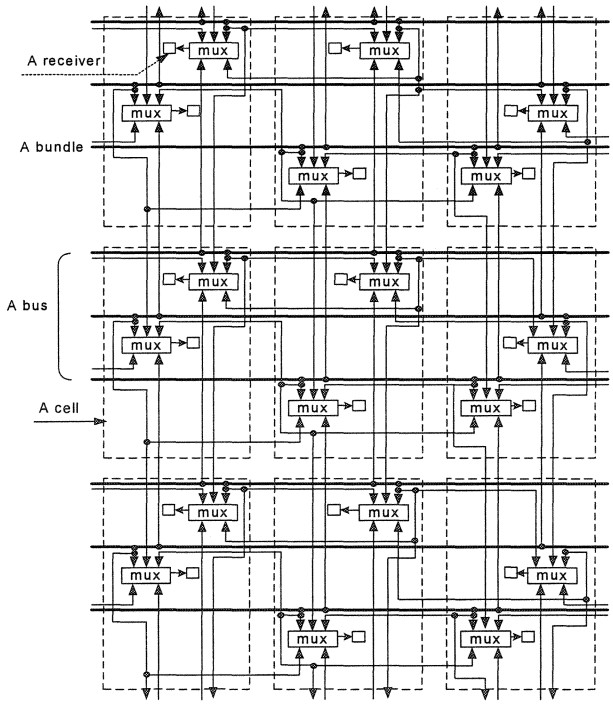


Figure 3.9 Receiver multiplexer connections of type C connections

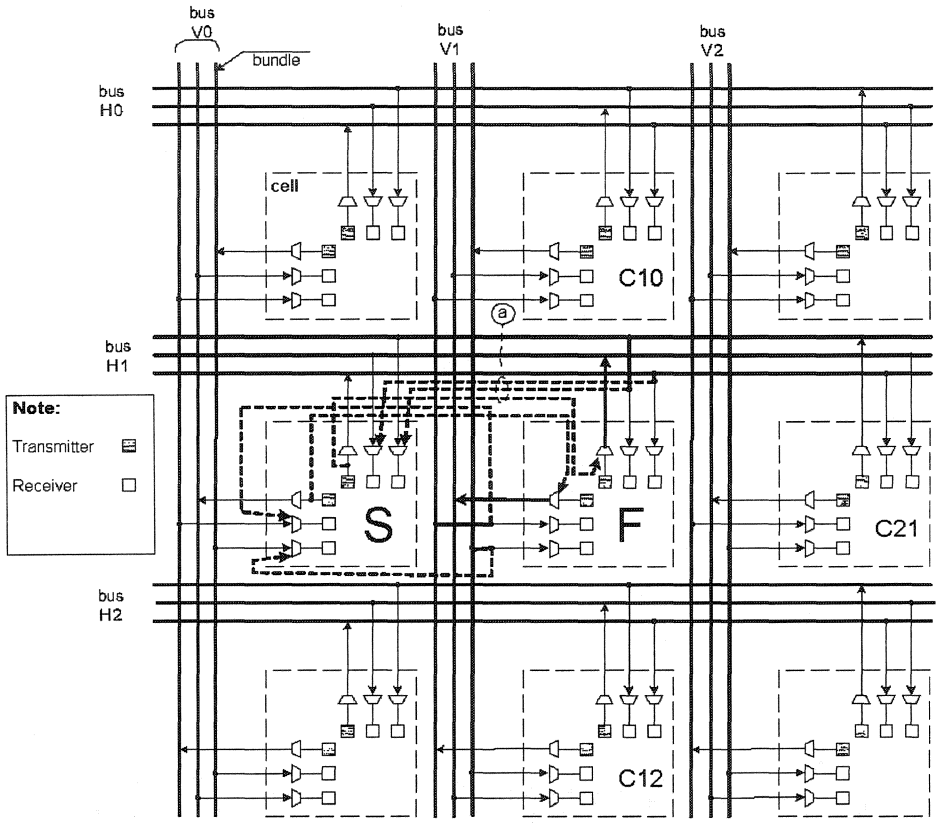


Figure 3.10 Two-dimensional bypassing supported by type C connections

There are two ways to replace a neighbour cell. The first one is one-dimensional bypass shown in Figure 3.5. In the second way, as shown in Figure 3.10, the replacement is achieved by bypassing vertical and horizontal transmitting and receiving connections of cell S to the buses that are associated with cell F. The group of connections @ connect cell S's horizontal transmitter to cell F's horizontal transmitting multiplexer, and connect cell S's receiving multiplexers to cell F's taps on horizontal bus. After a two-dimensional bypass replacement, cell S can communicate with the cells within the same logical row (C21) or column (C10 and C12) of cell F in the same way as cell F does with default connections. All connections, of types A, B and C, support one-dimensional bypass, whereas type C connections also support two-dimensional bypass. There is no significant

advantage for two-dimensional bypass over one-dimensional bypass when the objective is to replace a single isolated defective cell.

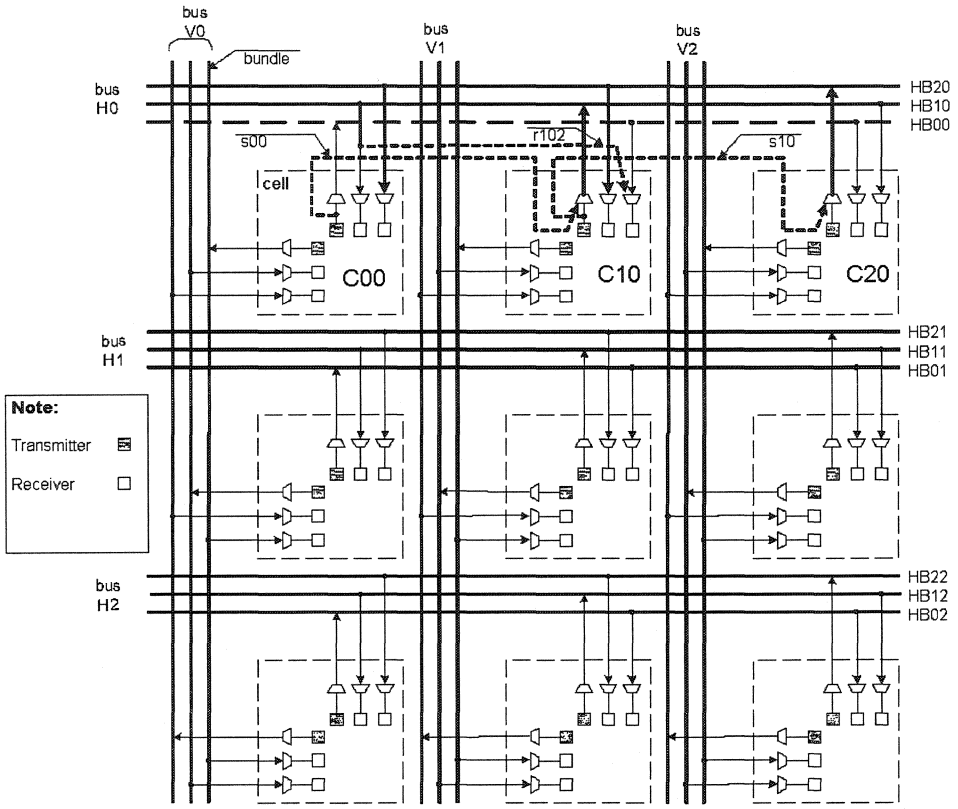


Figure 3.11 Intra-bus bundle replacement supported by type C connections

There are two interesting methods for bundle replacement. Inter-bus bundle replacement, which is shown in Figure 3.6, replaces a bundle with the bundles having the same rank in one of the neighbour buses. By contrast, intra-bus replacement replaces a bundle with the neighbour bundles of the same bus. As shown in Figure 3.11, cell C00 can borrow HB10 from cell C10 via connection s00; cell C10 can listen from bundle HB10 via connection r102; cell C10 can borrow cell C20's bundle via connection s10. These operations can be rippled until a spare bundle is reached. Inter-bus replacement is supported by type A, B and C connections. However, intra-bus replacement is only supported by type C connections. Connections of type A and B do not support intra-bus replacement since there are not enough bypass connections.

The number of receiving connection wires is far more than the number of transmitting connection wires. Type C connections require much more wires than types of A and B connections. It is interesting to take advantage of the 5-source transmitter connections of type B. As depicted in Figure 3.12, via connection s00, bundle HB00 can be replaced by bundle HB10. However, this replacement causes cell C10 to be useless, since once cell C10 lends HB10 to cell C00, cell C10 can no longer listen from that bundle. As a consequence, cell C10 cannot receive from the cell C00 anymore. Nevertheless, if cell C10 is defective, this replacement is worth while.

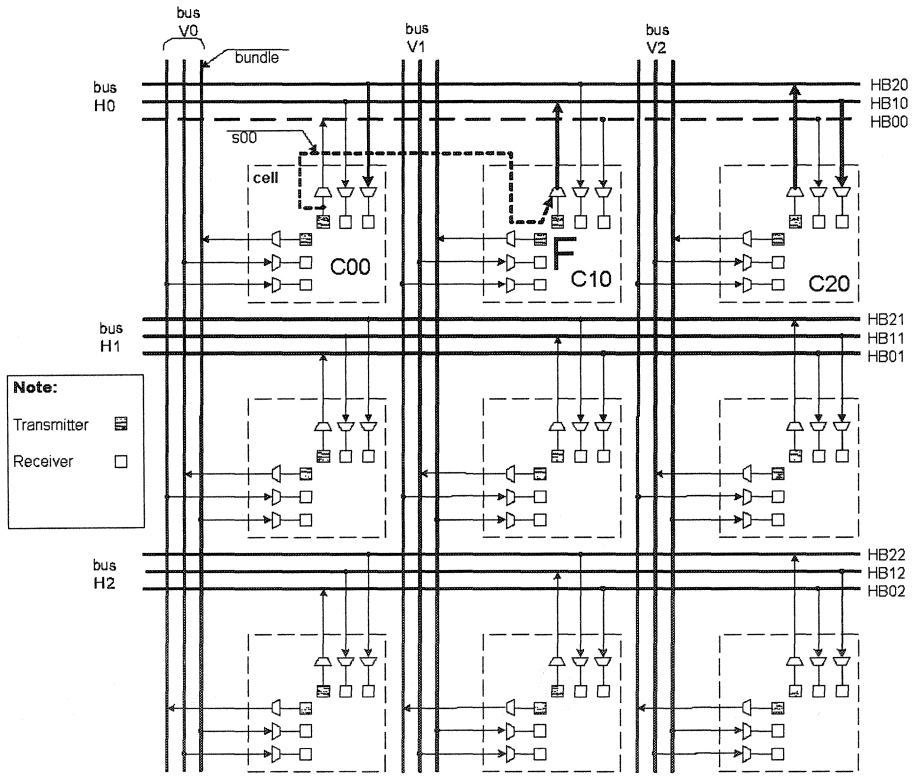


Figure 3.12 Bundle of a defective cell used by a neighbouring cell for type B connections

To support intra-bus bundle replacement, it is necessary to monitor the outputs of the naturally owned transmitting multiplexer. Instead of adding inter-cell configuration wires

r102 in Figure 3.11, the transmitting signal can be monitored locally with the self-monitoring wires r102 in Figure 3.13. This self-monitoring is a key feature of Pseudo-5-source connections (type D connections).

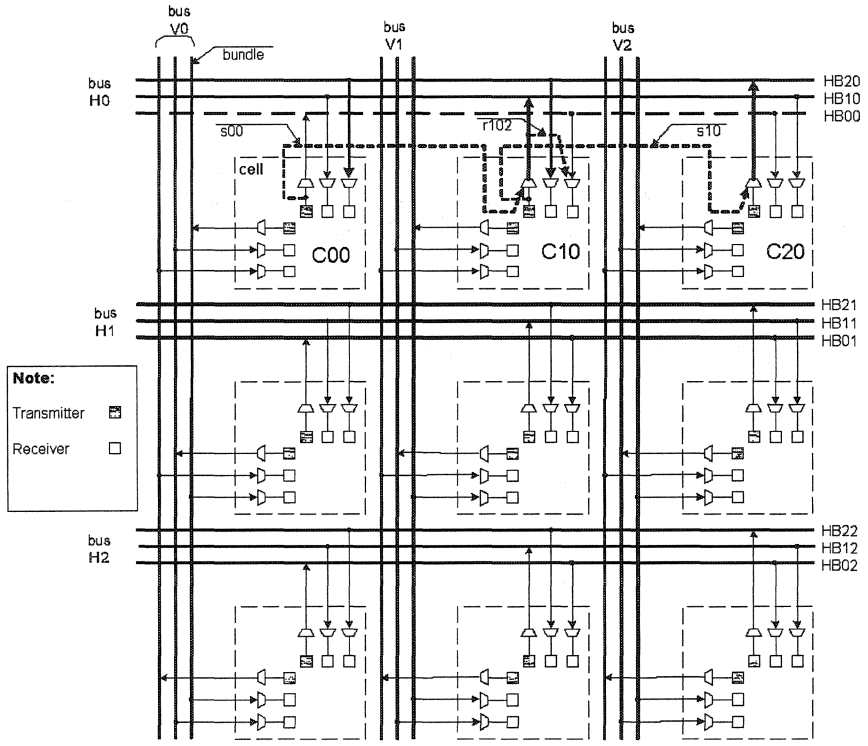


Figure 3.13 Intra-bus bundle replacement supported by type D connections

Figure 3.14 shows an example that the turned path cell replacement is supported by the self-monitoring link. The intended turned path replacement shown in Figure 3.7 succeeds with a combination of one-dimensional cell replacements and an intra-bus bundle replacement. Cell C11 talks to cell C01 via receiving link r01vr1; cell C01 talks to cell C11 via self-monitoring link sm11v1. Another turned path replacement example with full connection details is shown in Figure A.26.

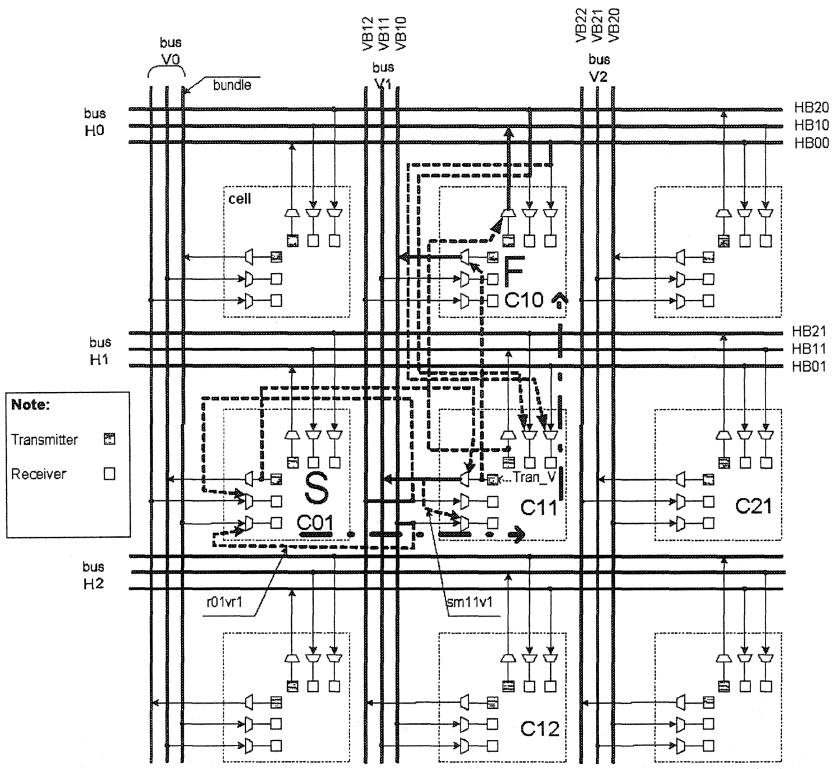


Figure 3.14 Turned path replacement supported by the self-monitoring

Connections of Pseudo-5-source Configuration Structure (Type D)

Pseudo-5-source connections (Type D) are the combination of type B connections and the self-monitoring links. Its ability to support replacements is close to type C connections, while its wiring overhead is close to type A and type B connections.

The multiplexers driven by transmitters are connected to 5 sources as shown in Figure 3.8. The typical receiver multiplexer connections have 3 sources (the bundle of the associated receiver, the two bundles with the same rank of the neighbour buses), which is the same as shown in Figure 3.4. The output bundles of the transmitter multiplexers of every cell on every bus are also connected to the receiver multiplexers listening to the neighbour cells through one of its inputs. A typical example is shown in Figure 3.15. The self-monitoring links are drawn by heavy dashed lines. The self-monitoring links of Cell 3 are connected to the receiving multiplexers listening to bundle 2 (the one where the left

neighbour, Cell 2, is connected by default) and bundle 4 (the one where the right neighbour, Cell 4, is connected by default). If Cell 2 borrows bundle B3, R4 is used for self-monitoring, since Cell 3 is going to borrow Cell 4's bundle, and it does not need to listen from bundle 4 any more. If Cell 4 borrows bundle B3, R2 is used for self-monitoring.

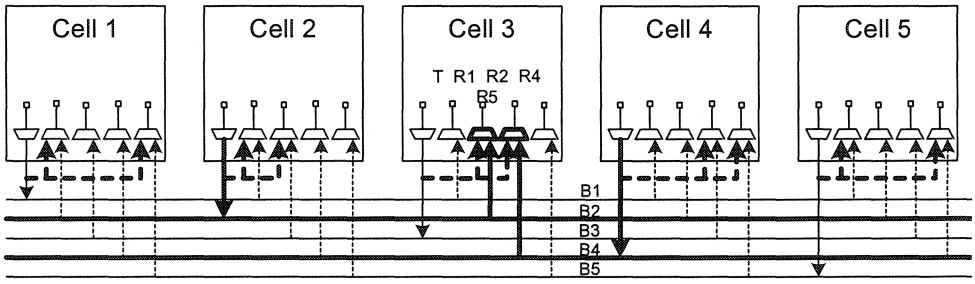


Figure 3.15 Self-monitoring links of Pseudo-5 receiver connection

After the integration with the regular bus structure shown in Figure 3.1, the regular layout presentation of self-monitoring is depicted in Figure 3.16. Since the positions of the receiving multiplexers in one cell listening to neighbor cells are fixed, the layout of the self-monitoring has its regularity.

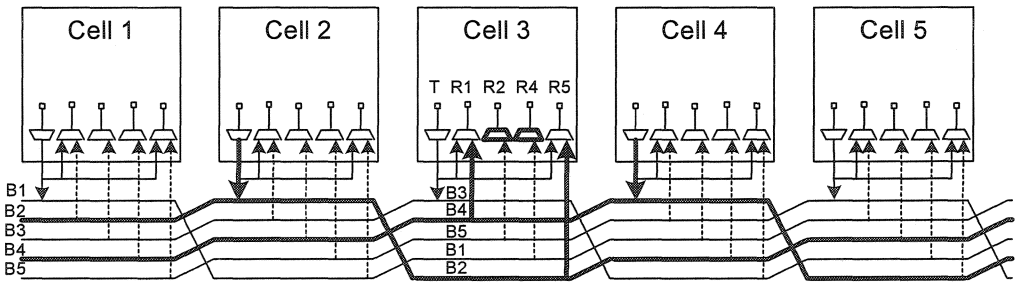


Figure 3.16 Regular layout version of self-monitoring links

3.3.3 Comparison and Conclusion of Types of Interconnections

This section compares the 4 connection types in terms of the number of configuration wires, the number of registers needed to control the configuration, and the replacements supported. The comparison is based on an N x N array with x lines per bundle and an example of a *20* x *20* array with *40* wires per bundle is given to show typical values. Table 3.1 to Table 3.8 demonstrate how the number of bits to control and the number of configuration lines are calculated and the numbers corresponding to the example. Table 3.9 summarizes the comparison on the four connection types.

Table 3.1 Bits to control for type A connections

Size of array		*N x N*	*20 x 20*
Receiver	**Bits / bundle**	*1* for direction + *2* for source	*3*
	Bundles / dimension	*N - 1*	*19*
	Dimension	*2*	*2*
	Total	*6 * (N – 1)*	*114*
Transmitter	**Bits / dimension**	*2*	*2*
	Dimensions	*2*	*2*
	Total	*4*	*4*
Total		*4 + 6 * (N – 1)*	*118*

Table 3.2 Configuration lines for type A connections

Size of array		*N x N*	*20 x 20*
Receiver	**Lines / bundle**	*x*	*40*
	Bundles / 2 way	*2*	*2*
	Bundles / bus	*N - 1*	*19*
	Neighbors / bus	*2*	*2*
	Buses / 2 dimensions	*2*	*2*
	Total	*8 * x * (N-1)*	*6080*
Transmitter	**Lines / bundle**	*x*	*40*
	Bundles / 2ways	*2*	*2*
	Neighborhoods	*2*	*2*
	Dimensions	*2*	*2*
	Total	*8 * x*	*320*
Total		*8 * x* N*	*6400*
Lines / bus = Total / 4		*2 * x* N*	*1600*

Table 3.3 Bits to control for type B connections

Size of array		*N x N*	*20 x 20*
Receiver	**Bits / bundle**	*1* for direction + *2* for source	*3*
	Bundles / dimension	*N – 1*	*19*
	Dimension	*2*	*2*
	Total	*6 * (N – 1)*	*114*
Transmitter	**Bits / dimension**	*3*	*3*
	Dimensions	*2*	*2*
	Total	*6*	*6*
Total		*6 * N*	*120*

Table 3.4 Configuration lines for type B connections

Size of array		*N x N*	*20 x 20*
Receiver	**Lines / bundle**	*x*	*40*
	Bundles / 2 way	*2*	*2*
	Bundles / bus	*N - 1*	*19*
	Neighbors / bus	*2*	*2*
	Buses / 2 dimensions	*2*	*2*
	Total	*8 * x * (N – 1)*	*6080*
Transmitter	**Lines / bundle**	*x*	*40*
	Bundles / 2ways	*2*	*2*
	Neighborhoods	*4*	*4*
	Dimensions	*2*	*2*
	Total	*16 * x*	*640*
Total		*8 * x* (N+1)*	*6720*
Lines / bus = Total / 4		*2 * x* (N+1)*	*1680*

Table 3.5 Bits to control for type C connections

Size of array		*N x N*	*20 x 20*
Receiver	**Bits / bundle**	*1* for direction + *3* for source	*4*
	Bundles / dimension	*N - 1*	*19*
	Dimension	*2*	*2*
	Total	*8 * (N – 1)*	*152*
Transmitter	**Bits / dimension**	*3*	*3*
	Dimensions	*2*	*2*
	Total	*6*	*6*
Total		*6 + 8 * (N – 1)*	*158*

Table 3.6 Configuration lines for type C connections

Size of array		*N x N*	*20 x 20*
Receiver	**Lines / bundle**	*x*	*40*
	Bundles / 2 way	*2*	*2*
	Bundles / bus	*N - 1*	*19*
	Neighbors / bus	*4*	*4*
	Buses / 2 dimensions	*2*	*2*
	Total	*16 * x *(N – 1)*	*12160*
Transmitter	**Lines / bundle**	*x*	*40*
	Bundles / 2ways	*2*	*2*
	Neighborhoods	*4*	*4*
	Dimensions	*2*	*2*
	Total	*16 * x*	*640*
Total		*16 * x* N*	*12800*
Lines / bus = Total / 4		*4 * x* N*	*3200*

Table 3.7 Bits to control for type D connections

Size of array		*N x N*	*20 x 20*
Receiver	**Bits / bundle**	*1* for direction + *2* for source	*3*
	Bundles / dimension	*N - 1*	*19*
	Dimension	*2*	*2*
	Total	*6 * (N – 1)*	*114*
Transmitter	**Bits / dimension**	*3*	*3*
	Dimensions	*2*	*2*
	Total	*6*	*6*
Total		*6 * N*	*120*

Table 3.8 Configuration lines for type D connections

Size of array		*N x N*	*20 x 20*
Receiver	**Lines / bundle**	*x*	*40*
	Bundles / 2 way	*2*	*2*
	Bundles / bus	*N - 1*	*19*
	Neighbors / bus	*2*	*2*
	Buses / 2 dimensions	*2*	*2*
	Total	*8 * x * (N – 1)*	*6080*
Transmitter	**Lines / bundle**	*x*	*40*
	Bundles / 2ways	*2*	*2*
	Neighborhoods	*4*	*4*
	Dimensions	*2*	*2*
	Total	*16 * x*	*640*
Total		*8 * x* (N+1)*	*6720*
Lines / bus = Total / 4		*2 * x* (N+1)*	*1680*

Table 3.9 Comparison on configuration connection structures

Connection types		A	B	C	D
Bits to control	***N* * *N*** array ***x*** lines/ bundle	*4+6*(N -1)*	*6*N*	*6+8*(N -1)*	*6*N*
	20 * 20 array **19** lines/ bundle	*118*	*120*	*158*	*120*
Bypass wires / bus	***N* * *N*** array ***x*** lines/ bundle	*2*x*N*	*2*x*(N+1)*	*4*x*N*	*2*x*(N+1)*
	20 * 20 array **19** lines/ bundle	*1600*	*1680*	*3200*	*1680*
Cell replacement (One-dimensional bypass)		*Supported*	*Supported*	*Supported*	*Supported*
Inter-bus bundle replacement		*Supported*	*Supported*	*Supported*	*Supported*
Intra-bus bundle replacement		*Not supported*	*Not supported*	*Supported*	*Supported*
Using defective neighboring cells' bundle in the same bus		*Not supported*	*Supported*	*Supported*	*Supported*
Turned path cell replacement		*Not supported*	*Not supported*	*Supported*	*Supported*

Intra-bus bundle replacement enables bundle shift in a second direction to align good cells and good bundles. Turned path cell replacement is supported with the combination of one dimensional bypass cell replacement and intra-bus bundle replacement. The Pseudo-5-source connection supports intra-bus bundle replacement and turned path cell replacement with a small amount of wires and control overhead. It has significant advantages over type A and B connections in terms of functionality, and has significant advantages over type C connections in terms of circuit simplicity.

3.3.4 Example of a 3 x 3 Physical Interconnection Array Structure

To demonstrate how does the configuration structure works, it is convenient to start with a simple example. Figure 3.17 shows an example of a 3 x 3 comprehensive interconnection structure combining both the interleaved bus structure and Pseudo-5-source connection configuration structure. The bold lines denote normal data buses and the other lines denote bypass links for configuration. Each line in this figure represents a bundle of wires. All configuration multiplexers in a cell are assumed to be controlled by a data register that is part of a JTAG chain embedded in each cell. In this 3 x 3 structure, the cell located at the centre has 8 neighbours, which reflects the normal cell situation, whereas the other cells reflect all possible boundary situations. The 3 x 3 structure is thus a representative of all possible situations. Most of the following cells and bundle replacement analysis is demonstrated using this 3 x 3 structure.

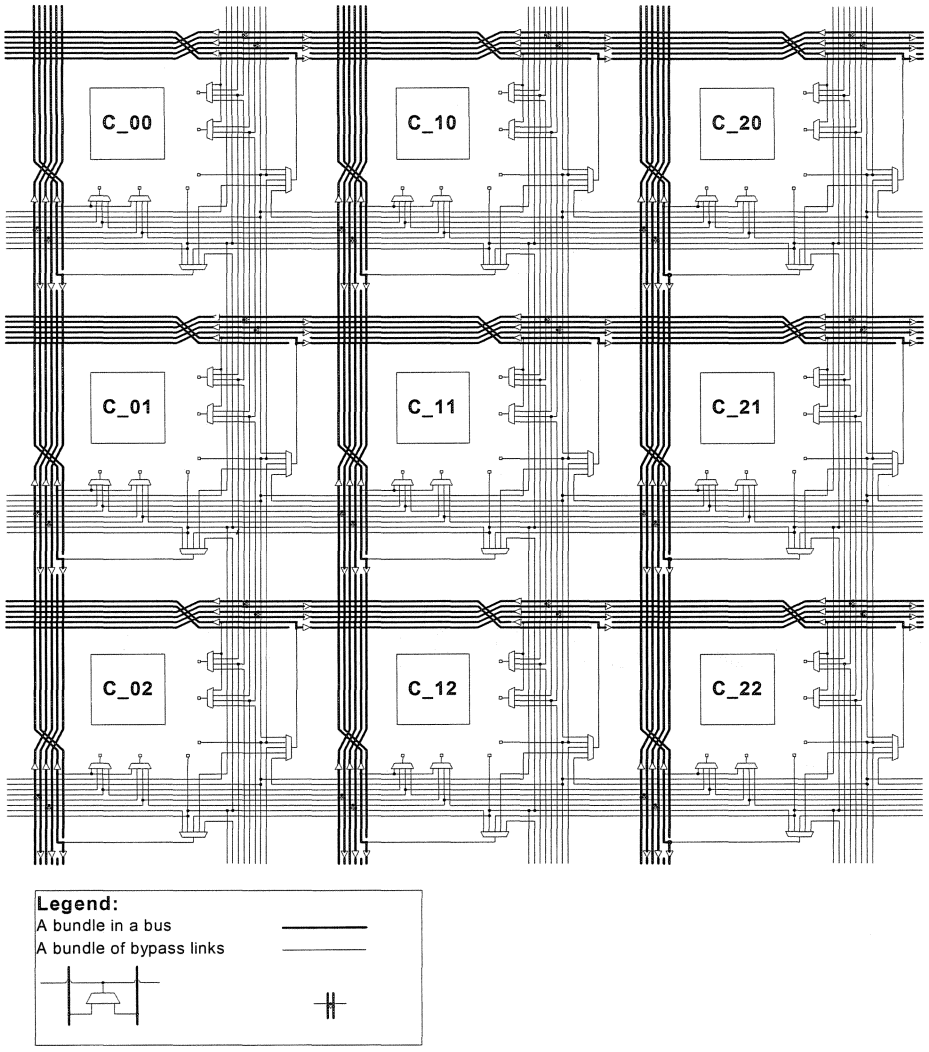


Figure 3.17 Example of a 3 x 3 array

3.4 Replacement Analysis of Pseudo-5–source Connection Structure

Pseudo-5-source connection structure supports 12 basic configuration replacement operations that can be combined to repair complex fault patterns under certain constraints.

3.4.1 Basic Replacement Operations

The following basic replacement operations are supported by this structure.

- Cell replacement: a cell can be replaced by one of its four (top, bottom, left and right) neighbours.

- Inter-bus bundle replacement: a bundle can be replaced by one of the bundles of the same rank in the neighbor buses.

- Intra-bus bundle replacement: a bundle can be replaced by one of the neighbor bundles in the same bus.

3.4.2 Notations

A set of notations is defined to describe the configuration procedure and its effect on the configuration control data for the multiplexers.

Since the same type of configuration operation can be rippled through cells or bundles from a replaced cell or bundle to reach an available cell or bundle, we define a parameter n as the number of stages a configuration replacement shifts.

Cell: includes a PE and its associated configuration control elements (JTAG and receiving multiplexers for the structure). The indexes of a cell are the same as those of the associated PE. The physical indexes of the 3 x 3 example are shown in Table 3.10. The indexes of configuration multiplexers and the corresponding inputs are shown in Figure 3.18. The descriptions of data structures of cells to demonstrate configuration connections are summarized in Table 3.11.

Table 3.10 Physical indexes of the structure

P(0,0)	P(1,0)	P(2,0)
P(0,1)	P(1,1)	P(2,1)
P(0,2)	P(1,2)	P(2,2)

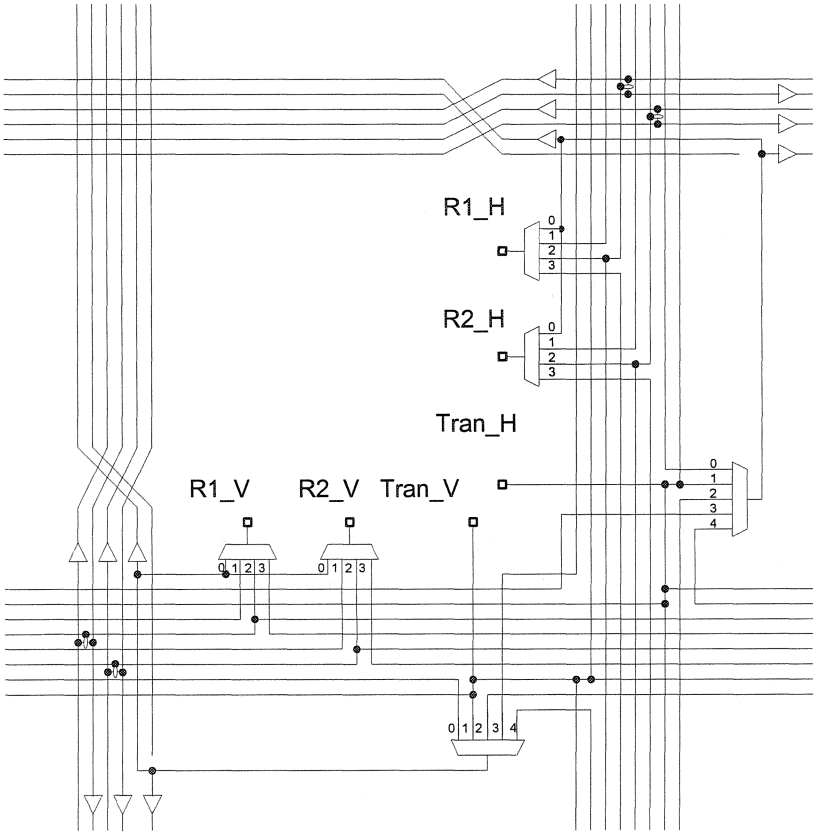


Figure 3.18 Notation for configuration multiplexers and their inputs

Table 3.11 Cell Data Structure

Symbol	Description
P(i,j)	Physical index: the index denotes where a cell is physically located
L(i,j)	Logical index: the index denotes where a cell is logically located
C(R1_H, R2_H, Tran_H, Tran_V, R1_V, R2_V)	Receiver and transmitter physical connections, the physical position is shown in Figure 3.15.
R1_H	The receiver of the 1st bundle of the horizontal bus
R2_H	The receiver of the 2nd bundle of the horizontal bus
Tran_H	Transmitter of the horizontal bus
Tran_V	Transmitter of the vertical bus
R1_V	Receiver of the 1st bundle of the vertical bus
R2_V	Receiver of the 2nd bundle of the vertical bus
S(s)	Status of the cell Where s = H healthy s = F faulty s = U undetermined

Bundle: A group of wires of the regular bus structure, which can be used to establish the connections among a horizontal/vertical transmitter of one cell and the horizontal/vertical receivers of a row/column of cells. A bundle includes long interconnect lines, drivers, and transmitting multiplexers. The data structure representing the configuration connections of a bundle is shown in Table 3.12.

Table 3.12 Bundle Data Structure

Symbol	Description
B[d,(i,j),(k,l)]	The location of a bundle and its current connection
d	Dimension: H for horizontal V for vertical
(i,j)	Default transmitter connection (i.e. the physical index of the naturally associated cell), it is the physical index of the bundle
(k,l)	Current transmitter connection, it is the physical index of the connected cell
S(s)	Status of a bundle, where s = H healthy s = F faulty s = U undetermined

Special indexes defined in Table 3.13 are used to represent indexes of a cell or a bundle when it is not convenient to represent it with indexes composed of certain values.

Table 3.13 Special indexes

Symbol	Description
(O,O)	The logical index of the cell which currently is out the target logical array
(-,-)	Don't care the connection
(X,X)	Unknown connection

Simplified Abstraction

Instead of drawing all the connection details to demonstrate configurations as the way shown in Figure 3.17, configuration connections can be demonstrated in a simplified way. As shown in Figure 3.19, (b) is the simplified abstraction of (a). The equivalence of (a) and (b) will be proved in section 3.5.1. It is similar for vertical bus connections.

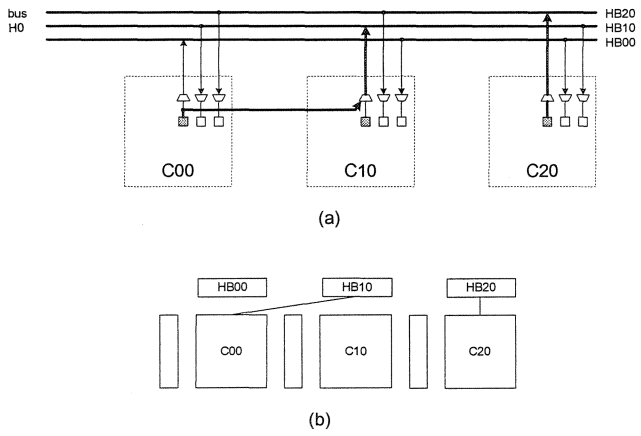


Figure 3.19 Diagram (b) is the simplified abstraction of (a)

Configuration Operations

There are total 12 basic configuration operations, 4 for cell replacements, 4 for inter-bus bundle replacements and 4 for intra-bus bundle replacement. The symbols representing the operations, the corresponding replacement operation functions, the indexes to the

associated example figures and the indexes to the equivalent simplified drawings are shown in Table 3.14, Table 3.15 and Table 3.16. (i,j) denotes the physical index of the cell being replaced. n denotes the number of steps of simple replacement. Examples of basic configuration operations are shown in Figure A.2 to Figure A.24 in Annex A.

Table 3.14 Cell replacement operations

Symbol	Description of Function	Representative Example	Simplified drawing
R_C_U[(i,j),n]	**R**eplace **C**ell **U**p shift	Figure A.2 of Annex A	Figure 3.20(a)
R_C_D[(i,j),n]	**R**eplace **C**ell **D**own shift	Figure A.3 of Annex A;	Figure 3.20(b)
R_C_L[(i,j),n]	**R**eplace **C**ell **L**eft shift	Figure A.4, of Annex A	Figure 3.20(c)
R_C_R[(i,j),n]	**R**eplace **C**ell **R**ight shift	Figure A.5 of Annex A	Figure 3.20(d)

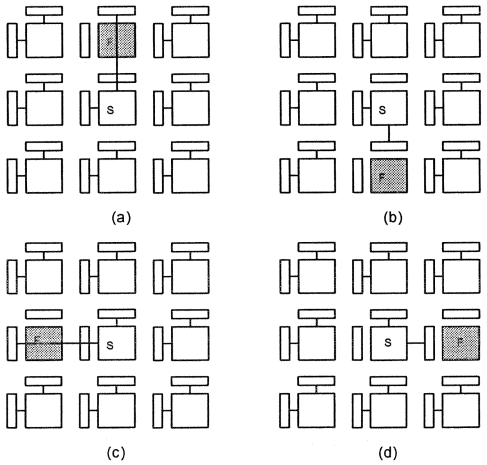


Figure 3.20 Simplified abstractions of cell replacements

Table 3.15 Inter-bus bundle replacement operations

Symbol	Description of Function	Representative Example	Simplified drawing
R_B_V_L[(i,j),n]	**R**eplace **V**ertical **B**undle inter-bus **L**eft shift	Figure A.9 of Annex A	Figure 3.21(a)
R_B_V_R[(i,j),n]	**R**eplace **V**ertical **B**undle inter-bus **R**ight shift	Figure A.10 of Annex A	Figure 3.21(b)
R_B_H_U[(i,j),n]	**R**eplace **H**orizontal **B**undle inter-bus **U**p shift	Figure A.13 of Annex A	Figure 3.21(c)
R_B_H_D[(i,j),n]	**R**eplace **H**orizontal **B**undle inter-bus **D**own shift	Figure A.14 of Annex A	Figure 3.21(d)

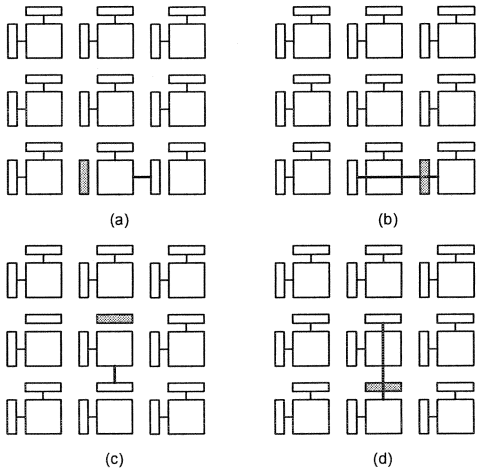


Figure 3.21 Simplified abstractions of inter-bus bundle replacements

Table 3.16 Intra-bus bundle replacement operations

Symbol	Description of Function	Representative Example	Simplified drawing
R_B_V_D[(i,j),n]	**R**eplace **V**ertical **B**undle intra-bus **D**own shift	Figure A.17 of Annex A	Figure 3.22(a)
R_B_V_U[(i,j),n]	**R**eplace **V**ertical **B**undle intra-bus **U**p shift	Figure A.19 of Annex A	Figure 3.22(b)
R_B_H_L[(i,j),n]	**R**eplace **H**orizontal **B**undle intra-bus **L**eft shift	Figure A.21 of Annex A	Figure 3.22(c)
R_B_H_R[(i,j),n]	**R**eplace **H**orizontal **B**undle intra-bus **R**ight shift	Figure A.23 of Annex A	Figure 3.22(d)

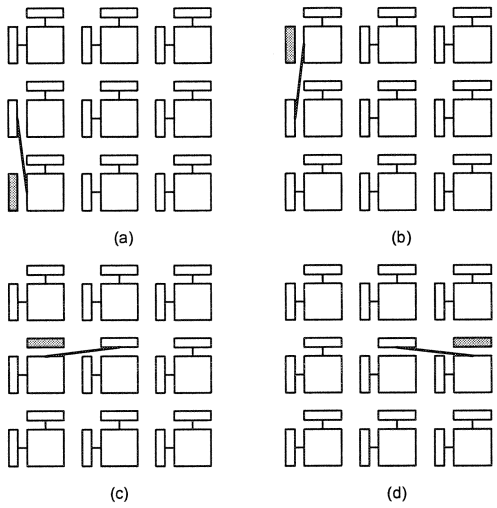


Figure 3.22 Simplified abstractions of intra-bus bundle replacements

3.4.3 Tabular Presentation of The Configuration Structure

Tabular presentation is essential for further development such as programming configuration algorithms. The tabular connectivity presentation of default configuration

(shown in Figure A.1 in Annex A) is shown in Table 3.17. The graphical detail connections of the configuration after a R_C_L[(0,1),1] operation is shown in Figure A.4 in Annex A. Its corresponding tabular presentation is shown in Table 3.18, in which the underscores indicate the changes made by the configuration operation.

Table 3.17 Tabular presentation of the default configuration

Physical Index	Logical Index	Multiplexer connection	Status
P (0,0)	L (0,0)	C [(1,0), (2,0), (0,0), (0,0), (0,2), (0,1)]	S(H)
P (1,0)	L (1,0)	C [(2,0), (0,0), (1,0), (1,0), (1,2), (1,1)]	S(H)
P (2,0)	L (2,0)	C [(0,0), (1,0), (2,0), (2,0), (2,2), (2,1)]	S(H)
P (0,1)	L (0,1)	C [(1,1), (2,1), (0,1), (0,1), (0,0), (0,2)]	S(F)
P (1,1)	L (1,1)	C [(2,1), (0,1), (1,1), (1,1), (1,0), (1,2)]	S(H)
P (2,1)	L (2,1)	C [(0,1), (1,1), (2,1), (2,1), (2,0), (2,2)]	S(H)
P (0,2)	L (0,2)	C [(1,2), (2,2), (0,2), (0,2), (0,1), (0,0)]	S(H)
P (1,2)	L (1,2)	C [(2,2), (0,2), (1,2), (1,2), (1,1), (1,0)]	S(H)
P (2,2)	L (2,2)	C [(0,2), (1,2), (2,2), (2,2), (2,1), (2,0)]	S(H)

The first row of Table 3.17 shows that the physical cell P(0,0) is assigned a logical index of L(0,0), that the R1_H is listening from physical cell C(1,0), that the R2_H is listening from physical cell C(2,0), that the Tran_H is connected to bundle B[H,(0,0)], that the Tran_V is connected to bundle B[V,(0,0)], that R2_V is listening to physical cell C(0,2), that R1_V is listening to physical cell C(0,1), and that the cell P(0,0) is healthy. The other data can be interpreted accordingly.

Table 3.18 Tabular presentation after a R_C_L[(0,1),1] operation

Physical Index	Logical Index	Multiplexer connection	Status
P (0,0)	L (0,0)	C [(1,0), (2,0), (0,0), (0,0), (0,2), (0,1)]	S(H)
P (1,0)	L (1,0)	C [(2,0), (0,0), (1,0), (1,0), (1,2), (1,1)]	S(H)
P (2,0)	L (2,0)	C [(0,0), (1,0), (2,0), (2,0), (2,2), (2,1)]	S(H)
P (0,1)	L (O,O)	C [(1,1), (2,1), (0,1), (-,-), (-,-), (-,-)]	S(F)
P (1,1)	L (0,1)	C [(2,1), (0,1), (1,1), (0,1), (0,0), (0,2)]	S(H)
P (2,1)	L (2,1)	C [(0,1), (1,1), (2,1), (2,1), (2,0), (2,2)]	S(H)
P (0,2)	L (0,2)	C [(1,2), (2,2), (0,2), (0,2), (1,1), (0,0)]	S(H)
P (1,2)	L (1,2)	C [(2,2), (0,2), (1,2), (1,2), (1,1), (1,0)]	S(H)
P (2,2)	L (2,2)	C [(0,2), (1,2), (2,2), (2,2), (2,1), (2,0)]	S(H)

3.4.4 Replacement Solutions by Combination of Operations

Some configuration operations such as turned path replacement can be conducted by a combination of operations. As showed in Figure A.32, it is possible to do a replacement of a cell with an U-turned path. As shown in Figure A.38 in Annex A, by the combination of an inter-bus bundle replacement with an intra-bus bundle replacement, a bundle can be replaced by an L-turned path. As shown in Figure A.41 in Annex A, a cell can be replaced diagonally by the combination of two cell replacements.

3.4.5 Serviceability of Configuration Wires

It is proved that all configuration connections are used to support the 12 basic configuration operations. Table 3.19 shows all configuration connections, and indexes of the cells and the pages in which the connections are used to support one of the 12 basic replacement operations. The index of the inputs of multiplexers is defined in Figure 3.18.

Table 3.19 Serviceability Analysis of All Configuration Wires

Connections	Page	Physical indexes of the cell	Configuration Operations
R1_H_0	A.22	(1,1)	R_B_H_L
R1_H_1	A.2	(1,1)	R_C_U
R1_H_2	A.1	All	Default
R1_H_3	A.3	(1,1)	R_C_D
R2_H_0	A.24	(1,1)	R_B_H_R
R2_H_1	A.2	(1,1)	R_C_U
R2_H_2	A.1	All	Default
R2_H_3	A.3	(1,1)	R_C_D
Tran_H_0	A.3	(1,2)	R_C_D
Tran_H_1	A.1	All	Default
Tran_H_2	A.2	(1,0)	R_C_U
Tran_H_3	A.21	(1,1)	R_B_H_L
Tran_H_4	A.23	(1,1)	R_B_H_R
R1_V_0	A.20	(0,1)	R_B_V_U
R1_V_1	A.4	(1,1)	R_C_L
R1_V_2	A.1	All	Default
R1_V_3	A.5	(1,1)	R_C_R
R2_V_0	A.18	(0,1)	R_B_V_D
R2_V_1	A.4	(1,1)	R_C_L
R2_V_2	A.1	All	Default
R2_V_3	A.5	(1,1)	R_C_R
Tran_V_0	A.5	(2,1)	R_C_R
Tran_V_1	A.1	All	Default
Tran_V_2	A.4	(0,1)	R_C_L
Tran_V_3	A.19	(0,1)	R_B_V_U
Tran_V_4	A.17	(0,1)	R_B_V_D

3.4.6 Ability and Limitation of Replacements

The Pseudo-5 connections structure is a very strong fault tolerant structure since the fault on interconnections as well as the fault on cells can be tolerated. Any amount of redundancy can be specified in this structure and any cell or bundle can be equally specified as a primary element or as a redundant element. Cell can be directly replaced by

one of its eight neighbors, and bundle can be directly replaced by the neighbor bundles of the same bus, as well as by the same rank bundles of its neighbor buses. The basic configuration replacements can be combined under certain constraints. No single defect can cause this structure to fail. With diverse cell replacements, bundle replacements and their combinations, many clustered fault patterns can be tolerated.

However, there do exist faults that significantly affect replacements. Firstly, faults on the JTAG configuration chain may cause one or a combination of following equivalent faults: a defective cell if its associated cell cannot receive signal correctly from an identified good bundle; 2 defective bundles (one vertical and one horizontal) if the 2 transmitter multiplexers cannot be properly controlled; a defective cell and serious connectivity problems to its 4 neighbourhoods if the direction multiplexers cannot be properly controlled. For the last situation, the 2 neighbour cells in the same row cannot receive from the vertical bus associated with this cell, the 2 neighbour cells in the same column cannot receive from the horizontal bus associated with this cell. Also, since there are bypass links between two neighbour cells only, a cell cannot jump over a defective cell to replace another defective cell, a cell with eight defective neighbors cannot be replaced. Another factor that limits replacements is that not all replacement operations can be combined freely in a region. It is interesting to investigate those constraints.

3.4.7 Inter-operation Conflicts

As the analysis presented in Annex B, some combinations of configuration operations are supported and some are not. It is very hard to formalize these constraints due to the many relevant details. This cross operations constraints analysis is helpful to understand configuration operations, but it is hard use to develop configuration algorithms due to its complexity. A set of simpler and thus more effective configuration rules should be developed.

3.5 Configuration Rules

Violating the inter-operation constraints described in section 3.4.7 leads to requesting connections that are conflicting or physically unfeasible. The strategy proposed in this thesis is to develop a set of connectivity-based configuration rules that ensure all the intended connections are feasible simultaneously. Configuration algorithms are easier to develop using simple connectivity-based rules rather than using more complicated inter-operations constraints rules.

A critical issue when formalizing configuration rules and developing configuration algorithms is the complexity of the interconnection structure. As proposed, a complex structure can be expressed by a simpler scheme, masking some details that can be recovered or synthesized. Thus, the analysis procedure can be simplified using the proposed abstract representation that is equivalent to the full connections.

3.5.1 Receiver Connection Recovering From Simplified Abstraction

Figure 3.23 shows the full connections of a configuration example. The cells marked by F represent defective cells. The bundles drawn by dashed lines represent defective bundles. The desired configuration is expressed by logical indexes of cells and the active interconnections, which are shown with bold lines. The corresponding simplified abstraction shown in Figure 3.24 with cell indexes and transmitter connections will be used for formalizing configuration rules and developing configuration algorithms. Since receiver connections are ignored in simplified abstraction, the feasibility of receiver connections should be ensured if the configuration rules described in section 3.5.2 are followed.

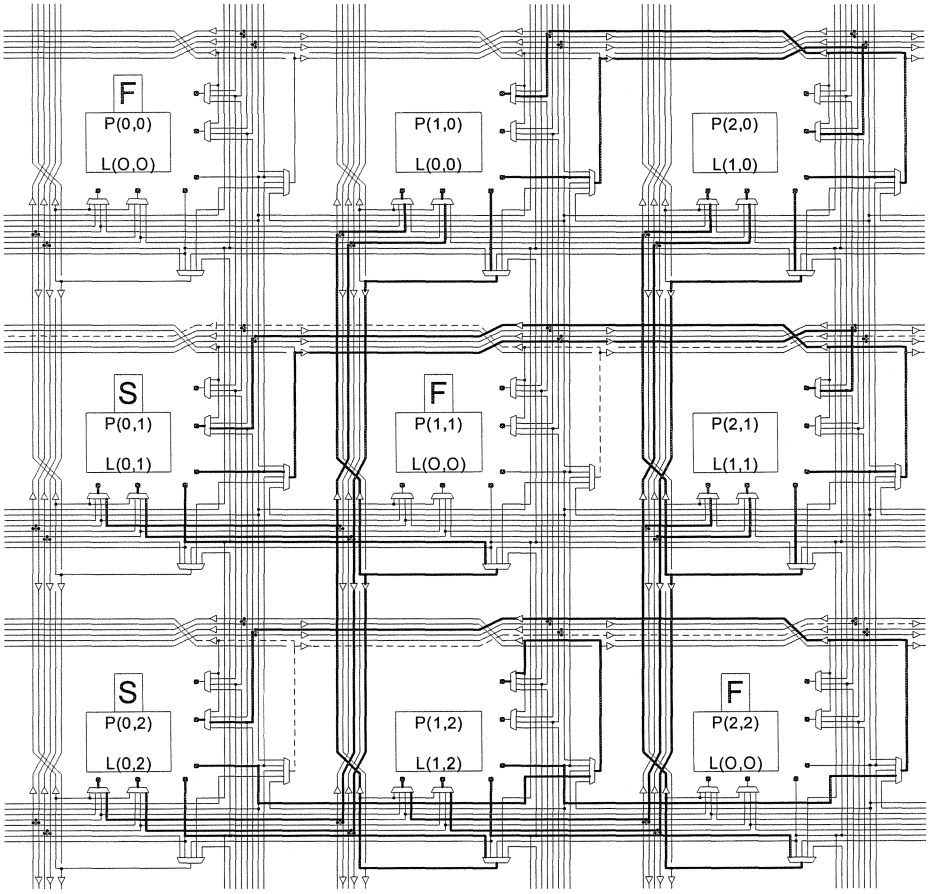


Figure 3.23 Full connections of a configuration

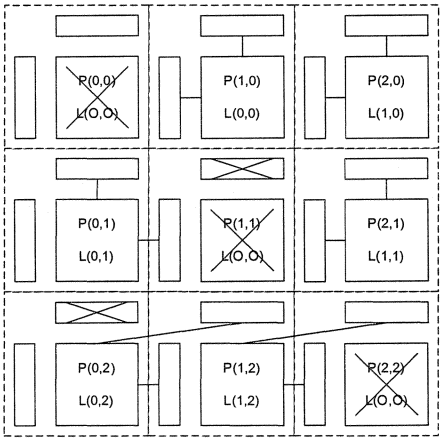


Figure 3.24 A simplified abstraction of Figure 3.23

In this sub-section, an algorithm used to recover receiver connections from simplified abstraction is proposed. The equivalence of simplified abstraction and full connection of a configuration is supported by the regular layout structure. Once a vertical bundle is connected to a cell, other active cells located at the same logical column should listen to the bundle. The vertical physical position of the listening cell relative to the bundle determines which receiving multiplexer of the cell should be used to link this bundle. The horizontal physical position of the listening cell relative to the bundle determines which input of the receiving multiplexer should be selected. The same is true for horizontal bundles.

The algorithm expressed by a flow chart shown in Figure 3.25 is used to determine the receiving multiplexer connections listening to relevant active bundles. This flow chart is for horizontal bundle connections. For vertical connections, the algorithm will be similar. In the flow chart, B stands for the transmitting bundle, CR stands for the receiving cell, CH[m] represents multiplexers associated with horizontal buses. The definition of the index m is described using the following example. Note that different values of this index correspond to various situations with respect to the target configuration that changes in the presence of defects. Taking the cell shown in Figure 3.18 for example, CH[0] corresponds to Tran_H, CH[2] corresponds to R1_H, and CH[2] corresponds to R2_H. In reference to Figure 3.26, index m1 points to the receiving multiplexer, which the bundle B is talking to when there is no intra bundle shift involved. In reference to Figure 3.27, index m2 points to the receiving multiplexer normally used to listen to the bundle BCR, which is borrowed by CR in this case. We refer to self-monitoring connection with respect to the output multiplexer, which may be borrowed by a neighbour cell if intra-bus bundle replacement is used. That connection uses the self-monitoring link required with intra-bus bundle replacement, which forces introducing a shift in the logical role of input multiplexers to a cell (analogous to a logical index shift for the bundles entering each cell, when a cell uses the natural bundle of its neighbour with intra-bus bundle replacement).

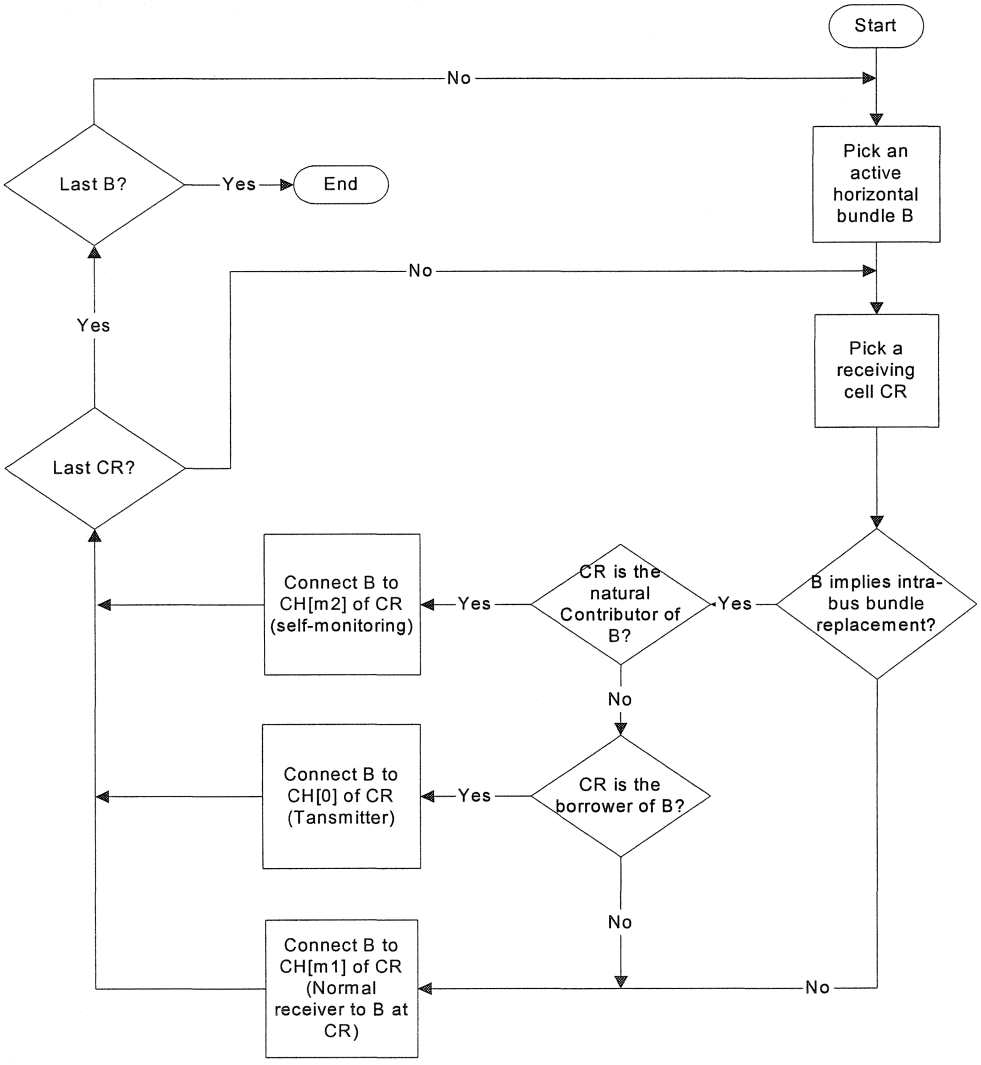


Figure 3.25 Flow chart of horizontal bundle transformation

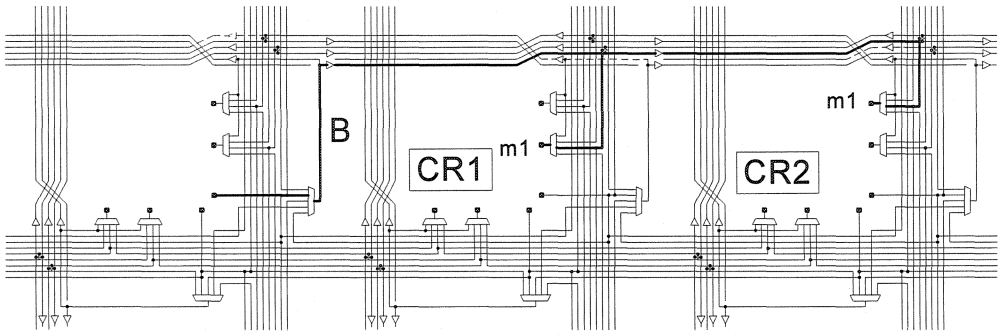


Figure 3.26 Example of Pointer m1

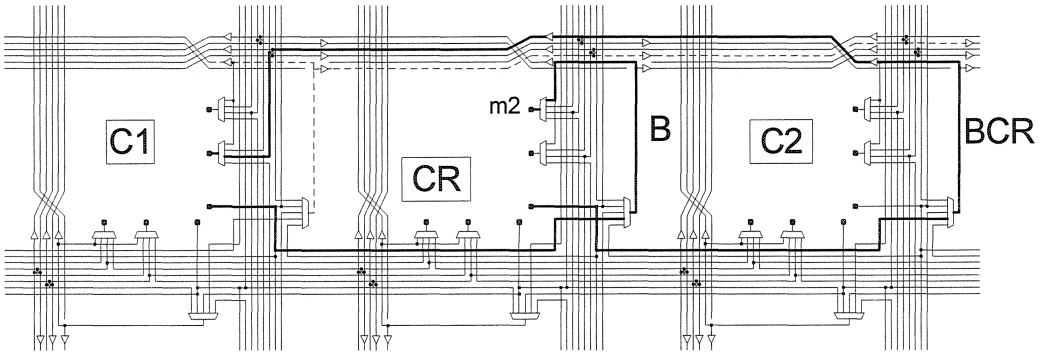


Figure 3.27 Example of Pointer m2

The pseudo-code presented in Figure 3.28 implements the transformation algorithm forming an MxN array from an (M+m)x(N+n) array. In the pseudo-code, line 1 to line 21 define the data structure to represent cells, bundles and their interconnections, line 23 to line 55 describe the transformation of horizontal bundle connections, line 56 to line 79 deal with the transformation of vertical bundle connections. For horizontal connections, line 38 to line 49 deal with intra-bus bundle shift, line 50 deals with natural connection or inter-bus bundle replacement. It is similar for vertical bundle connections. Note that the operation $x = ||a - b||c$ is equivalent to $x = a-b$ if $a - b \geqq 0$, $x = c - |a - b|$ if $a - b < 0$.

```
1    Index {
2      HI;                  // Index for horizontal dimension
3      VI;                  // Index for vertical dimension
4    }
5    Cell {
6      Index P;             //Physical index
7      Index L;             //Logical index
8      Index CH[M+m-1]; //Horizontal connections
9                           //CH[0]  for  Transmitter Tran_H
10                          //CH [1] for R1_H
11                          //CH [x] for Rx_H
12     Index CV[N+n-1]; //Vertical connections
13                          //CV[0]  for  Transmitter Tran_V
14                          //CV [1] for R1_V
15                          //CV [x] for Rx_V
16     }
17     Bundle{
18       int D;
19       Index P;          //Physical index
20       Index L;          //Logical index
21     }
```

Figure 3.28 Pseudo code of the transformation algorithm

```
22      Start transform() {
23       for each horizontal bundle B (which is connected to cell CT)
24       //Target at all the horizontal bundles used by logical cells
25          for x=0 to (M-1){  //x is the range of horizontal logical indexes of cells
26          //Target at all receiving cells listening to bundle B
27.             Let us consider the receiver of cell CR which CR.L.HI=x
28              and CR.L.VI=CT.L.VI;
29              // The cells that are located in the same logical row with CT
30              // CR.L.HI is the horizontal logical indexes of the receiving cell CR
31              //CR.L.VI is the vertical logical indexes of receiving cell CR
32              // CT.L.VI is the vertical logical indexes of talking cell CT
33              int m1=||B.P.HI-CR.P.HI||(M+m);
34              //Pointer to the receiver which B is talking to
35              int m2=||B.P.HI-CT.P.HI||(M+m);
36              //Pointer to the receiver associate to the bundle
37              //which cell CT  is talking to
38              if (|B.P.HI-CT.P.HI|==1) {
39              //There is a intra-bus bundle shift
40                  if (CR.P.HI == B.P.HI ) CR.CH[m2]=B.P;
41                  //The current CR is the contributor of bundle B
42                  elseif(CR.P.HI==CT.P.HI)  CT.CH[0]=B.P;
43                  //The current CR is the one which borrowed bundle B
44                  else CR.CH[m1]=B.P;
45                  //The cells, except for the cell of the contributor of bundle B
46                  //and the cell which borrows bundle B,
47                  //listen to B always with the same receiver,
48                  //no matter who is connected to B
49              }
50              else CR.CH[m1]=B.P;
51              // From receiver point of view,
52              //the physical HI distance between B and CR determine
53              //which receiver at cell CR listen to B
54          } End for each listening cell
55       } End for each horizontal bundle
```

Figure 3.28 (continue) Pseudo code of the transformation algorithm

```
56      for each vertical bundle B (which is connected to cell CT){
57        for y=0 to (N-1){                    //Range of logical index
58        Let us consider the receiver of cell CR which CR.L.VI=y
59        and CR.L.HI=CT.L.HI;
60        // The cells that are located in the same logical colomn with CT
61        int n1=||B.P.VI-CR.P.VI||(N+n);
62        int n2=||B.P.VI-CT.P.VI||(N+n);
63        if (|B.P.VI-CT.P.VI|==1) {  //There is a intra-bus bundle shift
64          if (CR.P.VI == B.P.VI ) CR.CV[n2]=B.P;
65          //The current CR is the contributor of bundle B
66          elseif(CR.P.VI==CT.P.VI)  CT.CV[0]=B.P;
67          //The current CR is the one which borrowed bundle B
68          else CR.CV[n1]=B.P;
69          //The cells, except for the cell of the contributor of bundle B
70          // and the cell which borrows bundle B,
71          //listen to B always with the same receiver,
72          // no matter who is connected to B
73        }
74        else CR.CV[n1]=(i,j);
75          // From receiver side of view, the physical VI distance
76          // between B and CR determine
77          // which receiver at cell CR listen to B
78        } End for each listening cell
79      }End for each vertical bundle
80    }End transform
```

Figure 3.28 (continue) Pseudo code of the transformation algorithm

The pseudo-code has been executed on the configuration of Figure 3.24. The first bundle to be processed is B[H,(1,0)] and the first receiving cell CR is C(1,0). In this case, index m1 = 0 and index m2 = 0. The resulting connection is CH[0] = (1,0), which means cell C(1,0) is connected to the horizontal bundle of B[H,(1,0)]. For the second CR, C(2,0), m1 = ||1-2||3 = 2 and m2 = 0. The resulting connection is CH[2] = (1,0), which means cell C(2,0) is listening from cell C(1,0) via receiver CH[2]. Receiver connections of other

bundles can be processed accordingly. By using the algorithm, the generation of receiver connections can be performed effectively.

3.5.2 Connectivity Based Configuration Rules

A set of connectivity-based configuration rules is proposed. These rules are used for determining configurations at the proposed simplified abstraction level. The feasibility of the configuration is ensured by the structure of the connections if there is no violation of the rules.

Rule 1, Cell Replacement Offset Rule

As shown in Figure 3.29, a cell can directly replace only one of its 8 nearest neighbors.

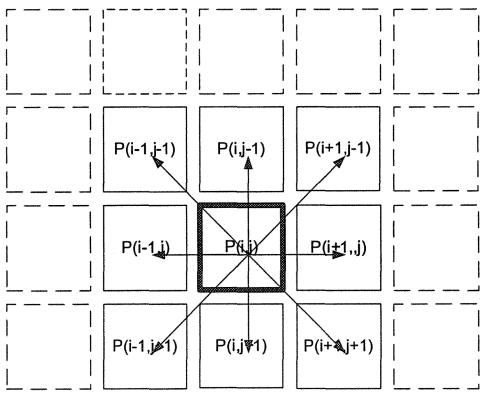


Figure 3.29 Cell Replacement Offset

Rule 2, Transmitter Connectivity Rule

A cell can be physically connected to the transmitters of 5 cells (its own and those of the up, down, left and right neighbor cells). As shown in Figure 3.30, the horizontal transmitter of the filled cell can only be connected to one of the 5 filled bundles.

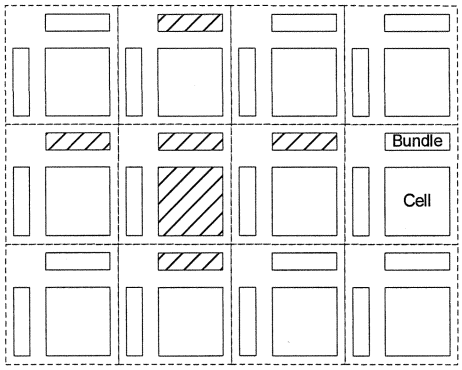


Figure 3.30 Transmitter Connectivity rule

Rules 3, Receiver Connectivity Rule (Row and Column Integrity Rules)

The purpose of introducing this rule is to ensure that a group of cells forming a complete logical row or column is able to communicate together. For example, to form a logical row, each cell of the group should be able to listen to all the horizontal bundles connected to the transmitters of all the other cells of the group. There are three cases in this category to apply these rules.

Case 1, all the cells of the group belong to one physical row j. The horizontal bundle connections of all the cells of this group must be within the bundles of the same physical row j and its upper & lower neighbor (j+1 and j-1) rows. Figure 3.31 shows an example of this case.

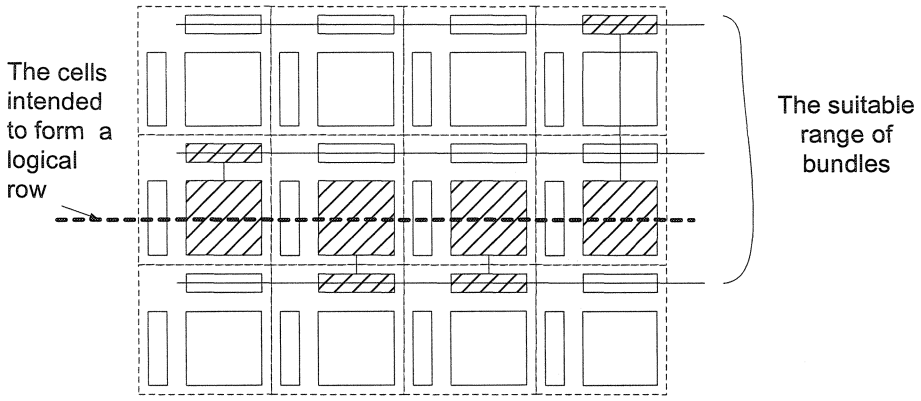


Figure 3.31 Target cells are from one physical row

Case 2, all the cells of the group belong to two neighboring physical rows (j and j+1). The horizontal bundle connections of all the cells of the group must be within the bundles of the two (j and j+1) physical rows. Figure 3.32 shows this case.

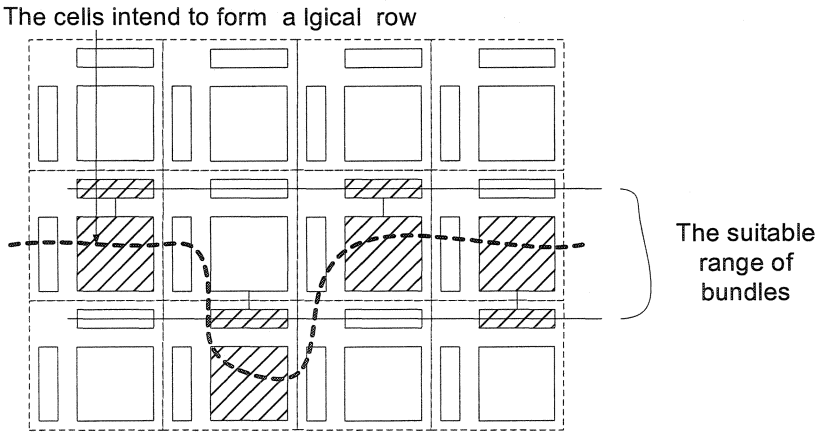


Figure 3.32 Target cells are from two neighboring rows

Details of a violation of this rule corresponding to the situation of Figure 3.32 are presented in Figure A.44 in Annex A.

Case 3, all the cells of the group belong to three neighboring physical rows (j-1, j and j+1) or the two outmost of the three neighbor rows (j-1 and j+1). The horizontal bundle

connections of all the cells of the group must come from the middle (j) physical row. Figure 3.33 illustrates this case.

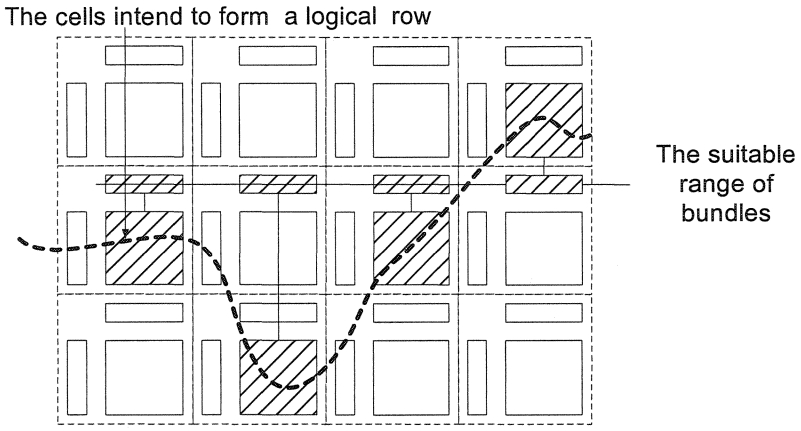


Figure 3.33 Target cells belong to three neighboring rows

The three cases discussed above are row constraints. Similar rules apply for columns, as illustrated in Figure 3.34.

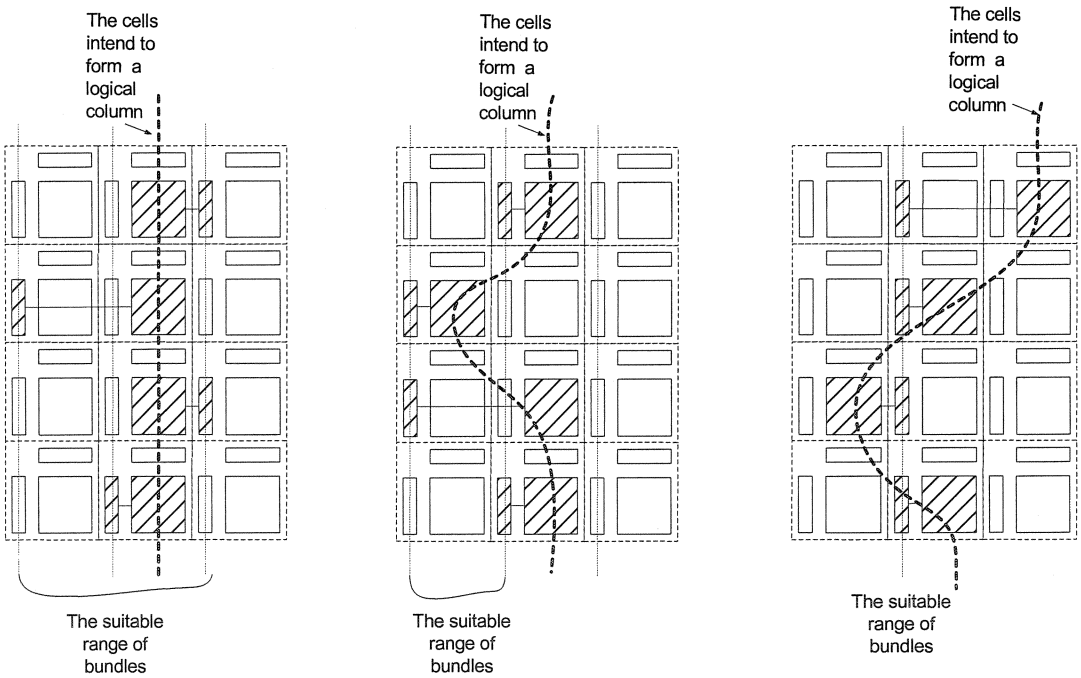


Figure 3.34 Constraints in column

(a) Group of cells belong to one physical column (b) Group of cells belong to two neighbor physical columns (c) Group of cells belong to three neighbor columns.

There is another constraint associated with the Receiver Connectivity Rules. As shown by the examples in Figure 3.35, when cell C1 borrows the horizontal bundle of cell C2, the vertical neighbors of cell C2, cell C2U and cell C2D cannot be brought into the same logical row with cell C1. There is not enough bypassing links to support those configurations. However, by adding a small number of bypass links, the above constraints can be waived. Such a structure with added bypass links to waive those constraints is defined as the type E connections. The comparison between the type D (the original pseudo-5-source connections) and type E connections, in terms of control and wiring overhead, is summarized in Table 3.20. It is indicated that the cost is only a small number of additional wiring and control overhead to waive significant constraints. It can be beneficial to choose the more versatile version for real implementations. As the work related to this constraint removal method was done at a late stage of this thesis, the analyses of type E connections was not incorporated in the thesis.

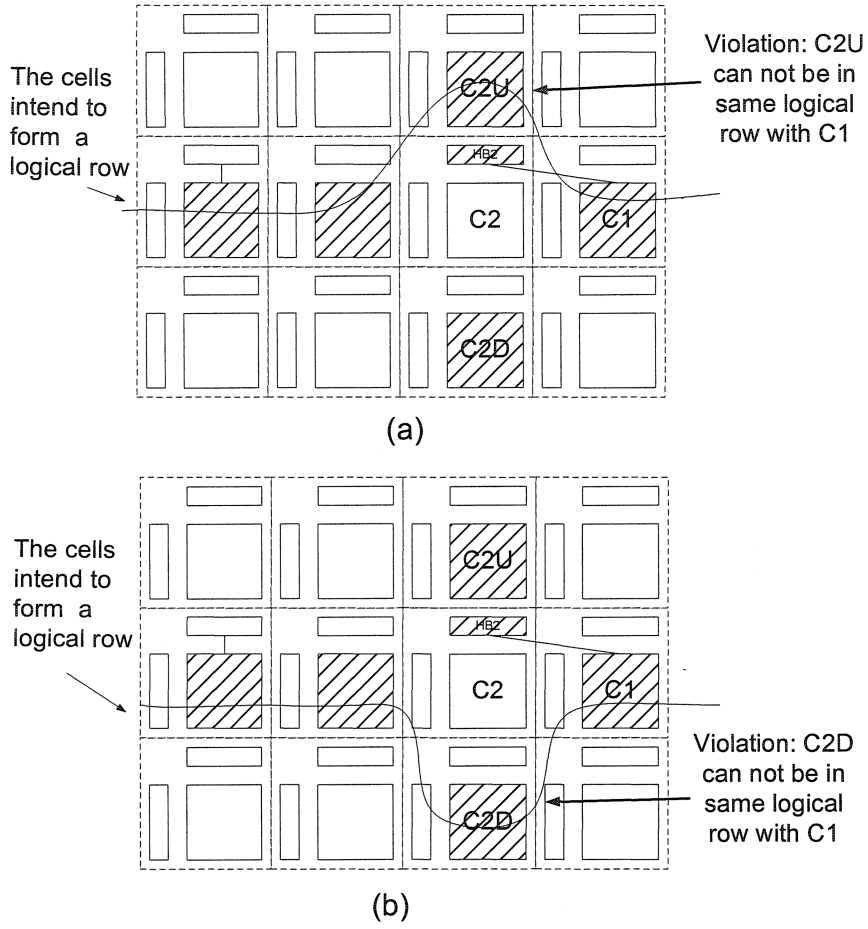


Figure 3.35 Intra-bus bundle borrowing caused connectivity loss

Table 3.20 Comparison of control and wiring overhead

Connection types		D	E
Bits to control	*N* * *N* array *x* lines/ bundle	*6*N*	*6*N+4*
	20 * 20 array **19** lines/ bundle	*120*	*124*
Bypass wires / bus	*N* * *N* array *x* lines/ bundle	*2*x*(N+1)*	*2*x*(N+2)*
	20 * 20 array **19** lines/ bundle	*1680*	*1760*

Rule 4, Source Exclusive Receiver Rule

The cells in the same logical column should not connect to two or more than two vertical bundles that are physically located in the same row. Similarly, the cells in the same logical row should not connect to two or more than two horizontal bundles that are physically located in the same column. The violation of this rule shown in Figure 3.36 would force the same receiving multiplexer to receive two distinct signals, while only one can be selected. The detailed description of the conflict implied by Figure 3.36 is provided in Figure A.42.

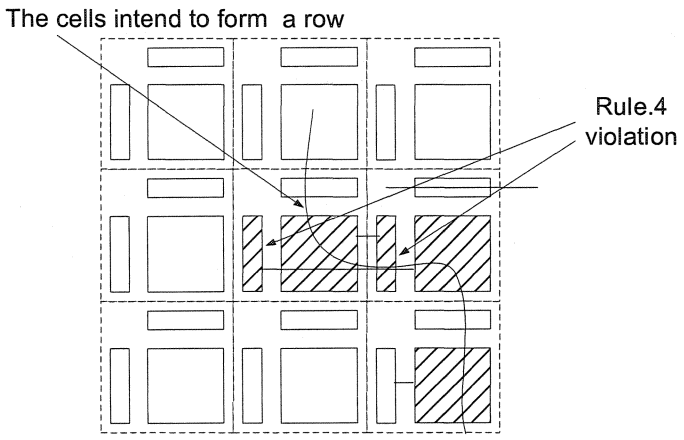


Figure 3.36 Source Exclusive Receiver Rule

Rule 5, Block Integrity Rule

One fault-free cell and two fault-free bundles, one horizontal and one vertical, can make a valid block only if the bundles are assigned exclusively to the cell, and the connections are done without violating the Transmitter Connectivity Rule. Figure 3.37 shows examples of two valid blocks

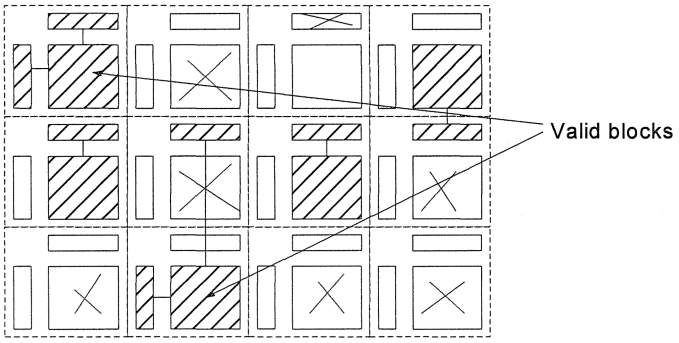


Figure 3.37 Block Integrity rule

Rule 6, Exclusive Logical Index Rule

Only one logical index can be assigned to a physical cell.

Rule 7, Logical Array Integrity Rule

A target logical array is properly formed if all logical cells are associated with valid blocks, all the active cells (cell is good and used) pass the Exclusive Logical Index Rule check, and all logical rows and columns pass the Receiver Connectivity Rule check and the Source Exclusive Receiver Rule check.

3.6 Reparability Analysis

Applying the connectivity-based configuration rules with the simplified abstraction, we can further evaluate the reparability of the array. The reparability is closely related to the amount of redundancy. Killer patterns that make the array not repairable tend to be more complex as the amount of redundancy increases. Before presenting a general analysis, it is convenient to start with an example where there is 1 column of redundancy.

3.6.1 Tolerable Patterns for 1 Column Redundancy

Figure 3.38 shows how several clustered fault patterns are tolerated with an array that has 1 column of redundancy. Figure 3.38 (a), (b), (c) and (d) show how several clustered fault patterns with 7 faulty cells can be repaired in a 7 x 8 array. Figure 3.38 (e) and (f) show how two clustered patterns can be repaired by composing a logical column with cells from 3 physical columns.

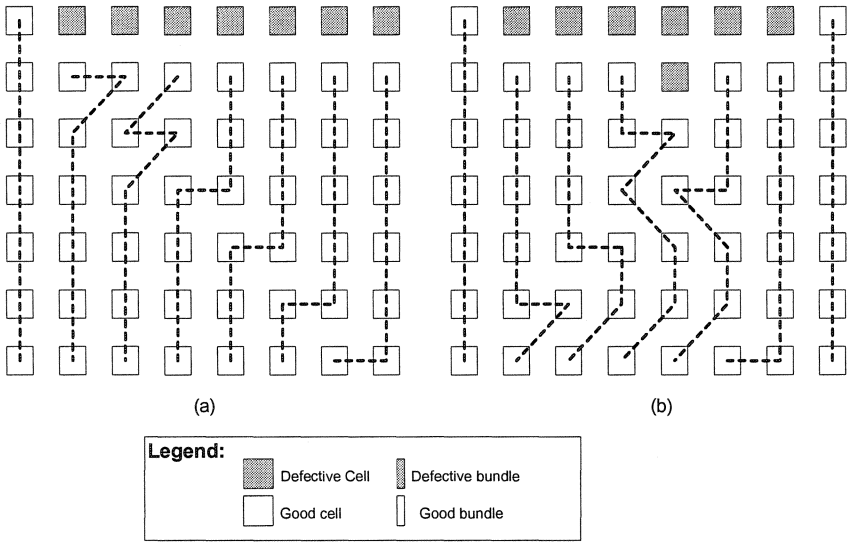


Figure 3.38 Several tolerable patterns of 1 column redundancy

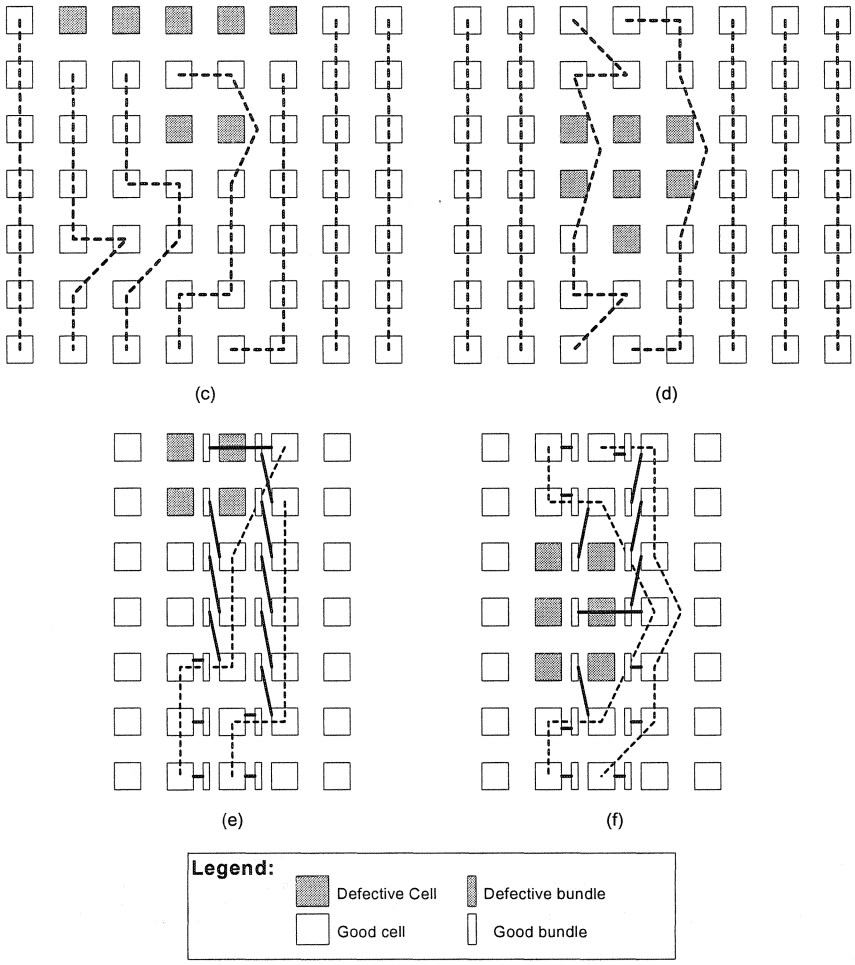


Figure 3.38 (continue) Several tolerable patterns of 1 column redundancy

3.6.2 General Reparability Analysis

The reparability of the fault tolerant structure is strongly related to the amount of redundancy. As the redundancy increases, more fault patterns become repairable, and the unrepairable patterns become larger. The yield can be improved since the probability for the bigger fault patterns is drastically smaller than that for small fault patterns.

However, the effectiveness of increasing the redundancy to improve reparability can be limited by the array size and the amount of redundancy. It becomes more significant as the size of the array and redundancy gets bigger that the target array need to be distorted too much to pass connectivity rules (Rule.3 and Rule.4 of section 3.5.2) checks.

In addition to the case where the number of spare elements is exhausted, configuration may fail when some configuration rules are violated. This is due to un-repairable combinations of cells and bundles.

Figure 3.39 shows some of the cases associated with the case where one column of redundant cells is available. Pattern (a) is an example of the situation where there are two defective vertical bundles in a row. Pattern (b) is associated with the situation that there are 3 defective cells in a corner and there is a vertical defective bundle in the associated corner row. Patterns in (c) and (d) are associated with the situation that a square of 4 defective cells is at the top or bottom of the array, and two defective bundles are in 2 distinct columns selected from 4 specific columns (the two columns associated with the defective cells, and the left neighbour column and right neighbour column of the defective cell pattern). Patterns in (e) and (f) are associated with the situation where a rectangle of 6 defective cells (in 2 columns and 3 rows) is at any location except for the boundary of the array, and two defective bundles are in 2 distinct columns selected from 4 specific columns (the two columns associated with the defective cells, and the left neighboring column and right neighboring column of the defective cell pattern). Pattern (g) is associated with the situation where a square of 4 defective cells is at the top or the bottom of the array and the 2 neighbour cells at the associated boundary row are also defective. Pattern (h) is associated with the situation where a rectangle of 6 defective cells (in 2 columns and 3 rows) is at any location except for the boundary of the array and the 2 neighbour cells in the associated middle row are also defective. Pattern (i) is a variation of patterns (g) and (h) where the 4 defective cells of the pattern are in a corner.
The detailed analysis of patterns (a) and (b) is shown in [44]. As indicated by the Receiver Connectivity Rule (Rule.3) of section 3.5.2, the cells in one logical column

must come from 3 or less neighboring physical columns. To repair the 4-square or 6-rectangle fault pattern, at least one logical column has to be composed of cells from 3 physical columns, as shown by the partial solutions of (e) (f) in Figure 3.38. The bundle connections of the two columns shown are mandatory. The defective bundles in Figure 3.39 (c), (d), (e), and (f) prevent the construction of the logical columns. The defective cells showed with bold lines in Figure 3.39 (g), (h) and (i) prevent the construction of the logical column by the additional faulty cells due to similar reasons.

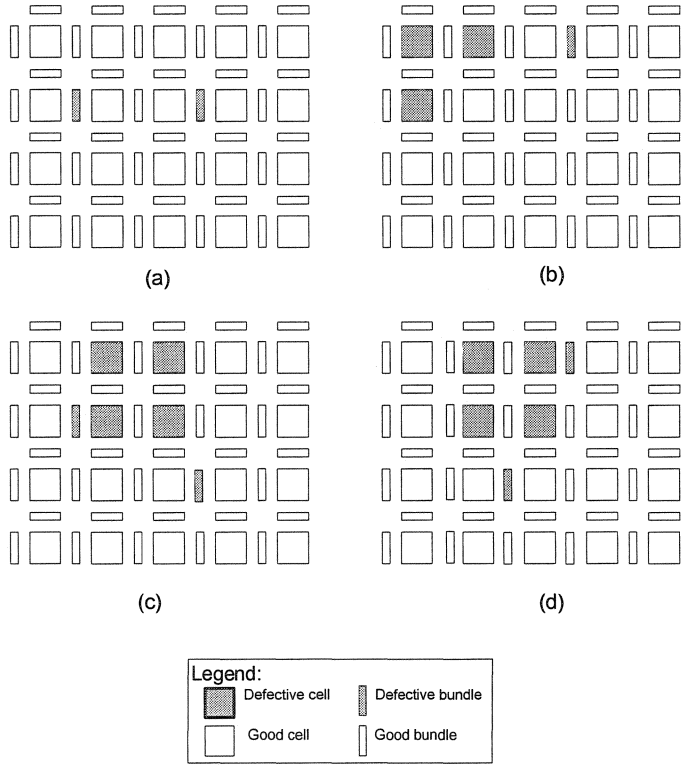


Figure 3.39 Several intolerable patterns of 1 column redundancy

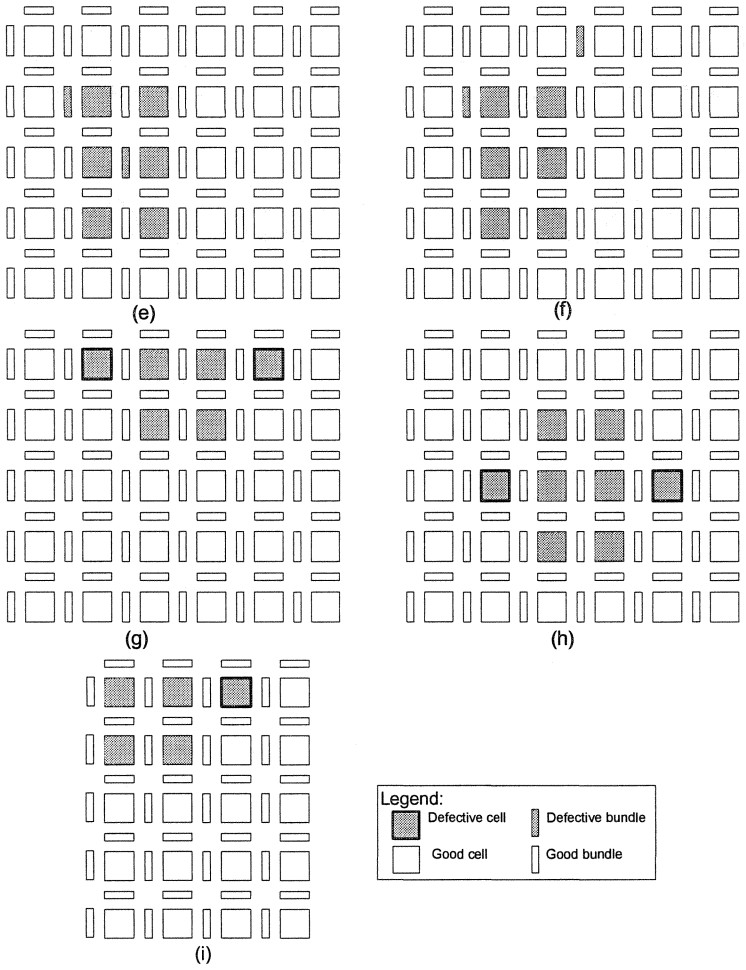


Figure 3.39 (continue) Several intolerable patterns of 1 column redundancy

With an increased redundancy, fault patterns with a bigger number of faulty elements can be repaired, and the intolerable patterns become more complex and contain even more elements. The pattern in Figure 3.40 is the simplest intolerable pattern when the array has one redundant column and one redundant row. One logical column must be composed with the cells from the first 3 columns in the left of the array. According to Receiver Connectivity Rule (Rule.3), the suitable bundles must come from one of the following alternatives, the first column in the left, the second column in the left, the 2 columns in the left, or from the 3 columns in the left. According to Source Exclusive Receiver Rule (Rule.4), all the vertical bundles in a logical column should be in distinct physical rows.

Unfortunately, for a logical column, enough vertical bundles belonging to distinct rows cannot be found in the two columns. Thus, this pattern prevents the construction of a logical column using the cells from the two or three columns in the left.

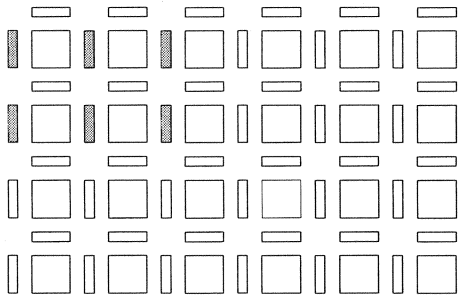


Figure 3.40 Simplest intolerable pattern for 1 column and 1 row redundancy

In addition to the cases described above, the unreparable and unusable cells significantly degrade the ability of the array to form a target logical array. A cell is unreparable if it has 8 defective neighbours, as shown in Figure 3.41 (a). A cell is unusable, if it has a defective vertical bundle and all its 4 (upper, bottom, left and right immediate) neighbours (a total of 5 cells) have defective vertical bundles as shown in Figure 3.41 (b). A similar situation exists for horizontal bundle.

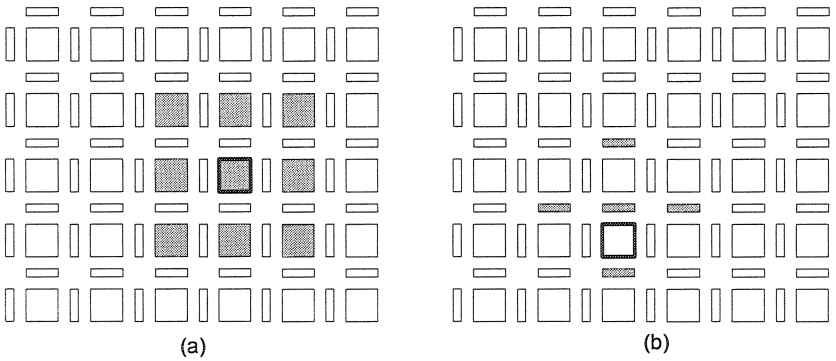


Figure 3.41 (a) Unrepairable cell (b) Unusable cell

3.7 Configuration Algorithms

3.7.1 Existing Approaches

Configuration algorithms are used to determine valid restructuring interconnections of a circuit for certain fault distributions. Numerous configuration algorithms have been developed to deal with different constraints and different fault distributions. Among some of these schemes summarized by Negrini, Sami, and Stefanelli [33], two approaches are applicable to the structure proposed in this thesis.

1) Graph-theoretical Approaches

The majority of the reconfiguration criteria that belong to the graph theoretical class involve the use of complex algorithms that are costly to implement by means of dedicated on-chip circuitry. They are thus more suitable for end-of-production restructuring or, possibly, for host-driven reconfiguration. A useful tool from graph theory, adopted for rectangular physical arrays, consists of so-called folding procedures applied to trees of meshes utilized to represent arrays. Spanning trees are further tools derived from graph theory. A spanning tree of a graph G is defined as a sub graph containing all the nodes of G and a subset of its edges such that there is exactly one path between each pair of nodes. Weights can be associated with edges and with paths in a spanning tree, so that optimisation criteria can be easily defined.

2) Global Mapping Approaches

An entire class of reconfiguration techniques can be described on the basis of one unifying philosophy. Such approaches consider reconfiguration as mapping of array functions represented by logical indexes associated with a suitable set of working cells in the array. Thus, in a way, reconfiguration is achieved by deforming the initial array, circumventing faulty cells and distorting inter-cell connections. Algorithms are then identified by rules creating suitable mapping that respects constrains imposed by the physical array on distorted links.

Since it can be easily integrated with configuration rules, the Global Mapping approach is adopted by the Greedy Configuration Procedure, which is proposed in this thesis and presented in section 3.7.3. However, it is also feasible to apply the Graph-theoretical approach on the structure shown in Figure 3.17.

3.7.2 Investigation of the Approaches for the Pseudo-5-sourse Structure

Applying the configuration rules at the simplified abstraction level, configuration algorithms are used to determine detailed connections of transmitters and to ensure that the receivers can be connected appropriately, i.e. the configuration algorithm can produce a global mapping of logical indexes onto the physical indexes. After the transmitters are configured, the corresponding receivers can be configured according to the logical and physical indexes of cells and their associated connections to bundles by the transform algorithm.

One of the features of the structure is its flexible bundle replacement. Three approaches of cell mapping and bundle mapping in different ways are discussed.

3.7.2.1 Exhaustive Searching Approach

With this approach, all the combinations of cell assignments and bundle assignments are checked one after another, till a valid configuration is found. It may be interesting to estimate how many combinations exist, which is presented below.

Specifying a target logical array can be done by specifying redundant rows and columns. After identifying the redundant cells from the physical array, target logical indexes can be assigned to all the positions of primary cells (no matter if they are good or faulty), as shown in Figure 3.42. To form an M x N logical array from an (M+m) x (N+n) physical

array, there are $\binom{M+m}{m}\binom{N+n}{n}$ ways to specify the redundancy. Each of these ways to specify redundancy define a possible target logical array.

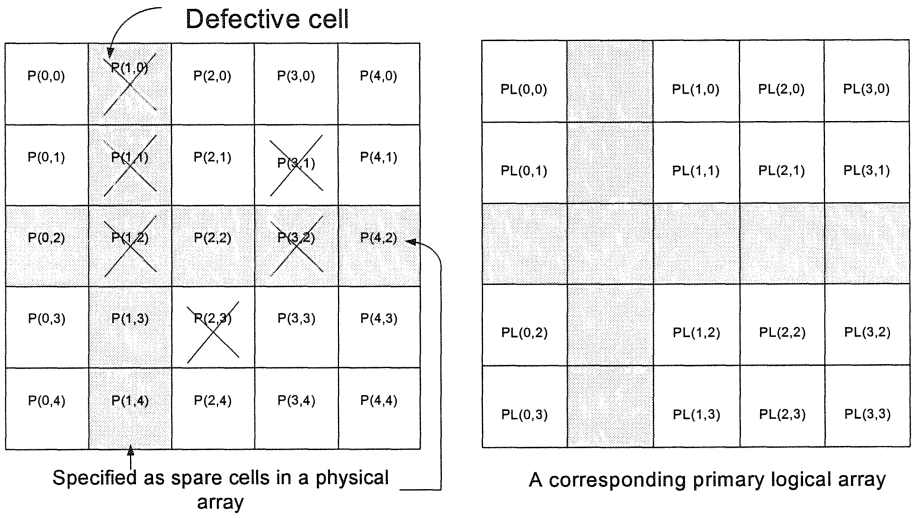


Figure 3.42 Physical array after specifying redundancy and its corresponding primary logical array.

The next step is the cell assignment phase for each target logical array. Except for the boundary conditions, any one of 9 physical cells (one natural cell and its 8 nearest neighbours) may be assigned to a logical position. For simplicity, we ignore the cases in which the same physical cell may be assigned to more than one logical position. The upper bound on the number of combinations for cell assignment of a target array could be estimated as $9^{(MN)}$.

For the transmitter connections, each transmitter may be linked to 5 bundles (its natural bundle and the bundle of is 4 physical neighbouring cells) and each cell owns 2 bundles. So far, there are at most $9^{MN}25^{MN}\binom{M+m}{m}\binom{N+n}{n}$ combinations of interconnections. These numbers get very large for arrays of practical sizes.

After constructing a possible configuration, the following step is to check if it is valid. The following rules specified in section 3.5.2 should be checked at this stage:

- Block Integrity Rule
- Receiver Connectivity Rule
- Source Exclusive Receiver Rule

If the target array passes all the above rule checks, the configuration is valid.

3.7.2.2 Discrete Approach

Since the yield of cells is less than the yield of bundles and there is an additional degree of freedom in bundle replacement (intra-bus bundle replacement), the mapping of cells may be more critical than the mapping of bundles. To reduce the search space, all the cell replacements can be done (assuming that all bundles are defect-free) before bundle routing. Since cell replacement is done with all bundles assumed defect-free, several existing algorithms could be applied. Thus, several approaches, such as the replacement approach described in Section 3.4, graph-theoretical approaches, and global mapping approaches can be taken for cell replacement. After the configuration of the cells, bundles are connected to the active cells following the configuration rules.

3.7.2.3 Integrated Approach

In this approach, the configuration rules are incorporated with cell assignments and bundle assignment. According to the configuration rule analysis, in the process of cells and bundle assignments, the current assignments of cells and bundles are constrained by the cell and bundle assignments done in previous steps. If bundle assignment and cell assignment are performed at the same time, the combinations that contain violations of configuration rules can be eliminated immediately. Among the valid combinations, the cell and bundle assignments are done according to the procedure implementing the configuration algorithm.

3.7.2.4 Summary of the 3 approaches

There is a trade-off between survival rate and configuration algorithm complexity. The Exhaustive Searching Approach provides a 100% survival rate but with a computational complexity that may be difficult to handle if the vast majority of candidate configurations fail to yield a valid array. Thus, to facilitate the computation task of the configuration procedure, a good heuristic search scheme is needed. The Discrete Approach may be integrated with existing heuristic search schemes, for example, it can be combined with the graph based analysis and the global mapping approach, which may be suitable for end-of-production restructuring. The Integrated Approach is the fastest among the three, since the bundle and cell assignments are processed at the same time, resource conflicts (the conflicts among bundle assignments and cell assignments caused by configuration rule violations) can be eliminated immediately. Moreover, the procedure can attempt targeting the most likely heuristics first. Because of its significant advantages, the integrated approach is adopted for the Greedy Configuration Procedure described below.

3.7.3 A Greedy Global Mapping Configuration Procedure

A greedy discrete assignment global mapping algorithm is proposed in this thesis. The strategy of this algorithm is to keep as many available elements as possible in the current assignment to maximize the flexibility of the following assignments, and to make all the computation local in order to minimize computational complexity. The details are described in the following.

3.7.3.1 Selecting A Possible Logical Array

Before mapping from a physical array to a logical array, we need to specify a possible target logical array, i.e. specifying the redundant cells. The redundant cells should preferably be specified in such a way that all of them can be used as a replacement to a

cell in the target logical array by a single basic operation (shift of one position). Otherwise, some available spares would not be reachable for the considered logical array, and it is conjectured (not proved) that a logical array that can use all spares is always better than one that cannot reach some of them. Four examples are given in Figure 3.43 for specifying a logical array that corresponds to 3 types of redundancy. When an array has one column of redundancy, we can specify logical array as that in Figure 3.43 (a). For one column and one row of redundancy, we can specify logical array as (b1) and (b2). For two columns and two rows of redundancy, we can specify logical array as (c). We can generate various logical arrays for different levels of redundancy. For instance, with one column and one row of redundancy, as shown in Figure 3.44, we can specify other two logical arrays that are located at the top left and top right of the physical array.

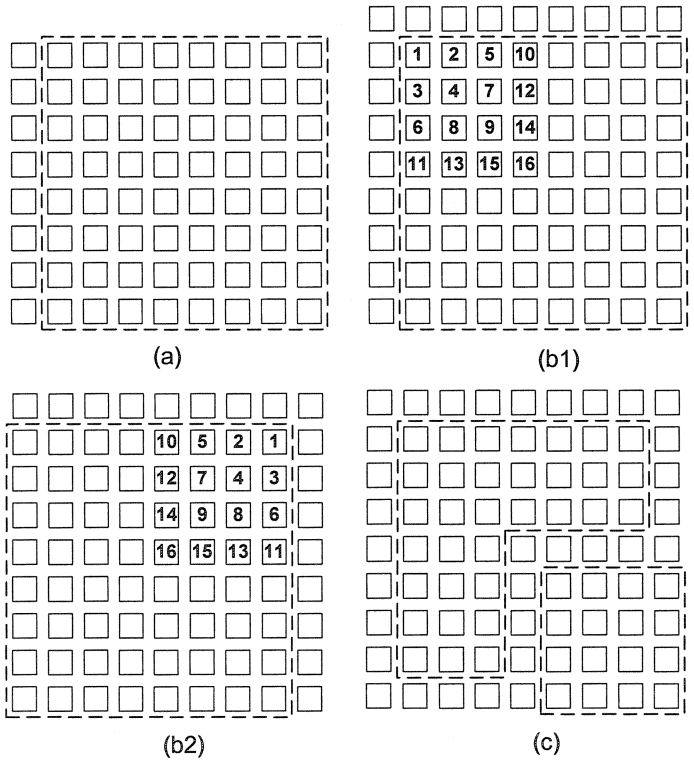


Figure 3.43 Specifying a possible logical array

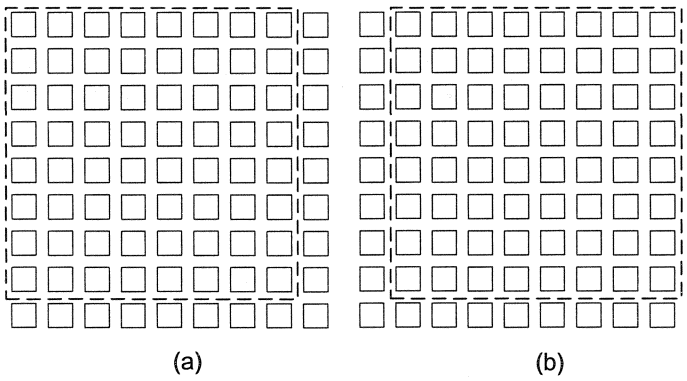


Figure 3.44 Additional possible logical arrays

3.7.3.2 Alternative Assignments

As shown in Figure 3.45, any one of the 9 neighbouring physical cells can be assigned to a logical cell, and any one of 5 bundles can be connected to the transmitter of a cell. It is obvious that this is not a directionally biased structure, since one cell can be replaced by anyone of its neighbours in all directions, and a cell can replace any of its neighbours in all directions.

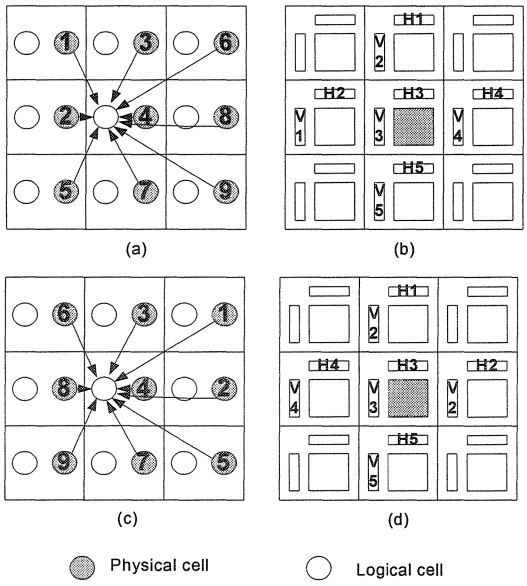


Figure 3.45 Domain of assignments and mapping priority

3.7.3.3 Perspective on PaP and PeP

Parallel placement (PaP) is to compose a logical column (row) using the cells physically located at different row (column), as shown in Figure 3.46. Perpendicular placement (PeP) is to compose a logical row (column) using the cells physically located at the same column (row), as shown in Figure 3. 46.

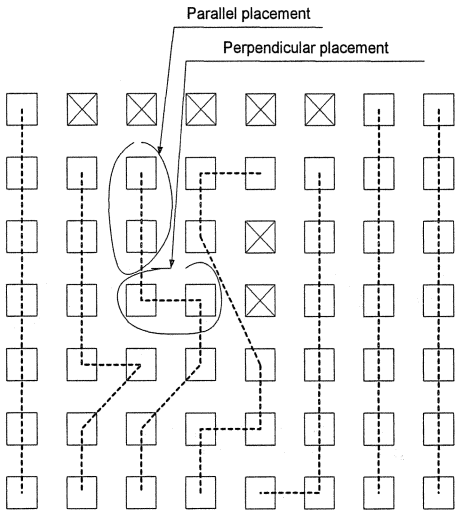


Figure 3.46 Parallel and perpendicular replacement

As shown in Figure 3.47, (a1) and (a2) are two examples of PeP and (a3) is an example of PaP. The cases shown in Figure 3.47 (b1), (b2) and (b3) are the bundle connections corresponding to the placement of (a1), (a2) and (a3) respectively. It is obvious that PeP forces some bundle connections, but PaP does not. To preserve the flexibility for bundle assignments, PaP should always be first considered.

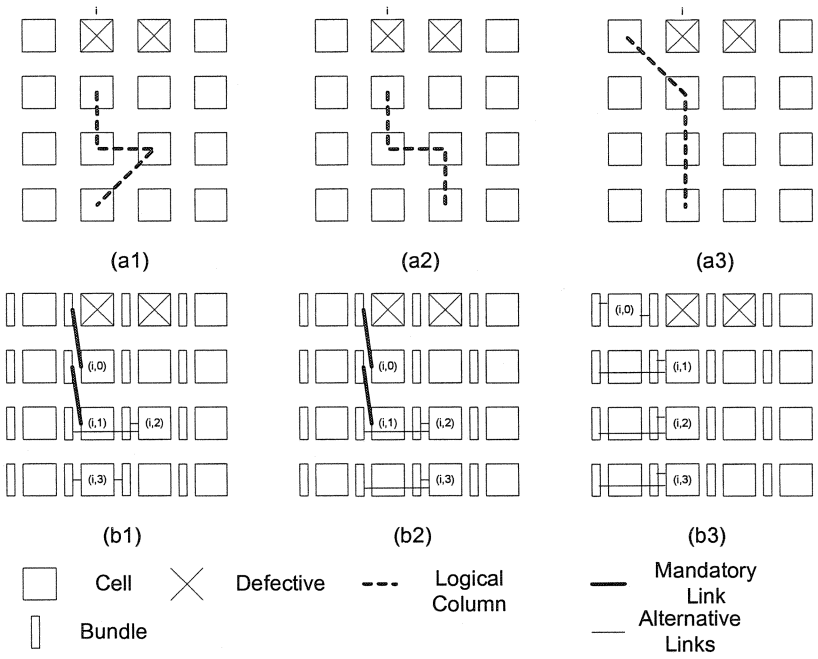


Figure 3.47 Pep and Pap vs. logical index and bundle connection

3.7.3.4 Priority and Sequence of Assignment

A proposed order to process cells from the target logical array in Figure 3.43 (b1) and Figure 3.43 (b2) is proposed as labels on the cells. When a cell is processed, the order for assigning physical cells to the correct target logical position is proposed (labels) in Figure 3.45 (a) and Figure 3.45 (c) respectively. There are 25 combinations of bundle assignment. A possible order of priority is as follows: H1V1, H1V2, H2V1, H2V2, H3V1, H1V3, H3V2, H2V3, H3V3, H4V1, H1V4, H4V2, H2V4, H4V3, H3V4, H4V4, H5V1, H1V5, H5V2, H2V5, H5V3, H3V5, H5V4, H4V5, H5V5. That order applies both to Figure 3.45 (b) and (d). Note, the difference is labelling between the two figures, where Figure 3.45 (b) defines the order for Figure 3.43 (b1) and Figure 3.45(d) defines the order for Figure 3.43(b2).

One criterion to define the priorities is to preserve the resources for the assignments to

come. For the case shown in Figure 3.43 (b1), if we make assignments according to the specified priorities, the elements close to the up-left corner are always used first. As a consequence, the modules remain available for the following assignments. Another criterion is to avoid PePs. One reason to put the priority of the assignment of 2 and 3 (shown in Figure 3.45) higher than the priority of assignment 4 is that assignment 4 is more likely to lead to PePs.

3.7.3.5 The Flow Chart

The flowchart shown in Figure 3.48 describes at high level the procedure of the configuration algorithm. The configuration succeeds if all the logical cells were assigned to physical cells, all active cells were linked to two bundles and all the assignments passed configuration rule checks. The configuration fails if there is no alternative assignment left for a failed assignment.

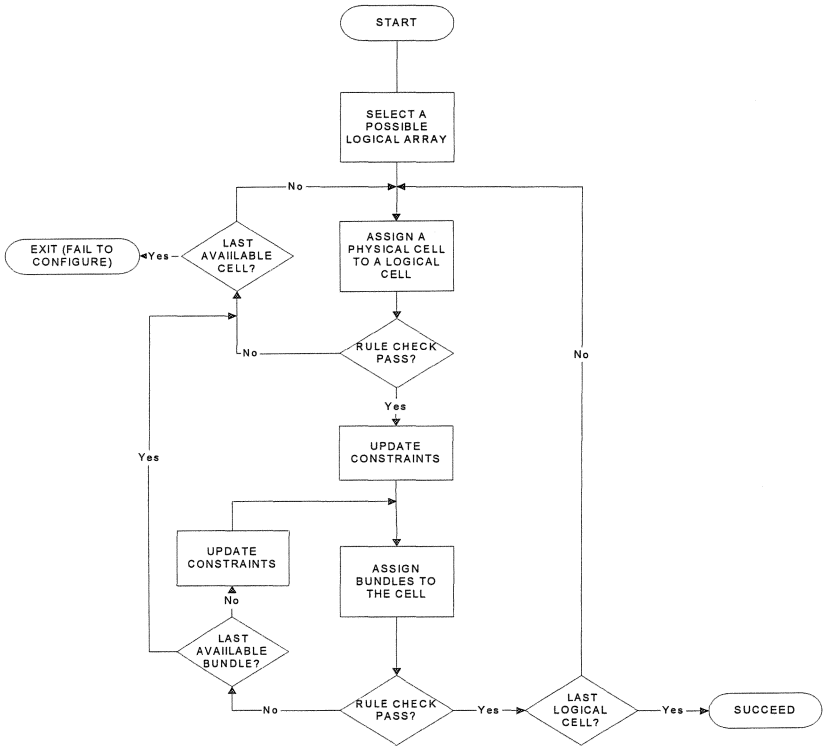


Figure 3.48 Flow chat of the greedy configuration procedure

The configuration procedure follows a rigid order. There is no look ahead and no backtrack. These shortcomings may suggest that the procedure may fail easily if there is a cluster of defect that requires bending logical rows and columns. Nevertheless, based on several experiments, it was found to be quite robust. Figure 3.49 shows an example, where a clustered defective pattern is repaired by manually applying the procedure.

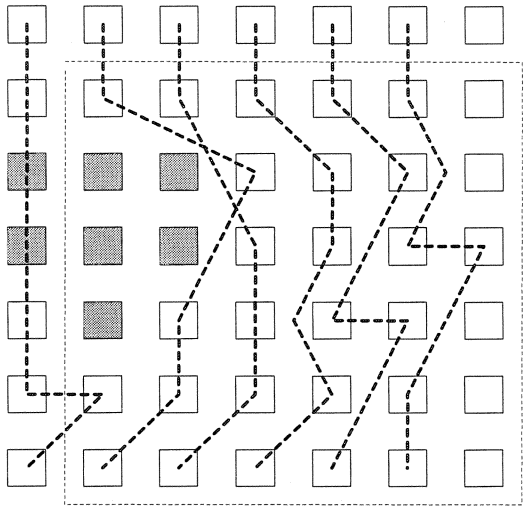


Figure 3.49 Clustered fault pattern repaired by the procedure

3.7.3.6 Rule Check Computation

Every assignment puts constraints on following assignments. For a valid assignment, configuration Rule 3, Rule 4, and Rule 5 should be followed. Rule 5 and a part of Rule 3 can be checked after cell assignments. Rule 3 and Rule 4 need to be checked after bundle assignments. For a logical row, the existing vertical range of physical cells defines the available vertical range of subsequent bundle assignments, and the existing vertical range of horizontal bundles defines the available vertical range of the subsequent cell assignments, as shown in Figure 3.50. It is similar for a logical column.

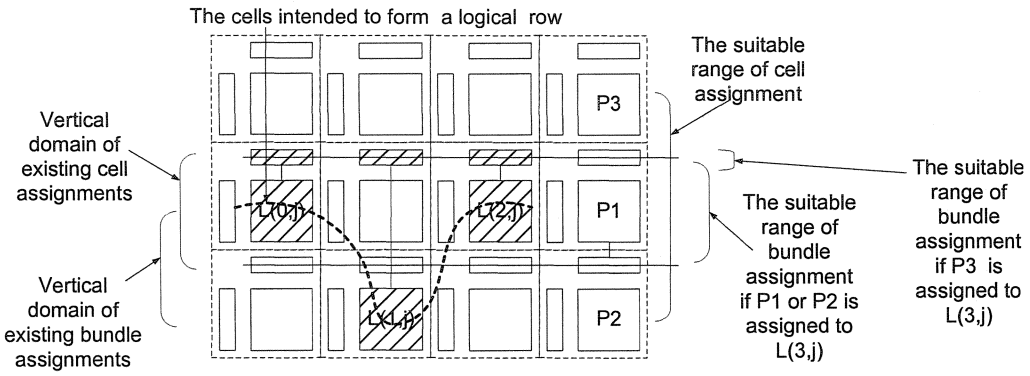


Figure 3.50 Existing domains and constrained domains

To minimize computational complexity, all the rule checks are executed locally. As configuration rule checks of one assignment need the information of some of the exiting assignments, a set of global variables is defined to make the rule check faster. The global variables record simplified existing assignment information that applies tightest constraints to subsequent assignments. Table 3.21 and Table 3.22 define the global variables. The data structure for rule check is defined in Figure 3.51. The pseudo code of cell assignment rule check is present in Figure 3.52. The pseudo code of bundle assignment rule check is present in Figure 3.53.

The computational complexity for the rule check of any cell is the same. The same is true for bundle assignment. The total computational time could be proportional to the number of cells.

Table 3.21 Vertical domain for row assignment

CVmin(l)	The physical row index of the cell with minimum vertical physical index
CVmax(l)	Physical row index of the cell with maximum vertical physical index
BVmin(l)	Physical index of the bundle with minimum vertical physical index
BVmax(l)	The physical index of the bundle with maximum vertical physical index

Table 3.22 Horizontal domain for column assignment

CHmin(k)	The physical column index of the cell with minimum horizontal physical index
CHmax(k)	The physical column index of the cell with maximum horizontal physical index
BHmin(k)	The physical index of the bundle with minimum horizontal physical index
BHmax(k)	The physical index of the bundle with maximum horizontal physical index

```
1    Index{
2      HI;                  // Index for horizontal dimension
3      VI;                  // Index for vertical dimension
4    }
5    Cell{
6      Index P;             //Physical index
7      Index L;             //Logical index
8      Index HBC;           //Horizontal bundle connections
9      Index VBC;           //Vertical bundle connections
10     }
12   Bundle{
13     Index P;             //Physical index
14     Index L;             //Logical index
15   }
```

Figure 3.51 Data Structure

```
1  //rule 5 check
2  If P(i,j) is not good return fail;
3  //rule 3 check row
4  If (BVmax(l)-BVmin(l)=2)
5  //If bundles of a logical row from 3 physical rows
6      if (j!= BVmin(l)+1) return fail endif;
7      //Cells must form the mid physical row
8   else if(BVmax(l)-BVmin(l)=1)
```

Figure 3.52 Cell Assignment Check (assign P(i,j) to L(k,l))

```
9  //If bundles of a logical row from 2 physical rows
10      if((j!= BVmax(l)) && (j!=BVmax(l)) return fail endif;
11      //Cell must from the two physical rows
12  else if(BVmax(l)-BVmin(l)=0)

13  //If bundles of a logical row from 1 physical row
14      if((j > (BVmax(l)+1)) || (j<(BVmax(l)-1))) return fail endif;
15      //Cells must from 3 physical rows
16  endif;
17  if (|i - L(i-1,j).P.HI| == |j – L(i-1,j).P.VI | == 1 and L(i-1,j).HBC.HI == i)
18  //If the logical neighbour is physical diagonal neighbour
19  // and it borrowed (i,j)'s horizontal bundle
20      return fail;
21  endif;
22  //rule3 check column
23  If (BHmax(k)-BHmin(k)=2)      //Bundles from 3 columns
24      if (i!= BHmin(k)+1) return fail endif;  //Cell from one column
25  else if(BHmax(k)-BHmin(k)=1)    //Bundles from 2 columns
26      if((i!= BHmax(k)) && (i!=BHmax(k)) return fail endif;
27     //Cell from 2 column
28   else if(BHmax(k)-BHmin(k)=0)     //Bundles from 1 column
29      if((i > (BHmax(k)+1)) || (i<(BHmax(k)-1))) return fail endif;
30      //Cell from 3 column
31  endif;
32  if (|i - L(i,j-1).P.HI| == |j – L(i,j-1).P.VI | == 1 and L(i,j-1).VBC.VI == j)
33  //If the logical neighbour is physical diagonal neighbour
34  // and it borrowed (i,j)'s vertical bundle
35      return fail;
36  endif;
37  //update flags
38  if(j > CHmax(l)) CHmax(l)=j;
39  //If the current physical row index j > maximum index of l logical row,
40  //the maximum index of l logical  row will be updated as j
41      elseif(j < CHmin(l)) CHmin(l)=j;
42  endif;
```

Figure 3.52 (continue 1) Cell Assignment Check (assign P(i,j) to L(k,l))

```
43 //If the current physical row index j < minimum index of l logical row,
44 //the minimum index of l logical row will be updated as j
45 if(i > CVmax(k)) CVmax(k)=i;
46 //If the current physical column index i >
47 // maximum index of k logical column,
48 //the maximum index of k logical column will be updated as i
49      elseif(i < CVmin(k) CVmin(k)=i;
50 //If the current physical column index i <
51 //minimum index of k logical column,
52 //the minimum index of k logical column will be updated as i
53 endif
```

Figure 3.52 (continue 2) Cell Assignment Check (assign P(i,j) to L(k,l))

```
1  //rule5 check
2  if (VB(i,j) or HB(k,l) is not good) return fail endif;
3  //rule3 check row
4  if(CVmax(p)-CVmin(p)=2)   //Cells from 3 rows
5       if(l!=(CVmin+1)) return fail endif;   //H bundle from 1 row
6  elseif(CVmax(p)-CVmin(p)=1)     //Cells from 2 rows
7       if((l>CVmax) || (l<CVmin)) return fail endif;     //H bundle from 2 rows
8  elseif(CVmax(p)-CVmin(p)=0)   //Cells from 1 row
9       if((l>(CVmax+1)) || (l<(CVmin)-1)) return fail endif;
10      //H bundle from 3 row
11 endif;
12 if (|m - L(o-1,p).P.HI| == |n – L(o-1,p).P.VI | == 1 and L(o-1,p).P.HI == k)
13 //If the logical neighbour of P is a physical diagonal neighbour
14 // and HB is at the same column with the logical neighbour of P
15       return fail;
16 endif;
```

Figure 3.53 Bundle Assignment Check (assign VB(i,j) HB(k,l) to P(m,n) L(o,p))

```
17 //rule3 check column
18 if(CHmax(o)-CHmin(o)=2)    //Cells from 3 columns
19      if(i!=(CHmin+1)) return fail endif;  //V bundle from 1 column
20 elseif(CHmax(o)-CHmin(o)=1)  //Cells from 2 columns
21      if((i>CHmax) || (i<CHmin)) return fail endif;  //V bundle from 2 columns
22 elseif(CHmax(o)-CHmin(o)=0)  //Cells from 1 columns
23      if((i>(CHmax+1)) || (i<(CHmin)-1)) return fail endif;
24       //V bundle from 3 columns
25 endif;
26 if (|m - L(o,p-1).P.HI| == |n – L(o,p-1).P.VI | == 1 and L(o,p-1).P.VI == j)
27 //If the logical neighbour of P is a physical diagonal neighbour
28 // and VB is at the same row with the logical neighbour of P
29      return fail;
30 endif;
31 //rule4 check row
32 if(k.L(o,p) = k.L(o-1,p) || k.L(o,p) = k.L(o-2,p)) return fail endif;
33 // k.L(o,p) denotes column index of Horizontal bundle of cell L(o,p)
34 // k.L(o-1,p) denotes column index of Horizontal bundle of cell L(o-1,p)
35 // k.L(o-2,p) denotes column index of Horizontal bundle of cell L(o-2,p)
36 //rule 4 check column
37 if(j.L(o,p) = j.L(o,p-1) || j.L(o,p) = j.L(o,p-2)) return fail endif;
38 // j.L(o,p) denotes row index of Vertical bundle of cell L(o,p)
39 // j.L(o,p-1) denotes row index of Vertical bundle of cell L(o-1,p-1)
40 // j.L(o,p-2) denotes row index of Vertical bundle of cell L(o-2,p-2)
41 //update flags
42 if(i > BVmax(m)) BVmax(m)=i;
43      elseif(i<BVmin(m)) BVmin(m)=i;
44 endif;
45 if(l > BHmax(n)) BHmax(n)=l;
46      elseif(l<BHmin(m))BHmin=l;
47 endif;
```

Figure 3.53 (continue) Bundle Assignment Check

(assign VB(i,j) HB(k,l) to P(m,n) L(o,p))

3.7.3.7 Summary of the Greedy Configuration Algorithm

The Greedy Configuration Algorithm is a directional biased configuration algorithm on a non-directional configuration structure. Selecting a possible logical array directionally biases the logical array or a part of the logical array from a physical array by a difference within the range that a direct replacement can be performed. The assignments always try the far most cell in the opposite direction of the logical array bias first. As a consequence, the assignments with good likelihood according to the proposed heuristic processing order are always performed first. This algorithm is not guaranteed to generate a solution if the array is configurable. The advantage of this algorithm is the relatively low computational complexity. One interesting point is that this algorithm can be integrated with other approaches. PODEM [36] (path-oriented decision making) was introduced to generate tests for combinational logic circuits. If the Greedy Configuration algorithm is integrated with PODEM like branch and bound and back track scheme, it could guarantee that a solution is found if the array is configurable.

3.8 Test Diagnosis and Configuration Processes

It is feasible to test, diagnoses, and configure the Pseudo-5 connections structure separately. Configuration is performed after testing and diagnosis. The flow chart of the procedure is illustrated in Figure 3.54.

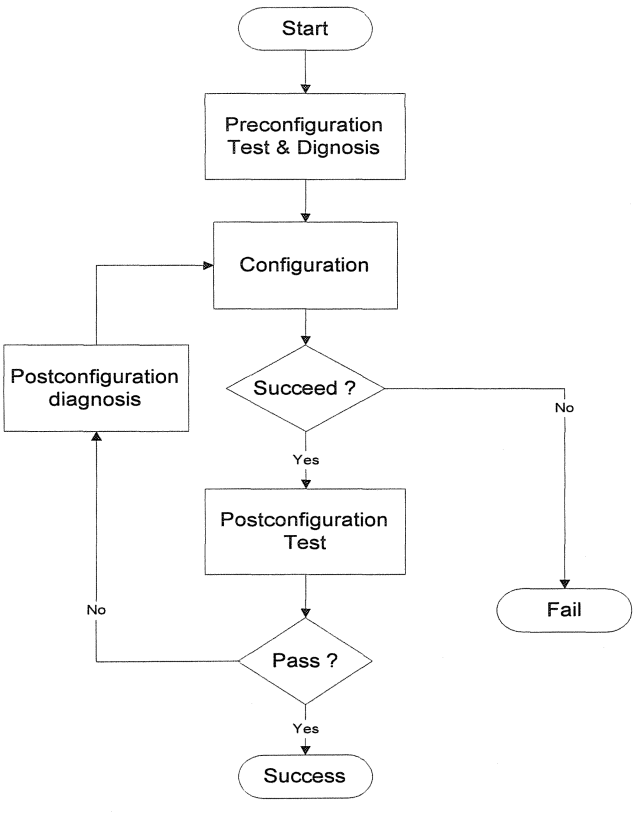


Figure 3.54 Test, diagnosis, and configuration procedure

In the procedure shown in Figure 3.54, pre-configuration test and diagnosis is to determine the status (functioning or faulty) of each element (cell and bundle). If the configuration cannot succeed, the array is not configurable. If it succeeds, post configuration test will follow. Post configuration test is to validate if this post-configured array is fully connected and functioning. Since a perfect pre-configuration test process that is guaranteed to identify all defects during the primary test is not realistic, a post configuration test is necessary. Post configuration diagnosis is to locate the additional faults based on the results of post configuration test. The above procedure should be repeated until a fully connected and functioning array is built or a conclusion is made that this array is not configurable.

It is planned to test this structure with JTAG. Boundary Scan Register is inserted at the ports of transmitters and receivers in each cell. Another data register, Configuration Scan Register, is dedicated to control configuration multiplexers in each cell. By definition, the receiver configuration multiplexers are considered to be part of a cell, whereas long lines, drivers, and transmitter multiplexers are part of a bundle. To locate the defects on this structure is a complex task. This section is only a feasibility exploration of the test and diagnosis of the structure. To simplify the problem, we assume that the defects on one bundle are independent of the defects on other circuits, and that the defects on a cell (PE, input multiplexers and associated configuration wires) are independent of the defects on other circuits. The following discussion assumes stuck at fault or faults that do not change the structure (no sequential behaviours or loops are considered).

3.8.1 Diagnostic Test

The interconnection between a transmitting port and a receiving cell through a transmitter multiplexer and a receiving multiplexer is tested. After setting up the path between the transmitter and the receiving cell, a 4-bit (assuming there are 4 lines per bundle) vector {T0, T1, T2, T3} of Table 3.23 is applied to one transmitter. If the expected pattern {R0, R1, R2, R3} of Table 3.23 is obtained from the associated receiver, the test is completed with successes. Otherwise, it is failed. Take the circuits of Figure 3.44 for example, the test is on the interconnection between the horizontal transmitter Trans_H of cell (0,i) and cell C(1,i) through the configuration multiplexer (Md, Mr) associated with BH(0,i). An example of test results table is shown in Table 3.24.

Table 3.23 Test vectors

T0	T1	T2	T3	R0	R1	R2	R3
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1

Status of a bundle (1, X, 0) 1 : good
X : unknown
0 : defective
Status of a cell (1, X, 0) 1 : good
X : unknown
0 : defective
Status of the test of a pair of terminals (1, --, 0) 1 : pass
-- : don't care
0 : fail

Table 3.24 A test result example

	C(0,0)	C(1,0)	C(2,0)	C(0,1)	C(1,1)	C(2,1)	C(0,2)	C(1,2)	C(2,2)
BH(0,0)(0,0)	1	1	1	1	1	0	--	--	--
BH(0,0)(1,0)	--	--	--	--	--	--	--	--	--
BH(0,0)(0,1)	1	1	1	1	1	0	--	--	--
BH(1,0)(1,0)	--	--	--	--	--	1			
BH(1,0)(0,0)	1	1				0			
BH(1,0)(2,0)	1	1				0			
BH(1,0)(1,1)	1	1				1			
BH(2,0)(2,0)	1	1	1	1	1	0			
BH(2,0)(1,0)	--	--	--	--	--	0			
BH(2,0)(2,1)	0	0	0	0	0	0			
BH(0,1)(0,1)	1	1	1	1	1	0			
BH(0,1)(0,0)	--	0	0	0	0	0			
BH(0,1)(1,1)	1	1				0			
BH(0,1)(0,2)	1	1				0			

3.8.2 Diagnostic Model

A bundle is composed of transmitter multiplexers (Mt), the bundles of wires and drivers on the data bus (the bold line part in Figure 3.55). The configuration receiver circuits are composed of receiver multiplexers (Mr), direction multiplexers (Md) and corresponding connection wires. The test results table shown in Table 3.24 is used to feed system level diagnosis problems, which have been well studied in literature [37][38][39]. In this thesis, there is no further investigation on this issue.

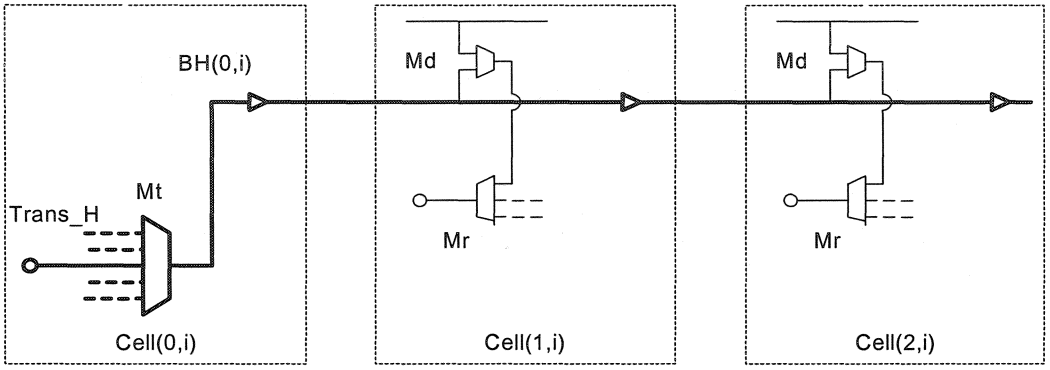


Figure 3.55 Configuration circuits partitioning

3.9 Prototype Demo Chip

To verify the functionality of the fault tolerant structure proposed in this thesis, a prototype demonstrator chip was implemented by a team, in which I was the primary designer, with a 0.18 μM CMOS technology available form Canadian Microelectronics Corporation (CMC) [43]. The chip was fabricated through CMC and was tested via CMC's remote testing services.

3.9.1 Summary of Specification

The demo chip contains an array of 3 x 3 cells with the regular interconnection structure shown in Figure 3.17. Each cell comprises a JTAG controller and they are chained on the chip. In this demo chip, each bus contains 3 bundles and each bundle contains 4 lines.

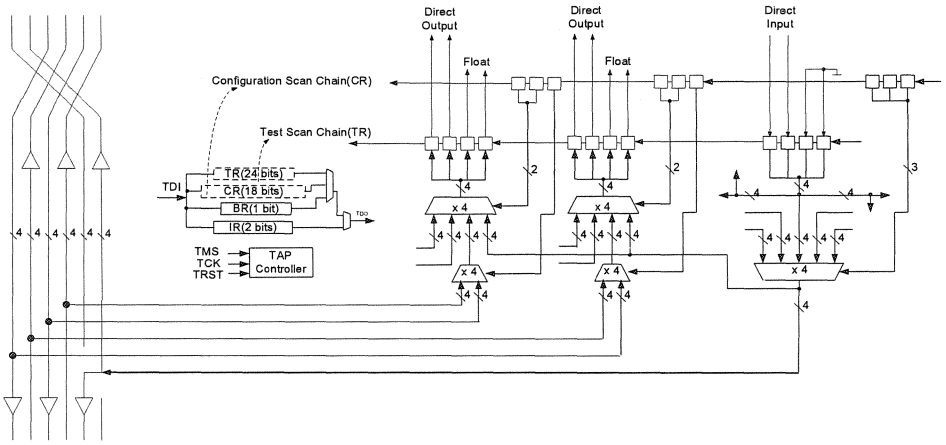


Figure 3.56 Configuration control and test structure

Figure 3.56 shows the vertical bus associated interconnection structure and JTAG elements. The details of the JTAG are presented in Chapter 4. There are similar circuits associated with horizontal bus. A configuration scan chain in each cell controls configuration multiplexers. Its data register comprises 18-bits: (1 (direction) + 2 (sources)) bits/bundle x 2 receiving bundles/dimension x 2 dimension + 3bits x 1 transmitter / dimension x 2 dimensions. An 1149.1 standard Boundary Scan Cell is associated with each emitter and receiver line (i.e. 24 in total in each cell = (2 emitters + 4 receivers) x 4 lines/bundle). The Boundary Scan Cells are located at core side to test the receiving and transmitting multiplexers. The boundary scan register and the configuration register share one JTAG controller in each cell. As shown in Figure 3.56, another test feature is direct inputs and outputs. In the four corner cells (i.e. Cell_00, Cell_20, Cell_02 and Cell_22), the core side of Boundary Scan Cells which are associated to two LSB in a bundle are directly linked to I/Os. For other Boundary Scan Cells, data output ports are left floating and data input ports are grounded. Each corner cell has a total of 12 direct I/Os.

As shown in Figure 3.57, TDI and TDO of each JTAG are chained together. An output pad is added at the TDO of the first cell in the chain (i.e. TDO_22). In case the chained JTAGs are not operating, this output pin allows testing a stand alone JTAG.

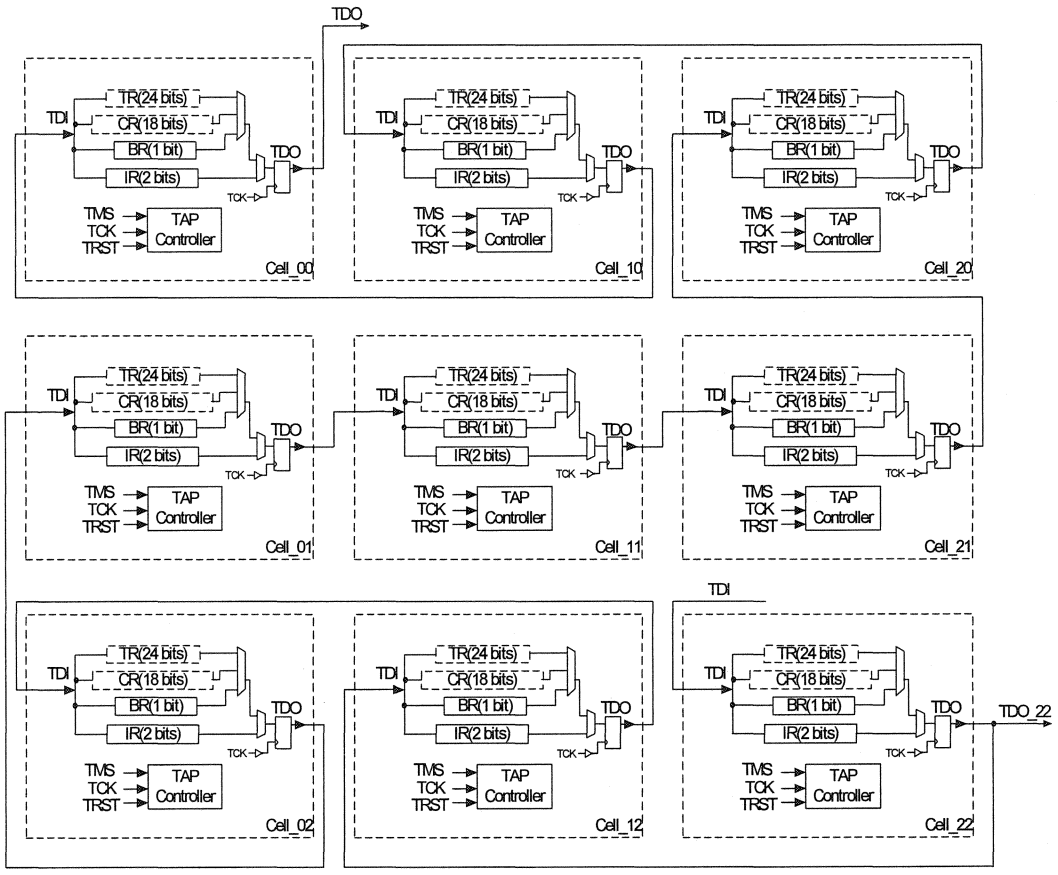


Figure 3.57 chained JTAGs

3.9.2 Design Flow and Design Verification

The design flow of this chip is shown in Figure 3.58. The layout of the regular structure was created manually. The layout of JTAG elements in one cell was generated by place and route tools following the normal digital design flow. The integration of one cell (connecting interconnection structure and JTAG in one cell) was done manually. The 3 x 3 cell structure is produced by reprinting the regular layout of one cell. The global links of 9 JTAGs are connected manually.

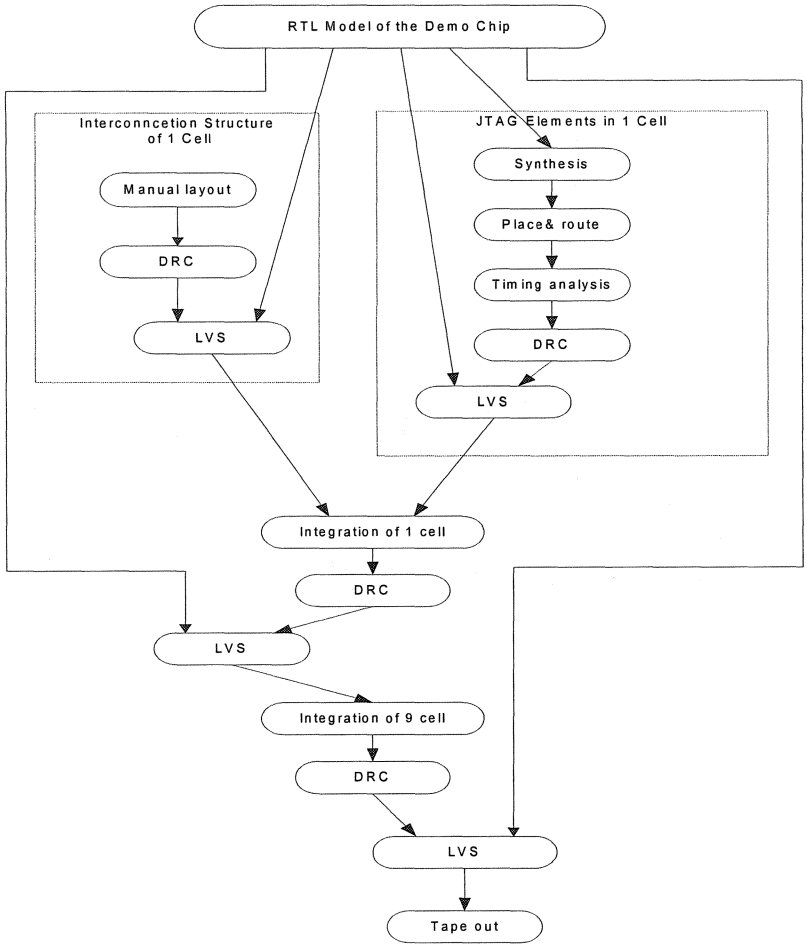


Figure 3.58 Demo chip design flow

The RTL model of the demo chip was coded in VHDL. This model was verified by very detailed simulations. Figure 3.59 shows the 1149.1 integrity test. Starting from the state of Run-Test/Idle, hold TMS at '1' two clock cycles, then hold TMS at '0'. The "10"s at parallel input if instruction registers is captured and shifted out from TDO. It is clear that nine "10" patterns are shifted out.

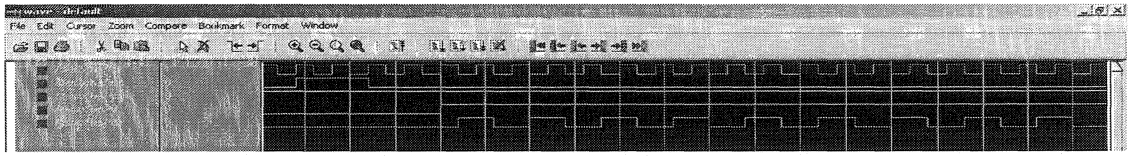


Figure 3.59 Simulation result of logic integrity test

Figure 3.60 shows an example of connectivity test. In this test, all the cells are set to default configuration, all inputs of vertical transmitters are "0000" and all inputs of horizontal transmitters are the associated cells' indexes. The input of horizontal transmitter of Cell_20 is "1000" and the input of the horizontal transmitter of Cell_11 is "0101". The bit sequence of "1110100100101010 11" of cfg_out is the control code for default configuration of Cell_22. For data_out "10100000", "1010" (the index of Cell_22) is the input of the horizontal transmitter of Cell_22, "0000" is the input of the vertical transmitter. For data_in "0010100111110000", "0010" is the data received by R1_H (sent by Cell_02), "1001" is the data received by R2_H (sent by Cell_12), "1111" is the data received by R2_V, and "0000" is the data received by R1_V. All the above results are as expected. Additional to the above general connectivity simulation, 4 configuration examples, in which all configuration wires in one regular cell are used, are chosen to perform functional simulation.

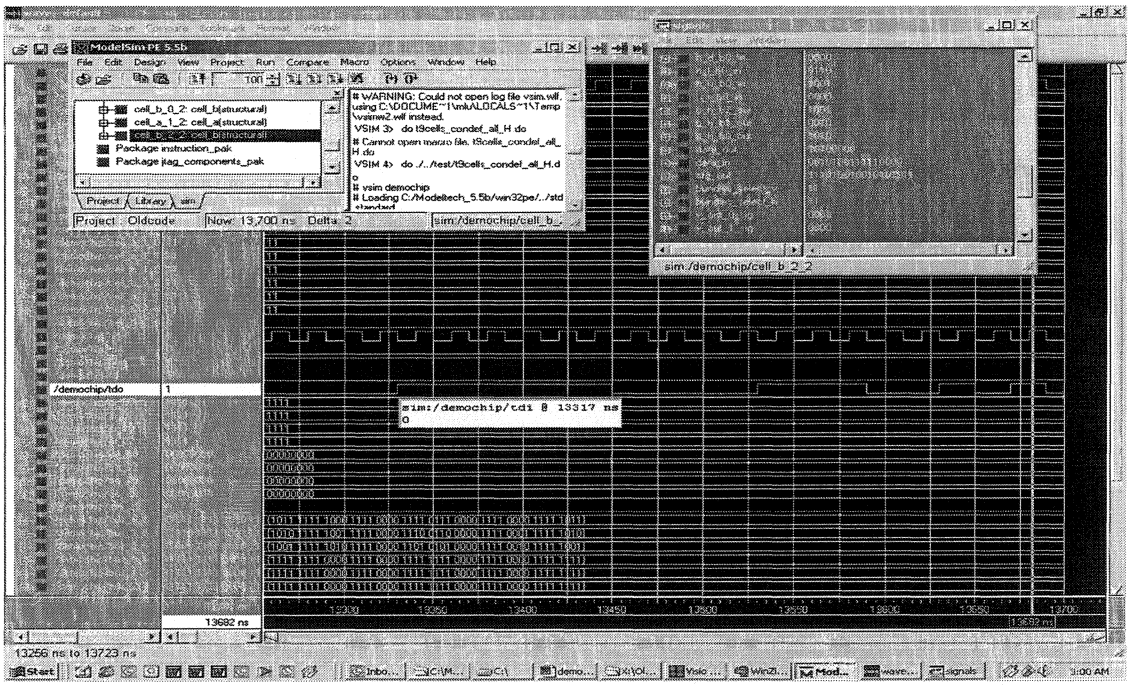


Figure 3.60 Simulation result of connectivity tests

It was verified by timing simulation that one JTAG cell can run at 10 MHz. The global links of JTAGs are not sensitive to clock skew. There are half period of margin for clock skew since data is shifted out on TDO on the falling edge of TCK and data is shift in from TDI on the rising edge of TCK.

The layout of one cell is shown in Figure 3.61. The layout of the demo chip is shown in Figure 3.62.

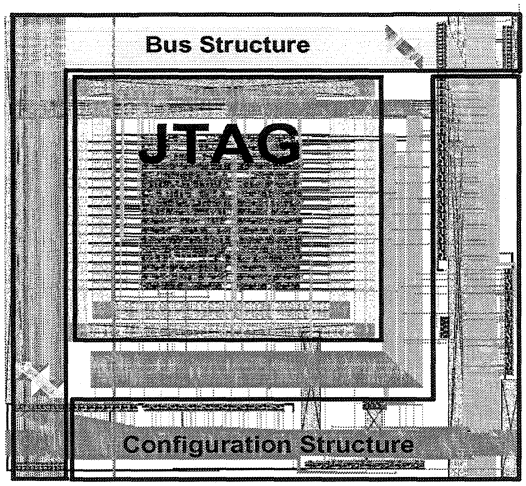


Figure 3.61 Layout of one cell

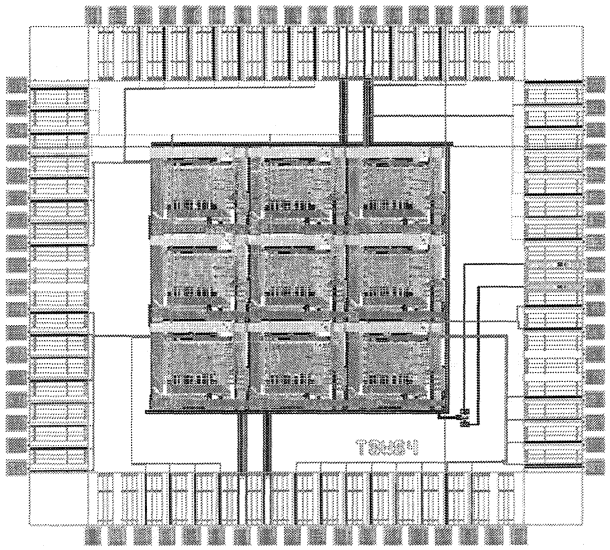


Figure 3.62 Layout of the demo chip

3.9.3 Test

Functional tests are performed to verify if the fault tolerant system can work as expected. Four (shown in Figure A.1, Figure A.32, Figure A.35 and Figure A.43) configurations are chosen to show the flexibility of the structure to support restructuring. All configuration connection wires in one regular cell are captured. Since the layout of each cell is identical, we can test different wires in arbitrary cell for a functional test. Table 3.25 lists the Figures and the physical indexes of cells, where each configuration wire is captured.

Table 3.25 Wire Capture Locations

Connections	Figure	Physical indexes of the cell
R1_H_0	A.35	(1,1)
R1_H_1	A.32	(2,1)
R1_H_2	A.1	All
R1_H_3	A.32	(0,1)
R2_H_0	A.32	(1,0)
R2_H_1	A.32	(2,1)
R2_H_2	A.1	All
R2_H_3	A.32	(0,1)
Tran_H_0	A.32	(0,0)
Tran_H_1	A.1	All
Tran_H_2	A.32	(2,0)
Tran_H_3	A.35	(1,1)
Tran_H_4	A.32	(0,0)
R1_V_0	A.32	(0,0)
R1_V_1	A.32	(1,0)
R1_V_2	A.1	All
R1_V_3	A.43	(1,1)
R2_V_0	A.43	(2,1)
R2_V_1	A.32	(1,0)
R2_V_2	A.1	All
R2_V_3	A.43	(1,1)
Tran_V_0	A.43	(2,1)
Tran_V_1	A.1	All
Tran_V_2	A.32	(0,0)
Tran_V_3	A.32	(0,0)
Tran_V_4	A.43	(2,0)

Figure 3.63 shows the sequence of a test. At first, a configuration of interest is set up by Configuration Scan Chain. Then the inputs of transmitting multiplexers are set by BSR (Boundary Scan Register). After this, capture BSR and shift contains out. Finally, we compare the data received by receiving multiplexers with the expected data to see if the circuits work in the expected way.

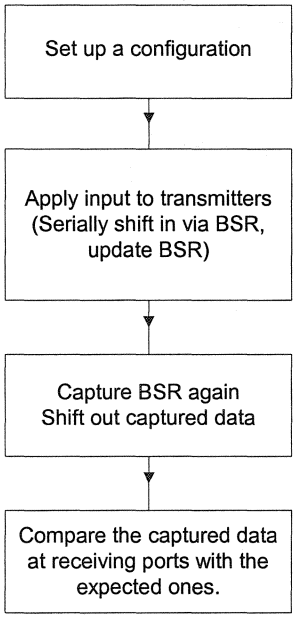


Figure 3.63 Sequence of a test

The test vectors are created by HDL simulation. All vectors for the 4 configurations passed testing. The circuits worked properly at 50MHz.

3.10 Summary

Based on earlier work, Pseudo-5-source configuration structure that supports 12 basic replacement operations with less connection wire overhead has been proposed. The intra-bus bundle replacement supports the bundle shift along the second direction and enables the array to support turned-path cell replacement. This allows repairing more clustered

defect patterns. Configuration algorithms and the procedures for testing, diagnosing, and configuring were discussed. A set of configuration rules based on the physical interconnection constraints was formalized. The limitations of this interconnection structure on the configuration algorithm were investigated. The PeP configuration should be avoided if there is an alternative PaP. The Greedy Global Mapping Configuration Procedure was developed. The strategy of this algorithm is to try leaving as many elements as possible for the following assignments while making all the computation local to minimize computational complexity. To verify the functionality of the fault tolerant structure, a prototype demonstrator chip was developed and successfully tested.

CHAPTER 4

DEFECT AND FAULT TOLERANT SCAN CHAIN

4.1 Introduction

The scan chain that allows access to a large number of state bits from a fixed number of pins is very popular in current sequential circuits and systems testing. Additionally, the scan chain concept has been used to control the state of a defect and fault-tolerant architecture [45][46].

It has been proved that a long chain without fault tolerance gives poor system yield, while a tolerant chain becomes insignificant to system yield [47]. Several fault tolerant scan chains have been proposed in the literature. The two tolerant chains with the combinational testing structure and the counter based tolerant chain have no critical portion, where a single defect would make the chain inoperative [47]. In their design, the elements of the scan chain need to be doubled, and the inputs of the a stage of scan chain elements are selected from two of the proceeding stage outputs according to its combinational or counter based testing results. In another multiple-defect tolerant scan chain design [48], a section of the scan chain is bypassed if it is identified as defective by self-testing, while the global scan path becomes shorter but defect-free. Defects in the primary I/O-buses are repaired by reconfiguration using spare lines, and arrangements of laser fuses and anti-fuses. A bi-directional scan chain, which is able to generate two different polynomials from the same linear feedback shift registers [49], can tolerate one defect in its structure.

IEEE 1149.1(JTAG) provides a standard mechanism for system testing, and is a convenient interface to some testing approaches, such as Built-In-Self-Test (BIST). Today, the majority of custom ICs and programmable devices contain 1149.1. New applications for the 1149.1 protocol have been introduced, most notably "In-System Configuration" (ISC) capability for Field Programmable Gate Arrays (FPGAs). The recently balloted standard, IEEE1149.4 Mixed-Signal Test Bus builds upon the base created by 1149.1 [50]. IEEE1149.1 was proposed by Landis [51] as a common interface for WSI system level processing tests, restructuring tests and system tests. In his proposal, the test is performed through probe cards contacting with 7 pads (VDD, VSS, TDI, TDO, TMS, TCK and TRST) of each JTAG associated with a cell on wafer. Since these JTAGs associated with each cell are not linked on a wafer, there exist certain constraints to use them. The simple links of JTAGs as they are on a PCB may cause a significant critical area. To enable the IEEE1149.1 supporting WSI testing and configuring in a more flexible way, building fault tolerance on IEEE 1149.1 is of interest.

In this section, we propose a scheme for on-wafer fault tolerance IEEE1149.1 global links. With this scheme, we can take advantage of features supported by the IEEE1149.1, while the yield loss caused by the global link can be reduced drastically by its fault-tolerant design. Additionally, an alternative approach that links the JTAGs off-wafer is proposed.

4.2 Overview of IEEE 1149.1

In conventional system design hierarchy, the basic architecture of 1149.1 Boundary-Scan is incorporated at the Integrated Circuit level. The serial links between the basic modules are created at PCB level.

4.2.1 Basic Architecture

The following discussion of the JTAG standard refers to the block diagram in Figure 4.1. First, four (optionally, five) new package pins are dedicated to Boundary-Scan. These pins form the Test Access Port (TAP) and must be dedicated to Boundary-Scan. They are used with a simple protocol to communicate with on-chip Boundary-Scan logic. The protocol is driven by two of the pins (three if the optional Test Reset nTRST input pin is included). These two input pins are a Test Clock (TCK) and a Test Mode Select (TMS). The remaining two pins are for serially shifting data into and out of the IC, labeled Test Data In (TDI) and Test Data Out (TDO)[50].

The TAP controller is a finite state machine with a state diagram containing 16 states. A transition between states only occurs on a rising edge of the TCK or asynchronously with the assertion of Test Reset (nTRST), if it exists.

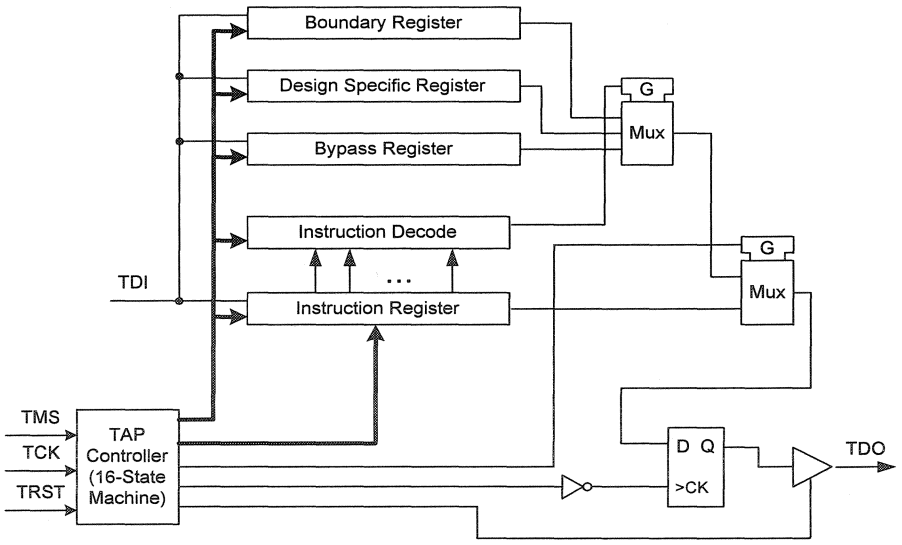


Figure 4.1 JTAG elements of a Boundary-Scan IC

The Instruction Register defines the mode in which Boundary-Scan data registers will operate. All Boundary-Scan instructions set operational modes that place a selected data register between TDI and TDO. This register is referred to as the target register. One mandatory data register is the Bypass Register. When selected by the BYPASS instruction, the Bypass Register shortens the shift path within an IC to a single cell. The Boundary Scan Register is used to control and observe activities on the IC's input and output pins. The standard also allows designers to implement user-defined registers. The Configuration Scan Register mentioned in this thesis belongs to this category.

4.2.2 Boundary-scan Chains

Boundary-Scan ICs are designed to form link together into chains. A simple chain on a printed circuit board is shown in Figure 4.2. Simple chains are a collection of Boundary-Scan ICs with common TCK and TMS, and with their shift paths linked together by connecting a TDO pin of one IC to the TDI pin of the following IC.

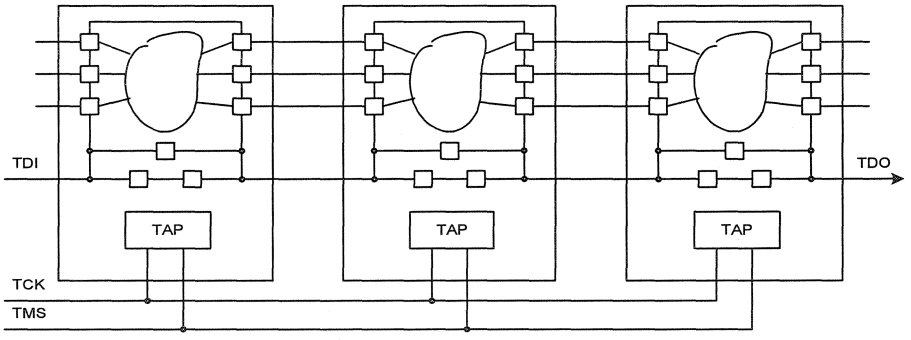


Figure 4.2 Simple chain of Boundary-Scan ICs

4.3 Tests and Configuration Structure in One Cell

As shown in Figure 4.3, in one cell of the Pseudo-5 connections array, the multiplexers are used to control the configuration (i.e. to control how the PE is connected to the bus

structure). There are three data registers. The Test Register (TR), which refers to the Test Scan Chain, is used to test the data buses and configuration structures. The Configuration Register (CR), which refers to the Configuration Scan Chain, is used to control the configuration multiplexers. The Bypass Register is used to shorten the path when it is necessary. The active data register is chosen by the loaded instructions.

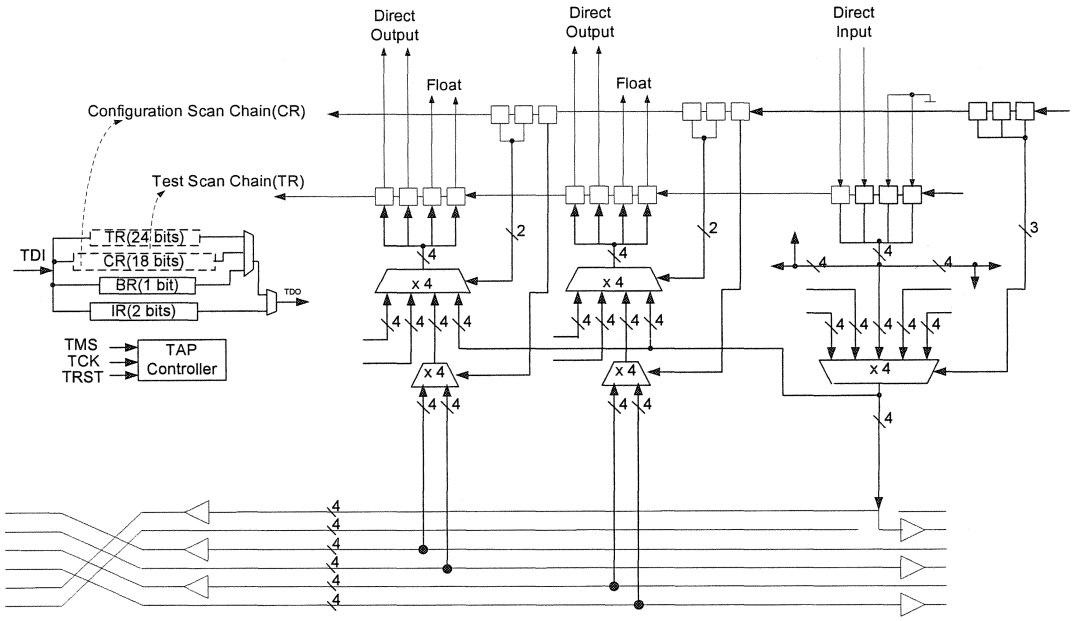


Figure 4.3 Configuration control and test structure

4.4 Fault Tolerant Scan Chain (FTSC)

To perform testing and configuration, it is necessary to connect all the JTAGs located at each cell serially by connecting the TDO of one cell to the TDI of the succeeding cell. For a PCB, chips with defective 1149.1's can be replaced by good ones. If the same simple links are implemented on a WSI device, one defective 1149.1 may cause the loss of the whole wafer. It was proven that the JTAG failure associated with one cell is not critical in this fault tolerant structure, because the cell can be replaced diagonally or to be specified as redundant cell. However, it is necessary to insure that the global scan path

should still function if some defects affect some 1149.1 modules and interconnect links. We propose two approaches to build a fault-tolerant scan chain. One is an on-wafer (internal) fault-tolerant scan chain; the other is an off-wafer (external) fault-tolerant scan chain.

4.4.1 On-wafer Fault Tolerant Scan Chain

For the proposed global scan chain being capable of tolerating multiple defects, some defects on the critical portion can be tolerated, such as in the global data path, the control logic, the clock, and the power supply. Some defective 1149.1 circuits and interconnections may cause the inability to use certain cells and bundles; however, the global scan path should be still functioning.

4.4.1.1 General Architecture

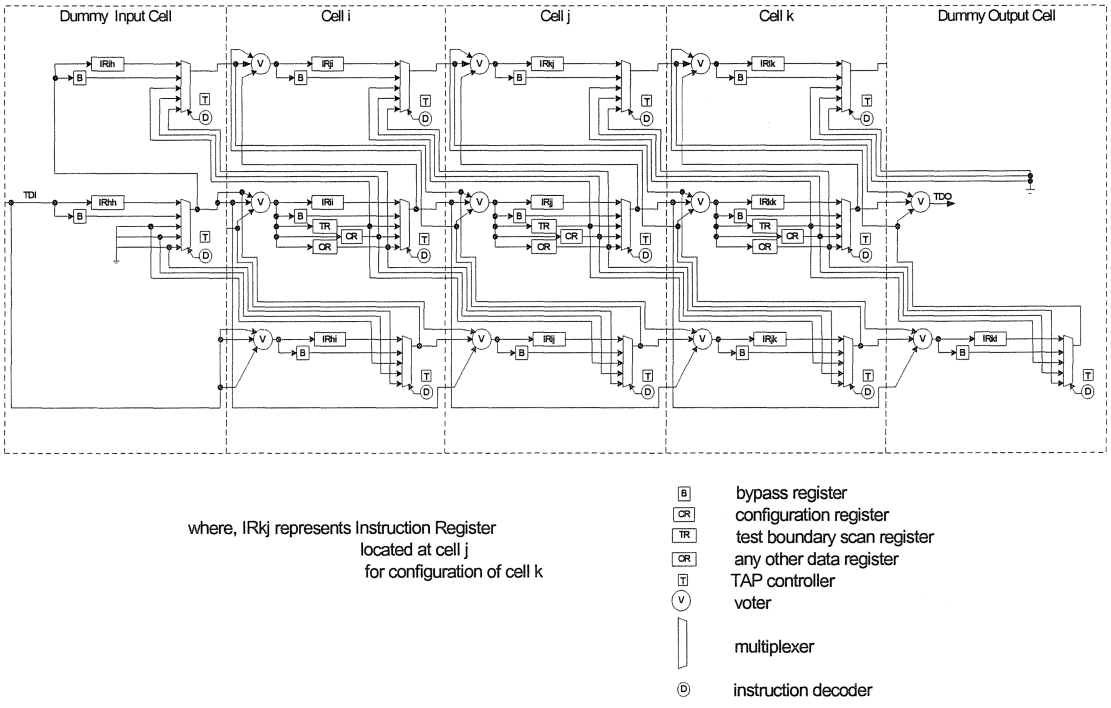


Figure 4.4 On wafer fault-tolerant scan chain

Figure 4.4 shows a conceptual diagram of an on-wafer fault tolerant scan chain. Cell i, Cell j and Cell k contain regular JTAG elements, and represent regular on-wafer cells. The regular cell layout that contains JTAG elements can be extended to produce any desired number of on-wafer cells. It is suggested a complete FTSC logic contains also a Dummy Input Cell and a Dummy Output Cell that are irregular, and those irregular parts can be implemented off-wafer. Critical parts in the data path (instruction registers, bypass registers and multiplexers) to propagate instruction and data are tripled compared with the JTAG without fault tolerance. Other data registers, i.e. configuration registers and boundary scan registers, are not tripled. Three groups of JTAG elements in one cell are stretched in such a way that they are associated with the IEEE 1149.1 functionalities of three cells. The mid group of JTAG elements is associated to the local cell. The top group is associated to the right neighbour. The bottom group is associated to the left neighbour. The three groups of JTAG elements located in any given cell have independent TAP controllers. Power supply and clock, are also critical parts. Thus they should be independent of each cell. A stage is the 3 groups of JTAG elements associated to one cell. The defects on one of the three groups of JTAG elements in one stage can be masked by the voters of the succeeding stage, since the two fault-free groups still can supply enough correct signals to the voters of the next stage. Defective modules associated to the local cell (the mid part) can be bypassed from the global scan chain if they are defective. This may cause the dropping of the associated cell and the two associated bundles from the array since they could not be configured properly.

The FTSC is able to tolerate one faulty group at each stage. Figure 4.5 shows a case where multiple faults are tolerated. There are three groups of faulty elements, group ii, group ji, and group kj, that are associated to the IEEE 1149.1 functionality of cell i, cell j and cell k correspondingly. The bold lines show a case of global data path of interest, in which the testing register of cell j and the bypass registers of the rest are linked. The dotted lines are the paths that may pass incorrect signal caused by the 3 faulty groups. It can be seen that the voters of the succeeding stages can always mask the faulty signals

caused by the previous faulty groups. Although there are certain faulty groups, the functionality of global scan path is restored. The failure of clock or power in one cell, that is equivalent to the failure of 3 groups of JTAG elements associated with 3 neighbour cells, is not critical. If the rest of the JTAG elements associated with the 3 cells are functioning, the global scan chain survives.

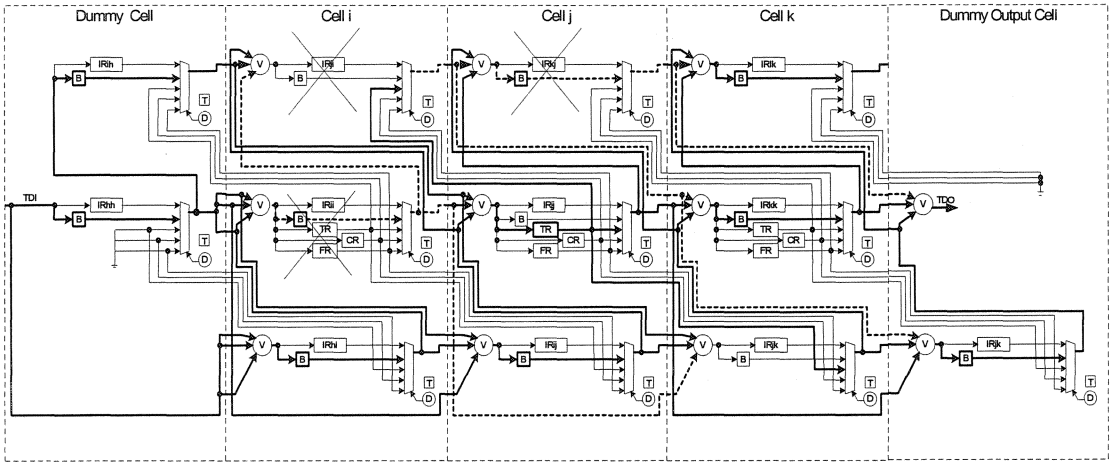


Figure 4.5 Example to show how multiple faults are tolerated

Instead of tripling all the controls for three groups of JTAG elements, three groups of JTAG elements in one cell may share the same state machine; however, the instruction decoder has to be tripled. The state-machine sharing reduces the ability of fault-tolerance. When there is only one state machine, a fault on that state machine is equivalent to the power or clock failure on the associated cell.

4.4.1.2 Timing Constrains

Since the JTAG elements that are linked together by voters and long interconnections need to be synchronized, timing is an issue. Fortunately, the synchronization over long interconnections is well considered in IEEE 1149.1. Data is shifted out on TDO on the falling edge of TCK. While, data is shifted in from TDI on the rising edge. Data is set up

on TDO a half TCK cycle before TDI of next stage is read. A negative clock skew can be tolerated up to a half clock period. Based on this scheme, there are two approaches to cope with timing problems. One is to increase the period of TCK to tolerate larger delay and skew. Another is to insert drivers on the long interconnections to optimize delay. The driver insertion should be implemented with care since improperly inserted buffers may cause significant sensitivity to power failure.

4 .4.1.3 Scan Chain Integrity Test

The IEEE 1149.1 standard supports a set of tests of the integrity of IEEE 1149.1 circuits [50] [52]. The visibility of the FTSC is decreased significantly by the involvement of the voting logic. Fortunately, the invisible defects are also the tolerable defects that are not necessary to locate. The 1149.1 supported integrity tests are still applicable for this fault tolerant scan chain integrity test.

4.4.1.4 Layout Regularity Constrains

Layout regularity is important for both economical and technical reasons. As shown in Figure 4.4, the regularity of each cell of the on-chip portion is kept. In order to keep the integrity of the fault tolerant scan chain, a dummy cell at input side and a dummy cell at output side are placed. The layout regularity demands that the layout of the JTAG elements in each cell should be the same, while the global scan chain is an irregular structure. An inter-reticle stitching method to produce irregular global scan path with identical reticle is proposed in this thesis.

Inter-reticle stitching is the manufacturing method with which the mask of one reticle is connected to reprinted copies to cover a wafer. The global scan chain can be generated with the mask of one reticle by playing with the spacing between two neighboring prints. As shown in Figure 4.6, the regular normal spacing connects the scan path along the row only, while the narrow and wide spacing at the boundaries makes the connections

between rows as well. Since the connectivity of data buses is not sensitive to horizontal spacing, the regular bus interconnect structure is maintained. A two-line version of the stitching method is shown in Figure 4.7. This is a generic scheme that can be applied to the cases of multi-line scan chains. This scheme should be implemented in combination with a regular interconnection cut of the scan chain between neighbour cells as shown in Figure 4.4.

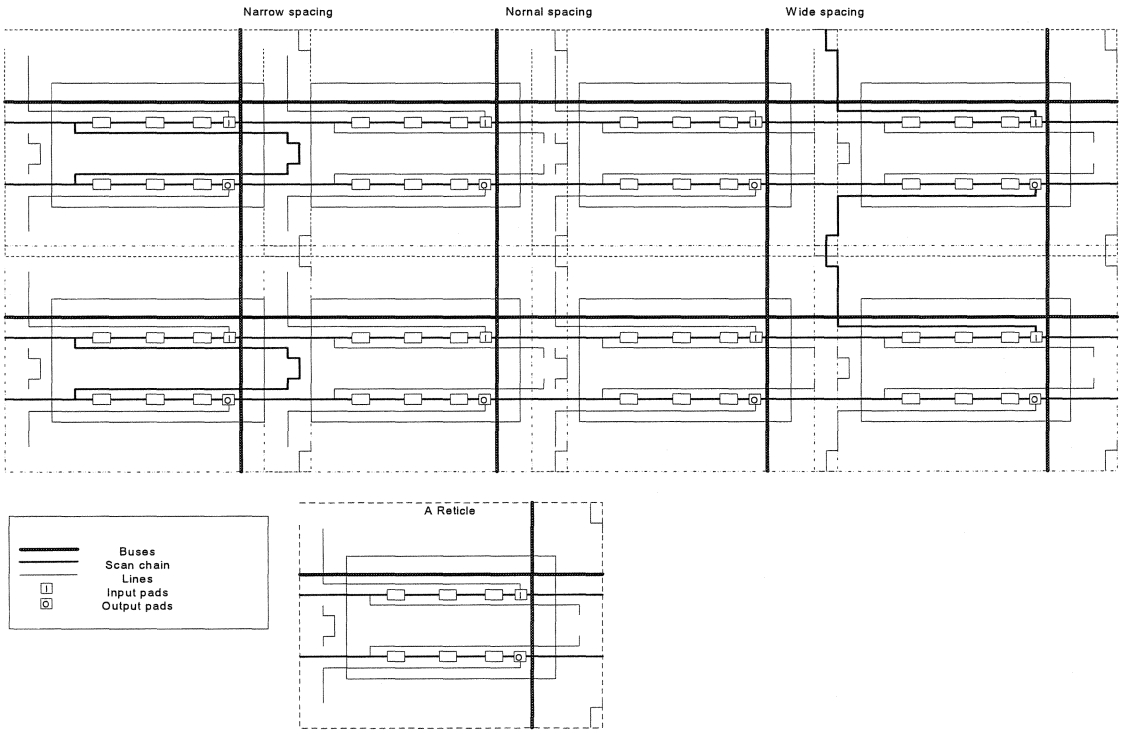


Figure 4.6 Stitching method to generate global scan chain

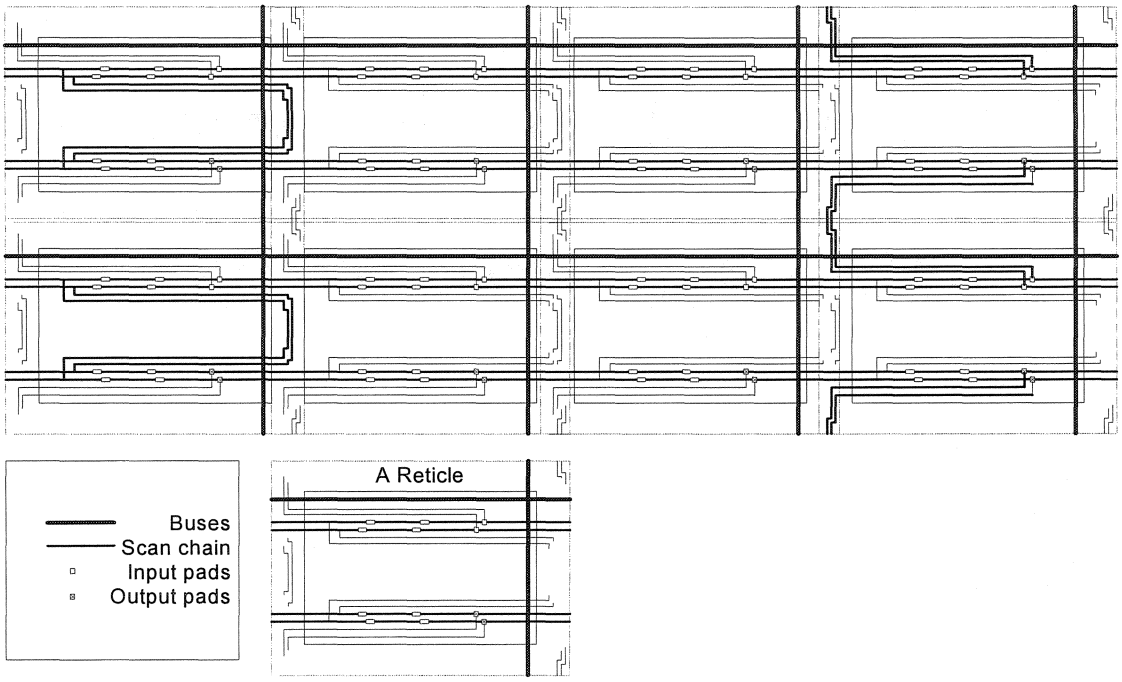


Figure 4.7 Two-line version of the inter-reticle stitching method

4.4.2 Off-wafer Fault Tolerant Scan Chain

The conceptual design of an alternative, in which the global scan chain is linked on an FPGA, is shown in Figure 4.8. As it is known that an FPGA is much less expensive than a whole wafer, it should not be costly to guarantee the global scan chain on an FPGA to be defect free. If one on-wafer JTAG is defective, then the failed one is bypassed using a corresponding module on the FPGA. It is very similar as with an on-wafer fault tolerant scan chain. The fault on the JTAG is equivalent to the faults on a cell and its two associated natural bundles.

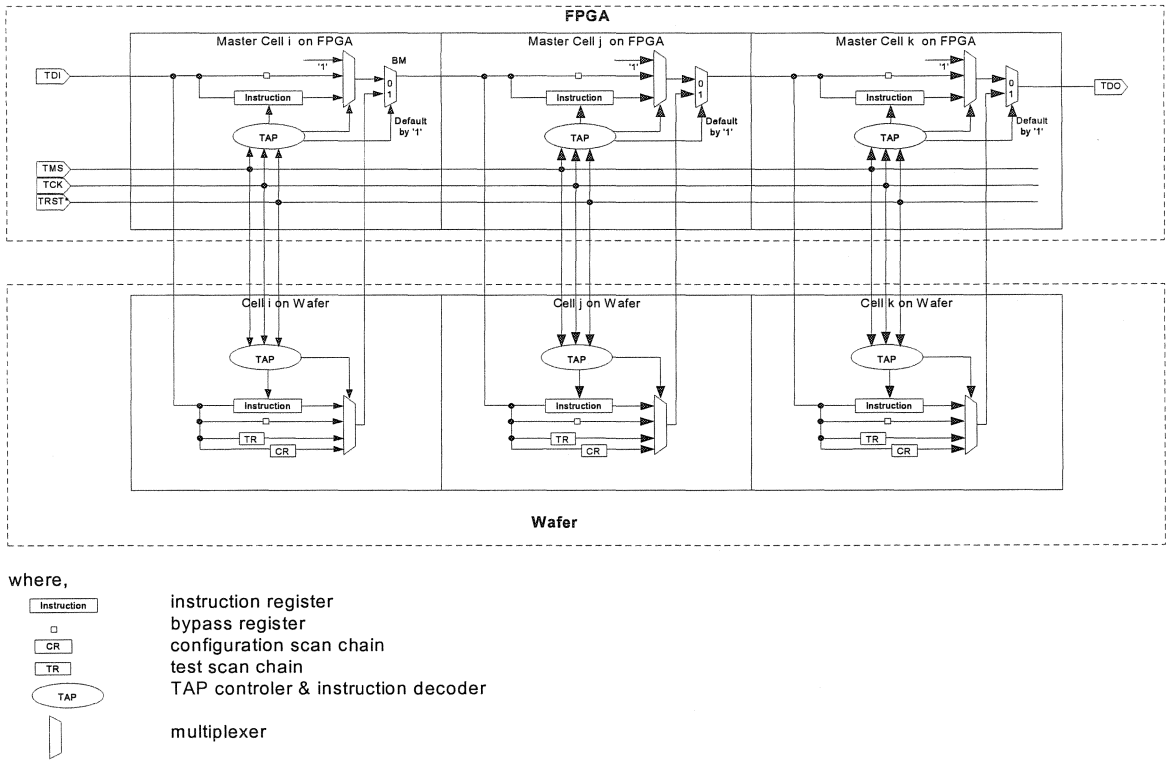


Figure 4.8 Global scan chain connected on FPGA

The off-wafer FTSC can be implemented with a master/slave architecture. There is a configuration scan chain (CR) and a test scan chain (TR) in a slave JTAG while these elements are not present in a master JTAG. The bypassing multiplexer only exists in the master JTAG to select the source of TDO from the master JTAG or from the slave JTAG. Except for the above differences, the rest of the parts of the on-wafer JTAG (slave JTAG) and the on-FPGA JTAG (master JTAG) are identical,

Since the TAP and Instruction Registers in the two JTAGs (master and slave) are identical, the two JTAGs can always keep the same mode and state. If there is an instruction that shifts the data along the test-scan register, the slave JTAG does the real operation of shifting, while the master JTAG just does dummy shifting (shift '1' out). If

the instruction is bypassing slave JTAG, the master JTAG controls the bypass multiplexer to select the output of master JTAG as TDO, while the slave JTAG just does a dummy null operation. From the beginning, all slave (on-wafer) JTAGs are put on the global chain. As a defective slave JTAG is found, it is bypassed from the FPGA. Even if the bypassed JTAG is still active as before the bypassing, it does not disturb the operation of the global scan chain any longer.

Before performing test and configuration on the interconnection structures, the scan chain itself should be tested and diagnosed first. The IEEE 1149.1 supports a set of IEEE 1149.1 integrity tests. The test and diagnosis of the Configuration Register and the Test Register can be grouped with the test and diagnosis of the whole interconnection structure.

Since there is no tripled structure as with the on-wafer JTAG, off-wafer FTSC is much less complex. The disadvantage of the off-wafer fault tolerant JTAG is that there are 4 dedicated pads (optional 5) in each cell. For a large number of cells, a larger number of IOs are needed.

4.5 Proposed Demonstrator Circuit to Verify the On-chip Fault Tolerant Scan Chain

The IEEE 1149.1 supported scan chain integrity testing is still available for this fault tolerant scan chain. However, to verify the concept of this design, it is preferred to gain more controllability and observability on its global data path and boundary scan cells. For this objective, as shown in Figure 4.9, a dedicated testing JTAG is used in a proposed proof of concept demonstrator chip. The testing JTAG contains 5 external boundary-scan registers that are associated with 5 logical stages of JTAG elements. The external boundary-scan registers serve as the means to monitor and control the activity of the

points between two stages of JTAG elements. In order to control and test different modes and states of operation of the fault tolerant scan chain, the test scan chain should have independent TMS. A series of faults on clock and power domain can be emulated by the IPCC (Independent Power and Clock Control).

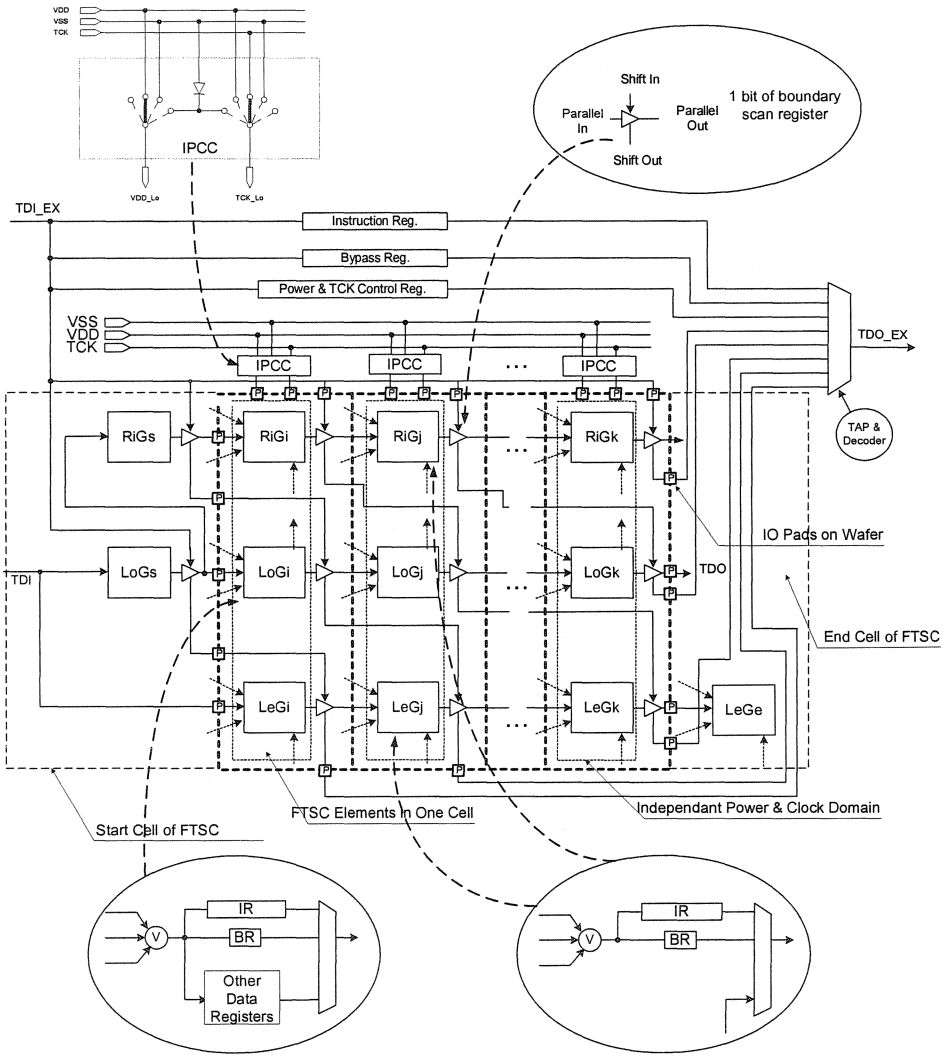


Figure 4.9 Demonstrator for on-chip fault tolerant JTAG testing

The above conceptual design is coded with VHDL. The VHDL model is validated with simulations. By the time of writing the thesis, there no on-chip implementation is performed.

4.6 Summary

The global IEEE 1149.1-based fault tolerant scan chain can be built on-wafer. The critical portions of data paths and controls are tripled. The proposed design is able to tolerant multiple defects. Multiple faulty JTAG element groups in exclusive stages can be tolerated simultaneously. The layout regularity is supported by the newly proposed inter-reticle stitching method. An alternative fault tolerant scan chain design was proposed to link the global scan chain on an FPGA. Defective on-wafer JTAGs can be bypassed from the FPGA. Both approaches demand independent power supply and clock domains for the JTAG modules located in each cell. The two versions of fault tolerant scan chain bring us the flexibility to benefit an IO bound design or an area bound design. The on-wafer fault tolerant scan chain is preferred for IO constrained designs and the off-wafer fault tolerant scan chain is preferred for area-constrained designs.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

This thesis aimed at research in several aspects of WSI system design that include characterization of thermal gradient induced skew, fault tolerance and testing.

The thermal gradient induced skew has been studied and characterized. The wiring resistance and lateral coupling capacitance are the main contributors to RC delay on long interconnections for deep submicron CMOS processes. There is approximately linear skew induced by temperature gradient. 70% of skew induced by temperature gradient is contributed by resistance variations of the interconnections, which is the dominant phenomenon. It is considered that temperature gradient may have significant impact on signal integrity of wafer scale interconnections. The results of the research show that a 1°C temperature gradient can generate 7.6 ps skew on a 20 cm bus. Temperature gradient induced skew is not significant that is negligible for local interconnections as well as for global buses; however, it may be a significant factor for clock distribution networks.

Based on earlier works, Pseudo-5-sources configuration structure with less wiring overhead has been proposed. The structure supports 12 basic replacement operations. The intra-bus bundle replacement enables the bundle shift along the second dimension and enables the array to support turned-path cell replacement, thus makes the repair of more clustered defect patterns feasible. It has been demonstrated in this thesis that the structure is able to repair many clustered defect patterns. In addition, some irreparable defect patterns have been identified and analyzed.

A set of configuration rules based on the physical interconnection constraints have been identified. A novel Greedy Global Mapping Configuration Algorithm has been proposed. The strategy of this algorithm is to keep as many elements as possible for the following mapping assignments and making all the computation local to minimize computational complexity. To verify the functionality of the fault tolerant structure, a prototype demonstrator chip was developed and tested.

The procedures for testing, diagnosing, and configuring have been investigated and analyzed. The test and diagnosis on the Pseudo-5 connections structure have been proved to be feasible. The further impacts of this interconnection structure on the configuration algorithm have been discussed.

The IEEE 1149.1 standard is considered to perform wafer scale testing and configuring. To reduce the yield loss contributed by the global IEEE 1149.1 links, an on-wafer global IEEE 1149.1 based fault tolerant scan chain has been proposed. In the proposed designs, the critical portion of the data path, the power supply, the clock and the controls for the operation of the IEEE 1149.1 structure are tripled. The proposed design is able to tolerate multiple defects. Multiple faulty JTAG element groups can be tolerated simultaneously as long as they are in exclusive stages. The layout regularity is supported by a novel inter-reticle stitching method and dummy off-wafer JTAGs. An alternative design of fault tolerant scan chain is to link the global scan chain on an FPGA from which the defective on-wafer JTAG can be bypassed. Both on-wafer and off-wafer approaches demand an independent power supply and an independent clock domain for the JTAG elements located at each cell.

5.2 Future works

For the wafer scale integration prototype design, many continued issues should be explored.

Thermal gradient is not a significant factor to induce skew on very long buses. However, the global bus design is still a combinational optimization problem, which faces tradeoffs among performance, power dissipation, area, and reliability.

A very important task for the prototype verification is the yield modeling of this fault tolerant structure. This is currently studied by Qiu. More redundancy and more bypass wires can tolerate more defects. The defect density varies from foundry to foundry, from generation to generation of technology and from different maturity stages of fabrication. The cost-effective implementation of a WSI device demands appropriate yield prediction tools.

As mentioned in Chapter 3, the combination with PODEM like decision tree scheme can guarantee that the Greedy Configuration Algorithm reaches a configuration solution if there is one. However, a detailed implementation would need further study. Such procedures should then be implemented and validated.

References

[1] SACK, K.A., et al, "Evolution of the concept of a computer on a slice", Proc. IEEE, vol.52, pp.1713-1770 (1964).

[2] AUBUSSON, R. C. and CATT, I., "Wafer-scale integration - a fault-tolerant procedure", IEEE J. Solid-State Circuits, vol. SC-13, pp. 339-344 (1978).

[3] TEWKSBURY, S. K., "Wafer-Level Integrated Systems: Implementation Issues", Kulwer Academic Publishers, 1989, ISBN 0-7923-9006-7.

[4] www.hyperchip.com.

[5] CHANDRAKASAN, A., BOWHILL, W.J., and FOX, F., "Design of High-performance Microprocessor Circuits", IEEE PRESS, 2001, ISBN 0-7803-6001-X.

[6] PARKER, K. P., "The BOUNDARY-SCAN HANDBOOK", ISBN 0-7923-8277-3.

[7] GWENNAP, L., "Power issues may limit future CPUs", Microprocessor Report, 10(10), Aug. 1996.

[8] CHENG, Y-K., et al, "ILLIADS-T: an electrothermal timing simulator for temperature-sensitive reliability diagnosis of CMOS VLSI chips", IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol.17, no.8, pp668-681, 1998.

[9] BANERJEE, K. et al, "On thermal effects in deep sub-micron VLSI interconnections", Proc. Design Automation Conference, 1999, pp. 885-891.

[10] CHENG, Y-K. et al, "iCET: A Complete Chip-Level Thermal Reliability Diagnosis Tool for CMOS VLSI chips", Proc. Design Automation Conference, 1996, pp.548-551.

[11] JARVIS, D. B., "The effects of interconnections on high-speed logic circuits", IEEE Trans. Electron. Computers, vol. EC-10, pp. 476-487, Oct. 1963.

[12] CHANDRADRAKASAN, A. et al, "Design of High-performance microprocessor Circuits", IEEE PRESS, ISBN 0-7803-6001-X, 2001.

[13] ELMORE, W. C., "The Transient Analysis of Damped Linear Networks With Particular Regard to Wideband Amplifiers", J. Applied Physics, vol.19, no. 1, pp.55-63, 1948.

[14] ISMAIL, Y. I. and FRIEDMAN, E. G., "Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuits", IEEE Trans. on Very Large Scale Integration (VLSI) systems, vol.8, no.2, Apr. 2000.

[15] HO, R., et al, "The Future of Wires", Proc. of The IEEE, vol.89, no.4, April 2001.

[16] ANAND, M. B, SHIBATA, H., and KAKUMU, M., "Optimization Study of VLSI Interconnect Parameters", IEEE Trans. on Electron Devices, vol.47, no.1, Jan. 2000.

[17] BAKOGLU, H. B., "Circuits, Interconnections, and Packaging for VLSI", Addison-Wesley, Reading, MA, 1990.

[18] CHATTERJEE, A., MACHALA, C. F. and YANG, P., "A Submicronic DC MODFET Model for Simulation of Analog Circuits", IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems, vol.14, no.10, Oct. 1995.

[19] POWER, J. A. et al, "An Investigation of MOSFET Statistical and Temperature Effects", Proc. IEEE 1992 Int. Conference on Microelectronic Test Structures, vol.5, Mar. 1992.

[20] OSMAN, A. et al, "An Extended Tanh Law MOSFET Model for High Temperature Circuit Simulation", IEEE JSSC, vol.30, no.2, Feb. 1995.

[21] AJAMI, A. H., BANERJEE, K., PEDRAM, M., and GINNEKEN, L., "Analysis of Non_Uniform Temperature-Dependent Interconnect Performance in High Performance ICs", DAC'2001, Jun. 18-22, 2001.

[22] CHENG, Z. Y. et al, "Full Chip Thermal Simulation", Proc. Of the first IEEE International Symposium on Quality Electronic Design, 2000, pp.145-149.

[23] WU, Q. et al, "Dynamic Power Management of Complex Systems using Generalized Stochastic Petri Nets", Proc. Design Automation Conference, 2000, pp.352-356.

[24] Private conversation.

[25] JIN, Z. , "Method for the Modeling and Analysis of MIS interconnects in VLSI Circuits " Ph. D's Thesis, Ecole Polytechnique de Montreal, Dec. 2002.

[26] TSUDA, N., "Fault-Tolerant Processoe Arrays Using Additional Bypass Linking Allocated by Graph-Node Coloring," Computers, vol.49, no.5, pp431-442, May 2000.

[27] CHEN, Y.-Y., UPADHYAYA, S. J., and CHENG, C.-H., "A Comprehensive Reconfiguration Scheme for Fault-Tolerant VLSI/WSI Array Processors," Computers, vol.46, no.12, pp1363-1371, Dec. 1997.

[28] TEWKSBURY, S. K., "Physical Boundaries of Performance: The Interconnection Perspective," Proc. 1991 Int'l Workshop Defect and Fault Tolerance in VLSI Systems, pp.227-246,1991.

[29] ROSENBERG, A. L., "The Diogenes Approach to Testable Fault-Tolerant Arrays of Processors," IEEE Trans. Computers, vol.32, no.10, pp.902-910, Oct. 1983.

[30] NEGRINI, R., SAMI, M., and STEFANELLI, R., "Fault Tolerance Techniques for Array Structures Used in Supercomputing," Computer, pp.78-87, Feb. 1986.

[31] HOSSEINI, S. H., "On Fault-Tolerant Structure, Distributed Fault-Diagnosis, Reconfiguration, and Recovery of the Array Processors," IEEE Trans. Computers, vol.38, no.7, pp.932-942, Jul. 1989.

[32] HAYES, J. P., "A Graph Model for Fault-Tolerant Computing Systems," IEEE Trans. Computers, vol.25, no.9, pp.975-884, Sep. 1976.

[33] NEGRINI, R. et al, "Fault Tolerance Though Reconfiguration in VLSI and WSI Arrays", The MIT Press, 1989, ISBN 0-2621-4044-6.

[34] VARVARIGOU, T. et al, "Some New Algorithms for Reconfiguring VLSI/WSI Arrays", 1990 Intern. Conf. On Wafer Scale Integration, pp. 229-235.

[35] WANG, K. and LIN, J., "Integrated Diagnosis and Reconfiguration Process for Defect Tolerant WSI Processor Arrays", 1994 International Conf. On Wafer Scale Integration.

[36] GOEL, P., "An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits", IEEE Tran. On Computers, vol. C-30, no.3 Mar. 1981.

[37] ALLAN, F. J. et al., "An approach to the diagnosability analysis of a system", IEEE Trans. On Computers, C-24:1040-1042, Aug. 1975.

[38] EVEN, S. et al, "Network flow and testing graph connectivity", SIAM J. Computing, 5:507-518, Aug. 1983.

[39] FRIEDMAN, A. D., "A new measure of digital system diagnosis", In Dig. 1975 Int. Symp. On fault tolerant computation, pp. 167-169, Jun. 1975.

[40] Private conversation.

[41] NORMAN, R. S., U.S. Patent, "Efficient Direct Replacement Cell Fault Tolerant Architecture", no.6,154,855, Nov.28, 2000.

[42] NORMAN, R. S. et al., "Communications Bus for a Parallel Processing System", Patent Pending.

[43] http: www.cmc.ca.

[44] QIU, B., "Diagnosis and Yield Analysis of the Complex Interconnection Architecture," M.ASc.thesis, Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada.

[45] KERMOUCHE, R., SAVARIA, Y., and AUDET, D., "Harvest Model of an Integrated Hierachical-Bus Architecture", 1994 Proceedings, Sixth Annual IEEE Intern. Conf. On Wafer Scale Integration, pp.69-78.

[46] SAVARIA, Y., "Parallel Microprocessor Architecture", United States Patent 5,276,893, Jan. 1994.

[47] KERMOUCHE, R., and SAVARIA, Y., "Defect and Fault Tolerant Scan Chains", Intern. Workshop on Defect and Fault Tolerant in VLSI Systems, pp.185-193, 1994.

[48] OTTERSTEDT, J, et al, "Test and Reconfiguration Experiments for Defect Tolerant Large Area Monolithic Multiprocessor System", 1994 Proceedings, Sixth Annual IEEE Intern. Conf. On Wafer Scale Integration, pp.315-323.

[49] BARDELL, P. H., MAANNEY, W. H., and SAVIR, J., "Build-in-Self-Test for VLIS: Pseudo random Techniques", John Wiley & Sons, New York 1987.

[50] PARKER, K. P. , "The BOUNDARY-SCAN HANDBOOK", ISBN 0-7923-8277-3.

[51] LANDIS, D. L., "A Self-test Methodology for Restructurable WSI", 1992 International Conference on WSI, pp.258-263.

[52] MAUNDER, C. and TULLOSS, R. E., "The Test Access Port and Boundary Scan Architecture", IEEE Computer Society Press Tutorial, ISBN 0-8186-9070-4

ANNEXE A

REPLACEMENT EXAMPLES OF A 3X3 ARRAY

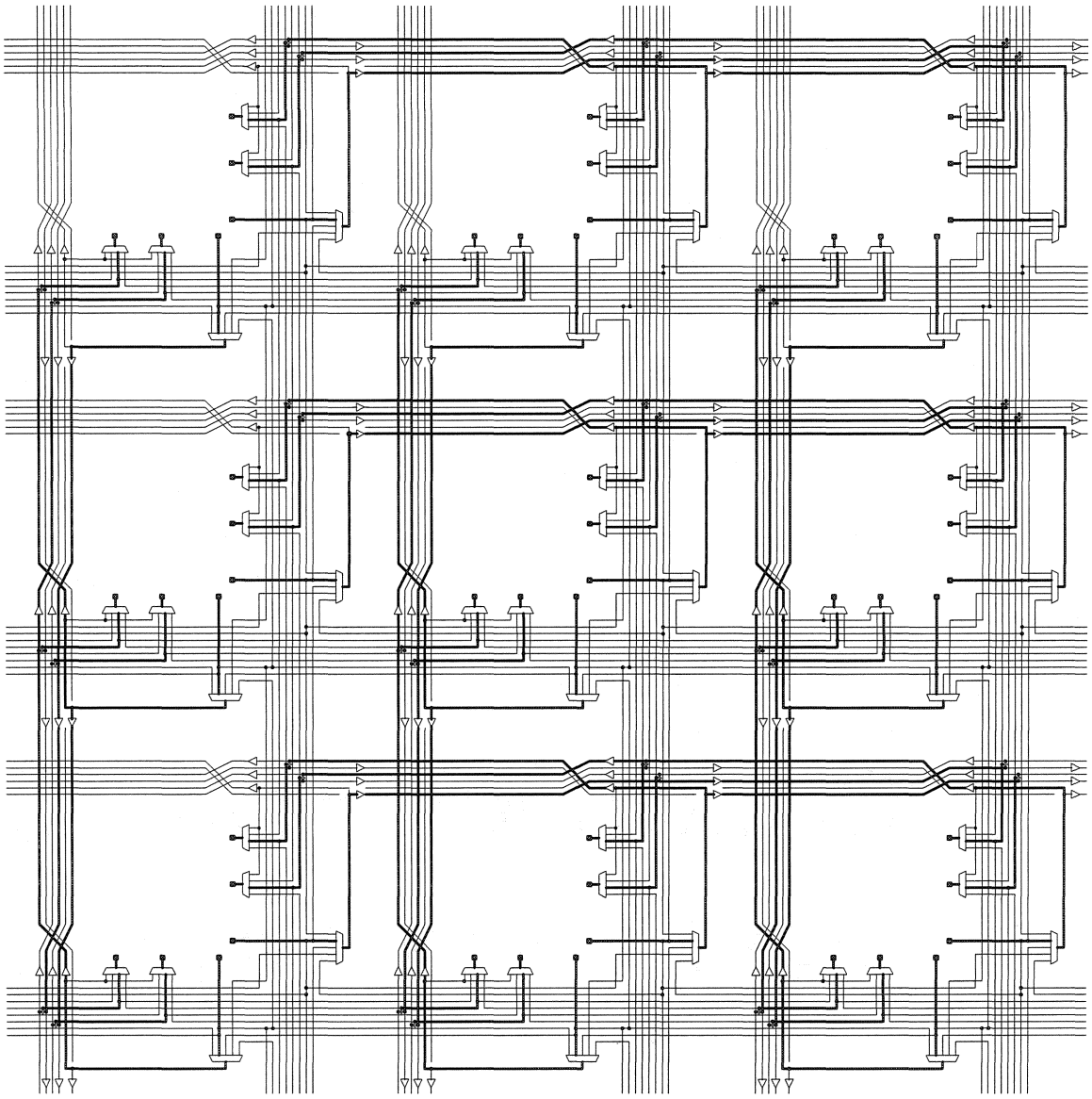


Figure A.1 Default configuration

All cells connect to their natural bundles

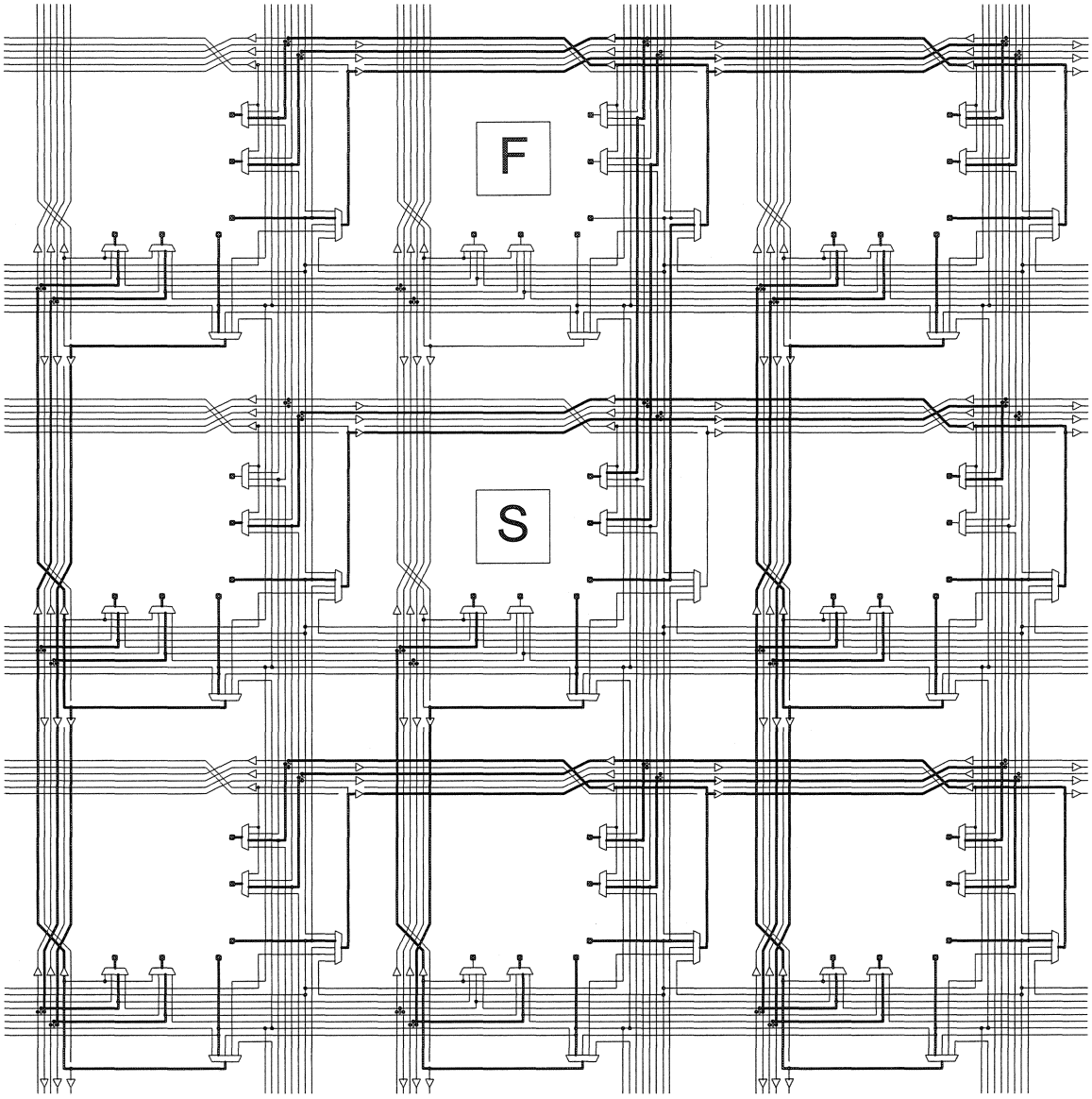


Figure A.2 Cell replacement (up shift) R_C_U[(1,0),1]

Cell S replaces cell F. Cell S takes over the connectivity to horizontal bus of cell F.

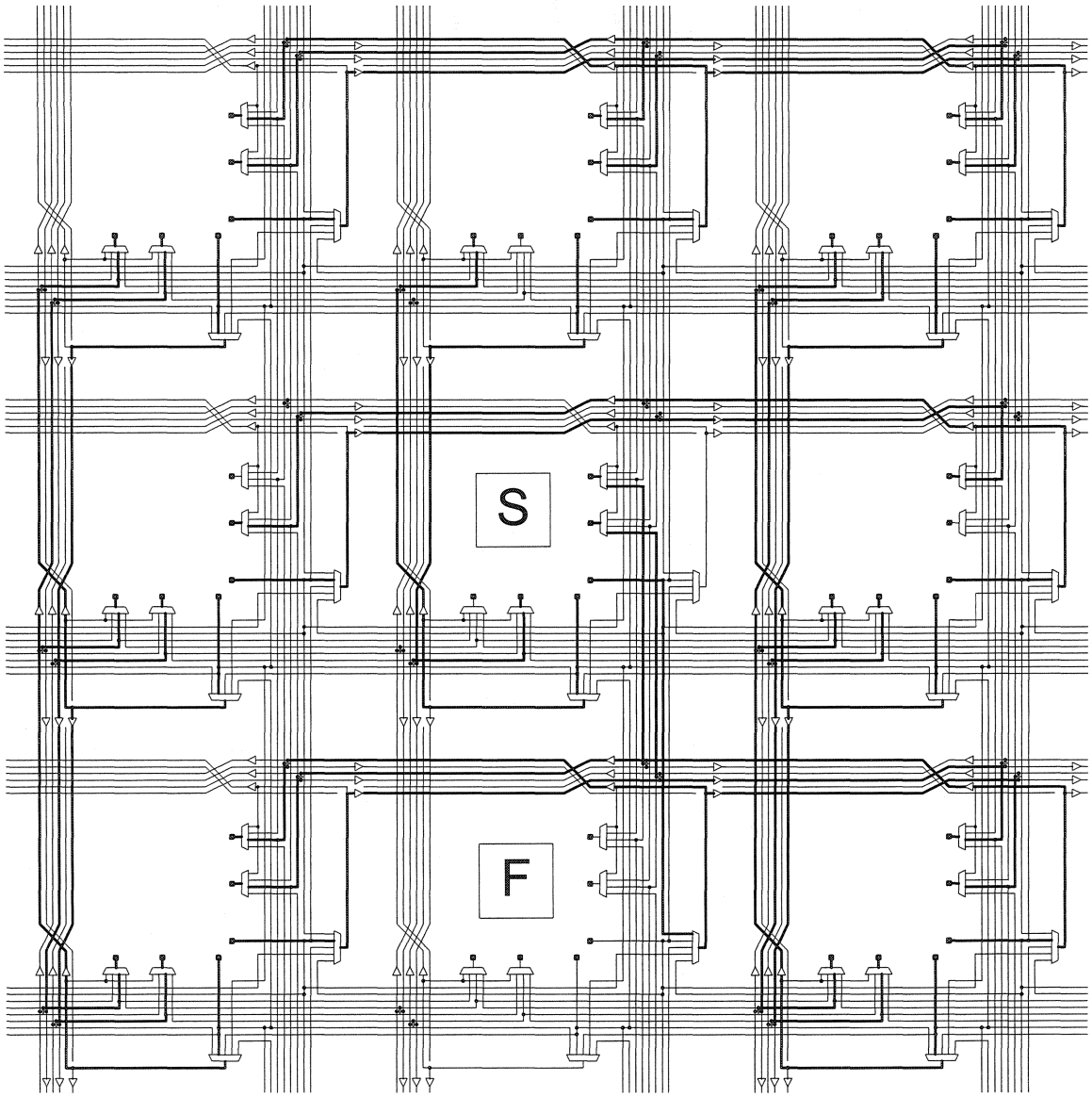


Figure A.3 Cell replacement (down shift) R_C_D[(1,2),1]

Cell S replaces cell F. Cell S takes over the connectivity to horizontal bus of cell F.

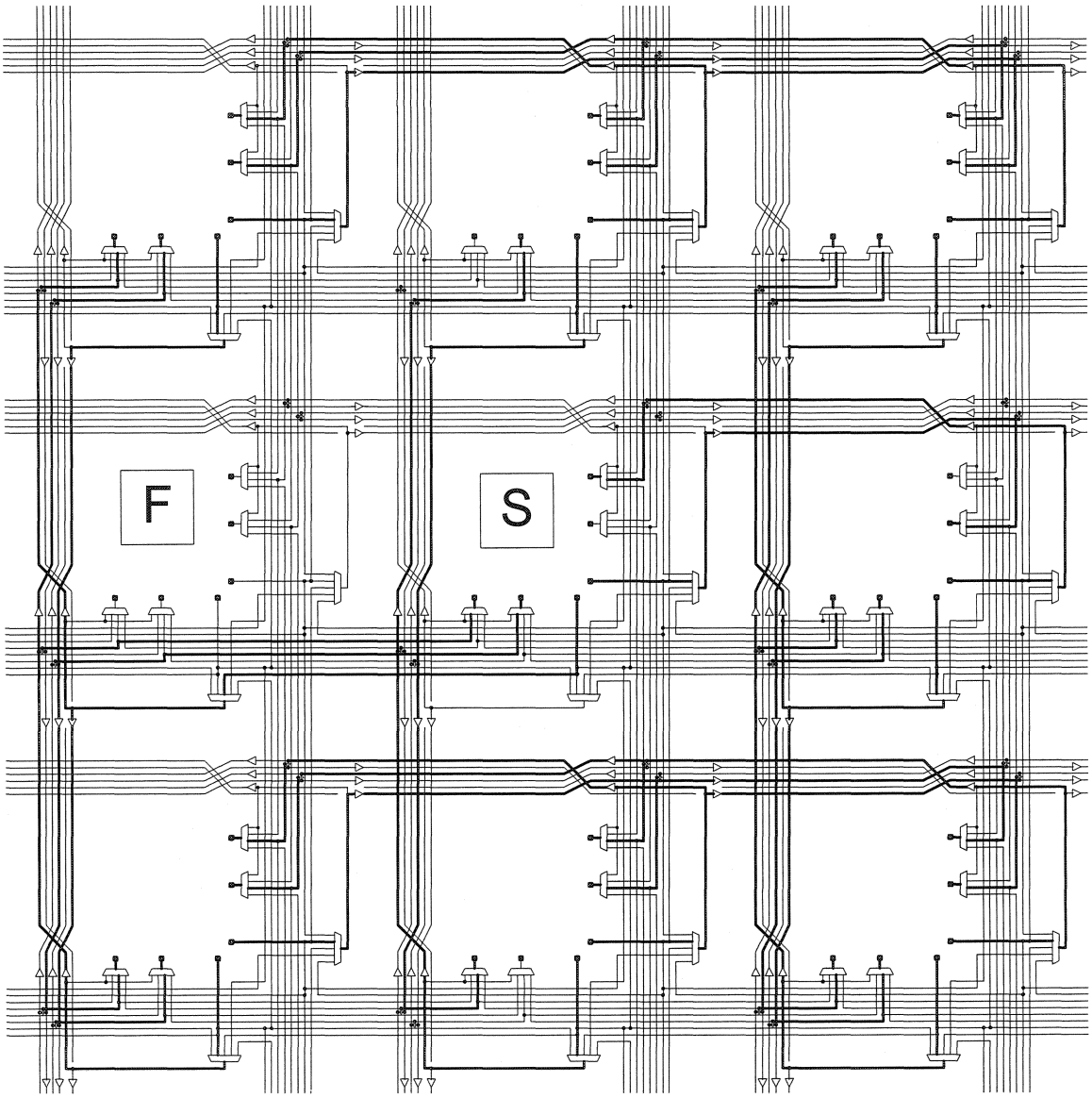


Figure A.4 Cell replacement (left shift) R_C_L[(0,1),1]

Cell S replaces cell F. Cell S takes over the connectivity to vertical bus of cell F.

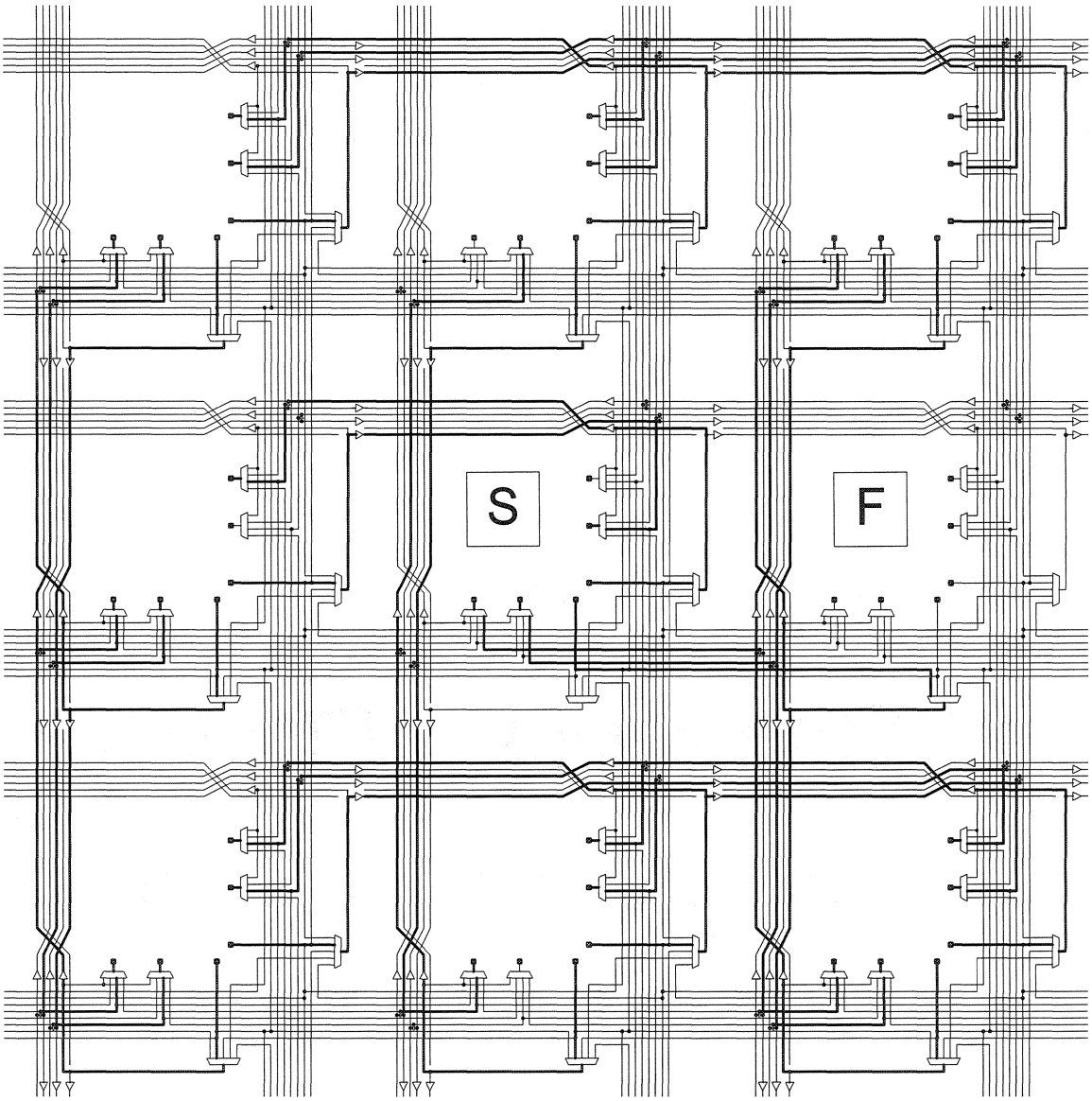


Figure A.5 Cell replacement (right shift) R_C_R[(2,1),1]

Cell S replaces Cell F. Cell S takes the connectivity to vertical bus of Cell F.

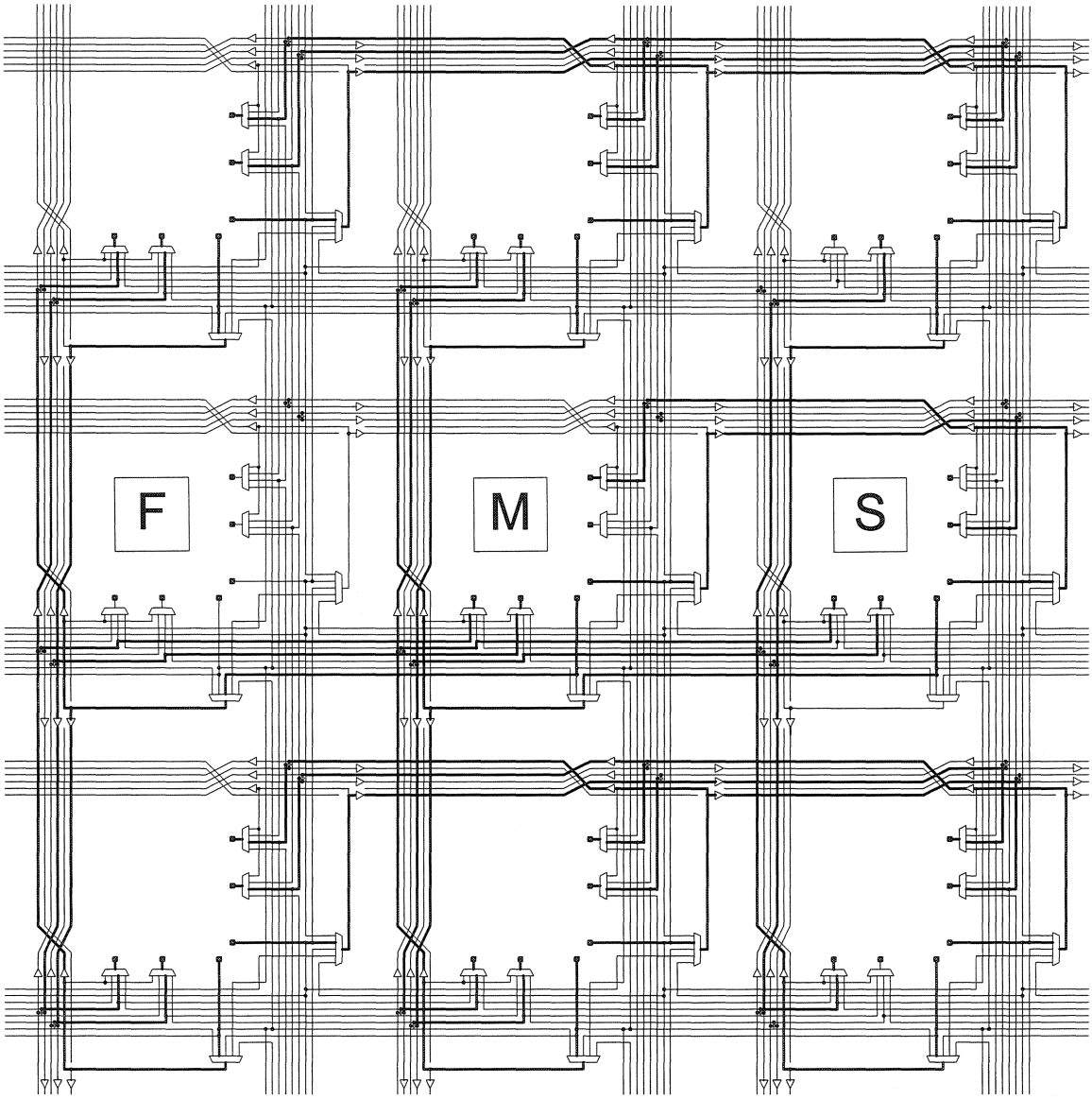


Figure A.6 Cell replacement (left shift) R_C_L[(0,1),2]

Cell S replaces cell F. The replacement is achieved by a replacement ripple. Cell S replaces cell M; cell M replaces cell F. Good cell is shifted to left.

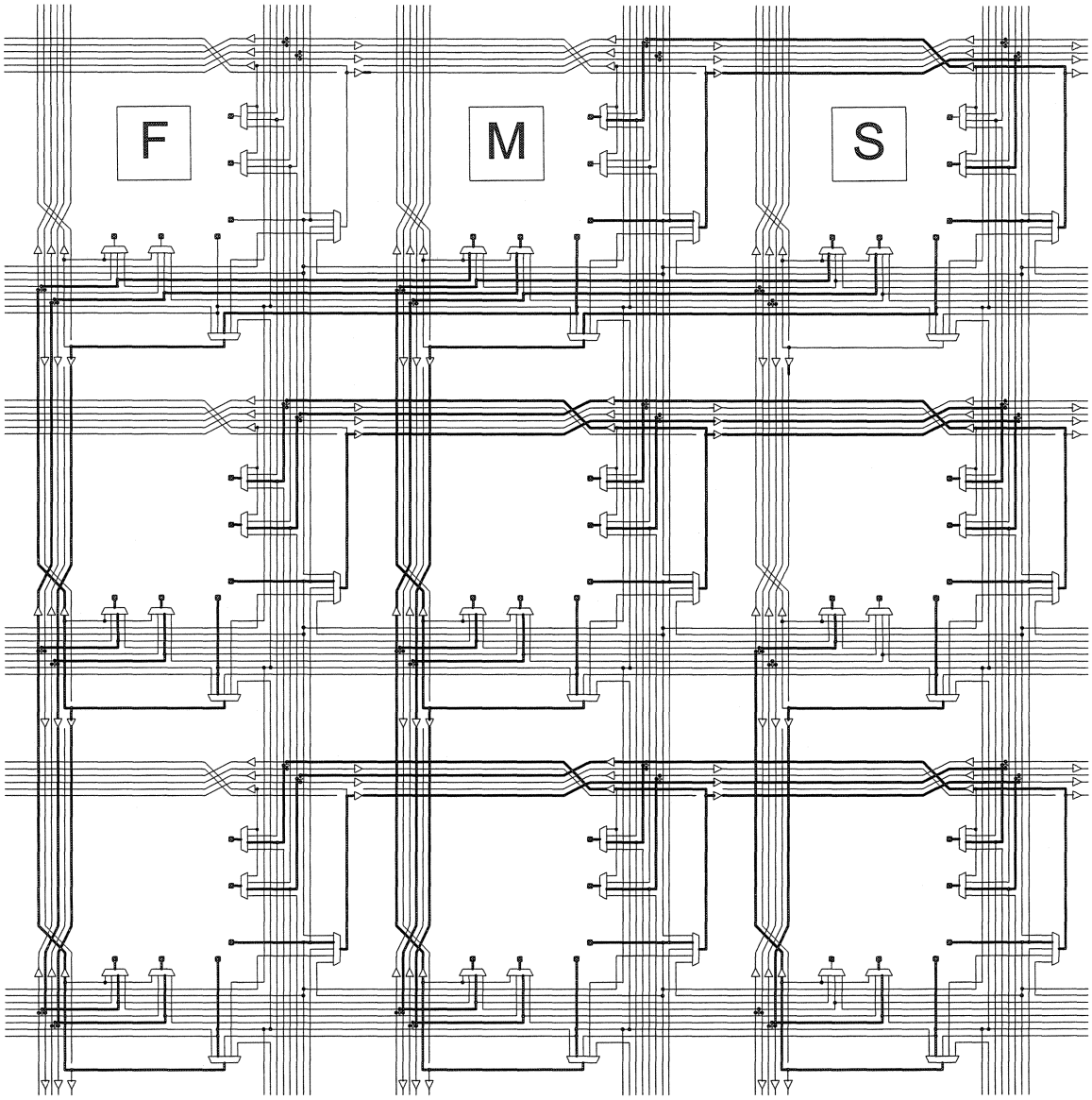


Figure A.7 Cell replacement (left shift) R_C_L[(0,0),2]

Cell S replaces cell F. The replacement is achieved by a replacement ripple. Cell S replaces cell M; cell M replaces cell F. Good cell is shifted to left.

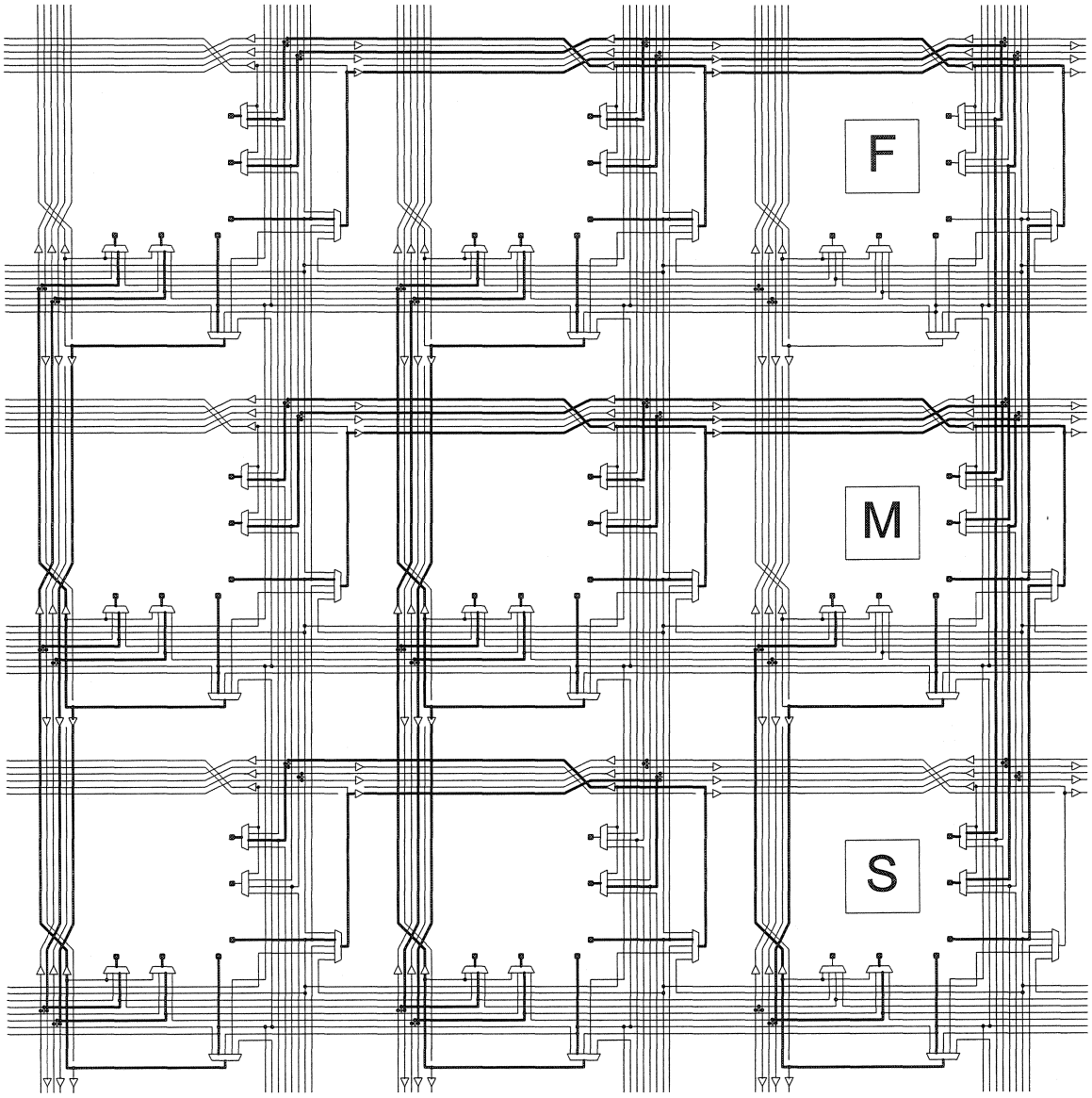


Figure A.8 Cell replacement (up Shift) R_C_U[(2,0),2]

Cell S replaces cell F. The replacement is achieved by a replacement ripple. Cell S replaces cell M; cell M replaces cell F. Good cell is shifted up

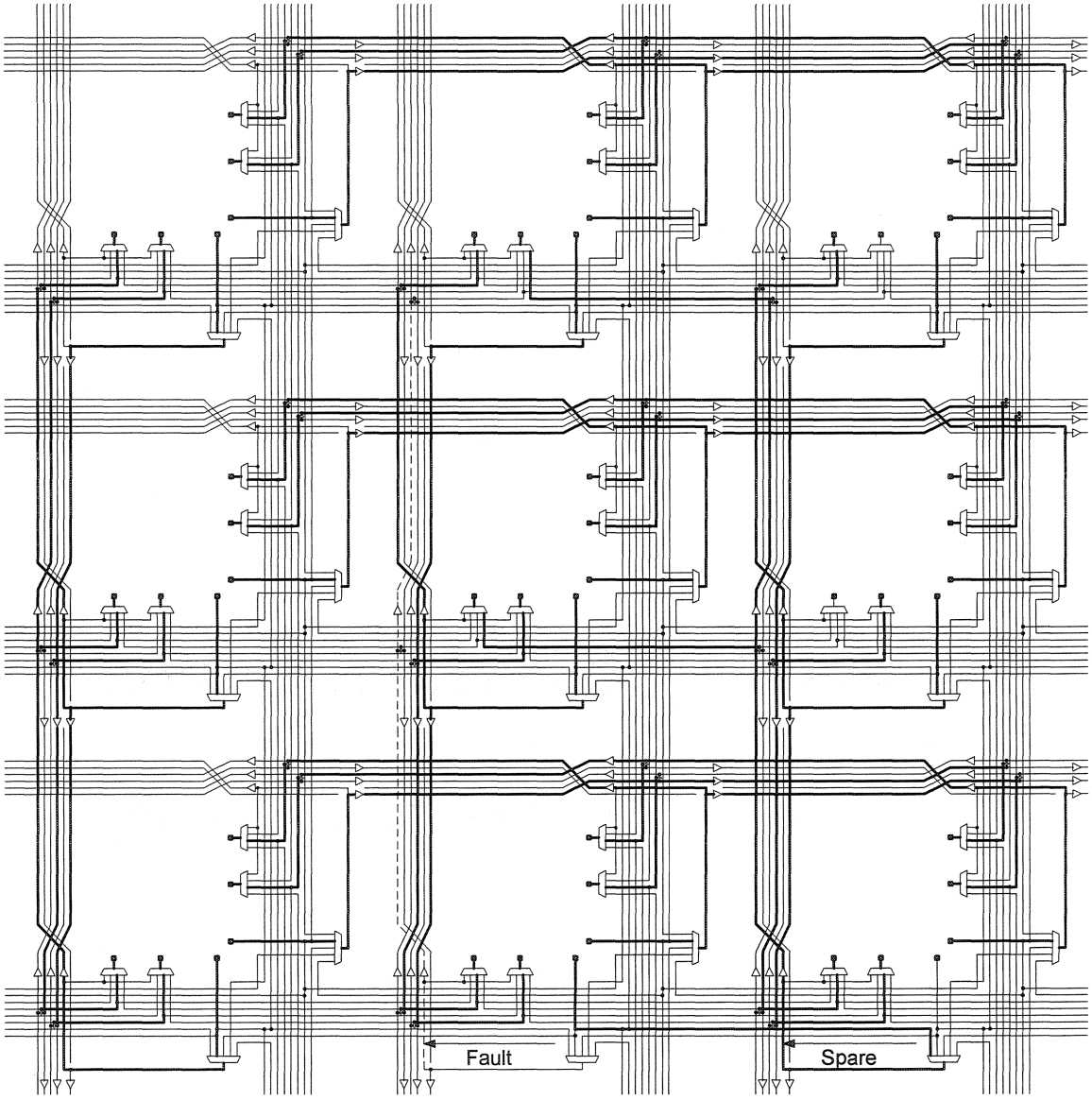


Figure A.9 Vertical Inter-bus Bundle Replacement (left shift) R_B_V_L[(1,2),1]

Replace a vertical bundle with the neighbouring bundle in the same bus

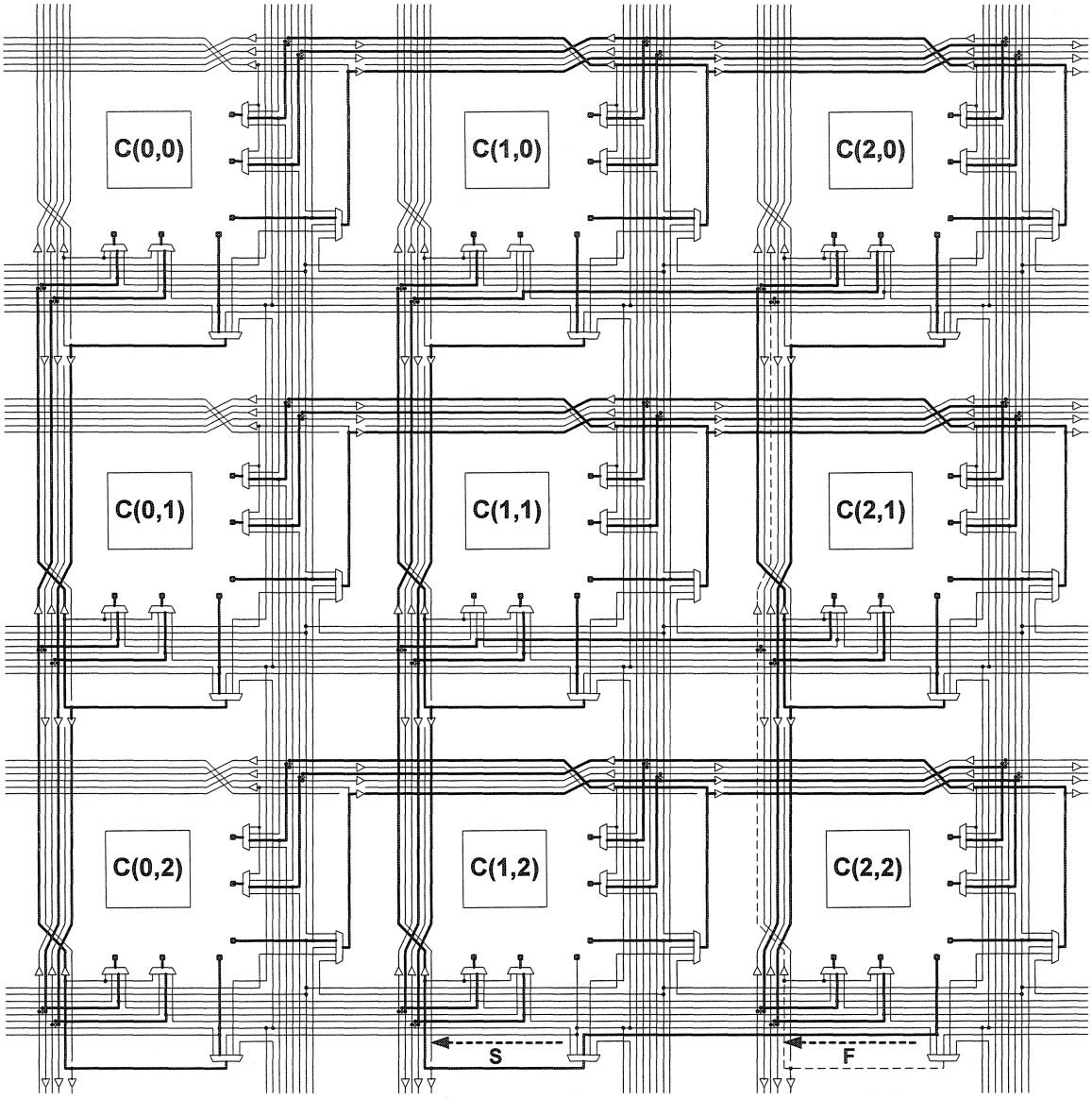


Figure A.10 Vertical Inter-bus Bundle Replacement (right shift) R_B_V_R[(2,2),1]

Bundle S replaces bundle F. After the replacement, cell C(2,2)talks to cell C(1,2)'s bundle, cell C(2,0) and cell C(2,1) listen to that bundle. Good bundle is shifted right.

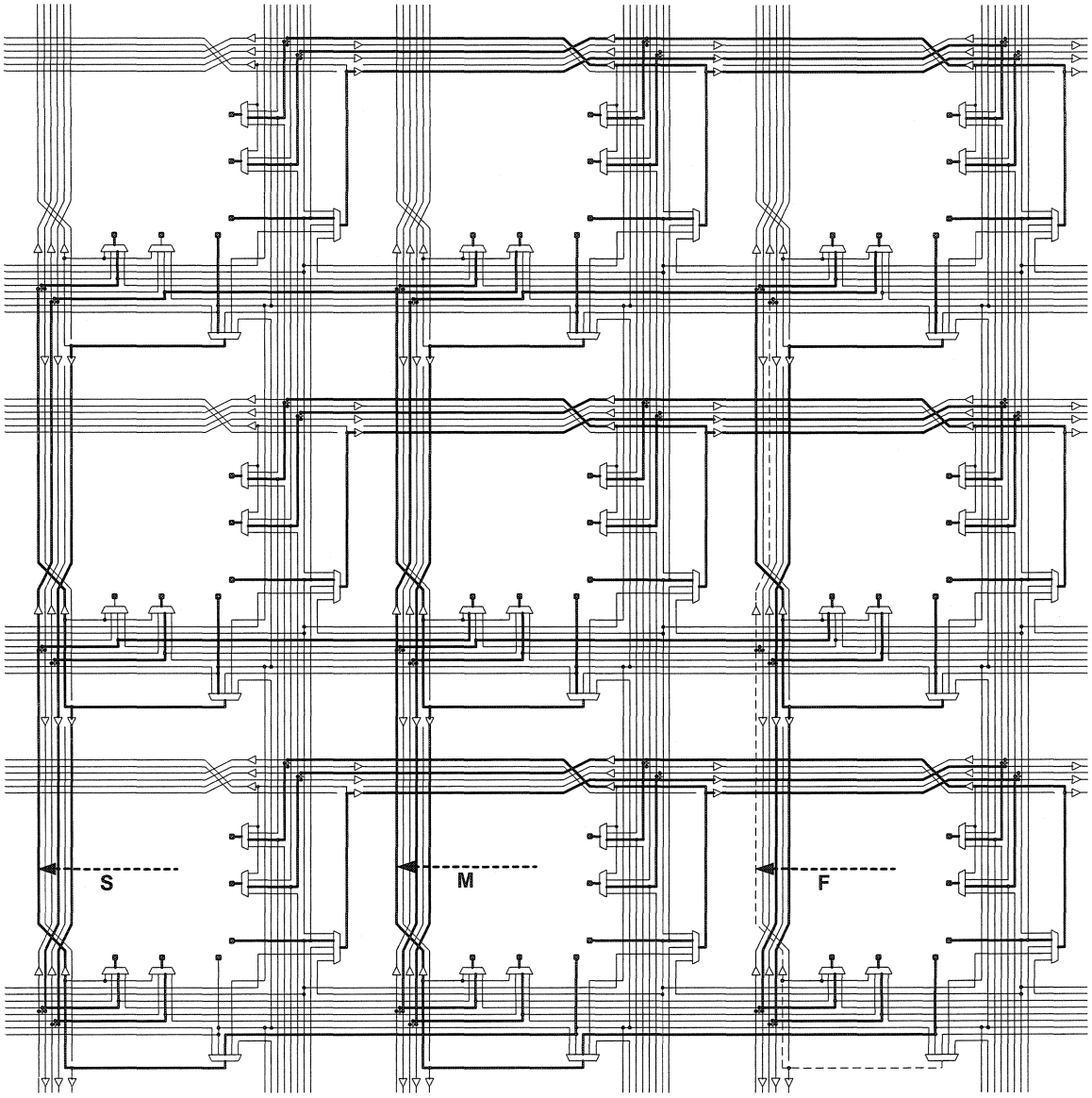


Figure A.11 Vertical Inter-bus Bundle Replacement (right shift) R_B_V_R[(2,2),2]

Bundle S replaces bundle F. The replacement is achieved by replacement ripple. Bundle S replaces bundle M; bundle M replaces bundle F. Good bundle is shifted right.

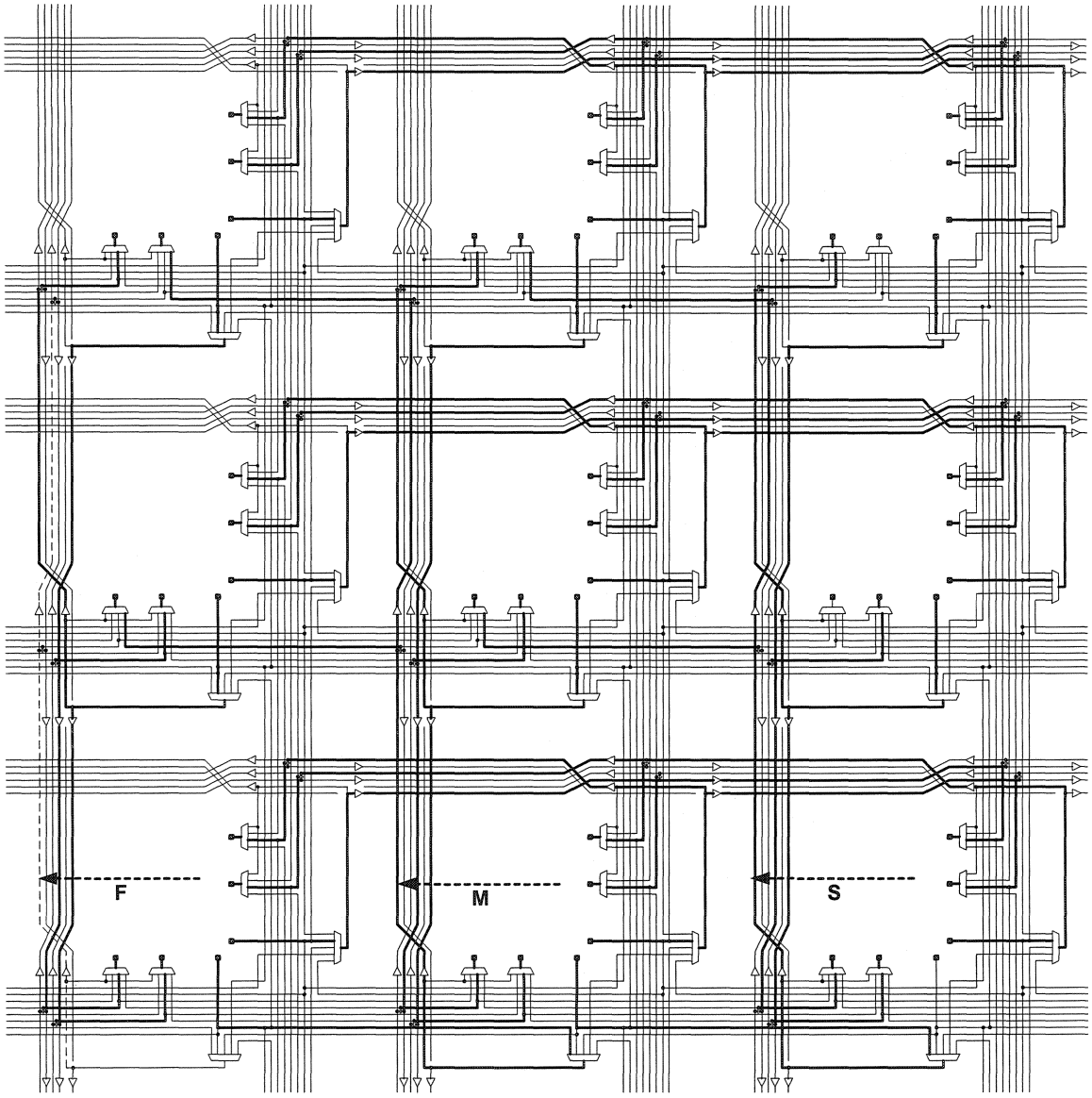


Figure A.12 Vertical Inter-bus Bundle Replacement (left shift) R_B_V_L[(0,2),2]

Bundle S replaces bundle F. The replacement is achieved by replacement ripple. Bundle S replaces bundle M; bundle M replaces bundle F Good bundle is shifted left.

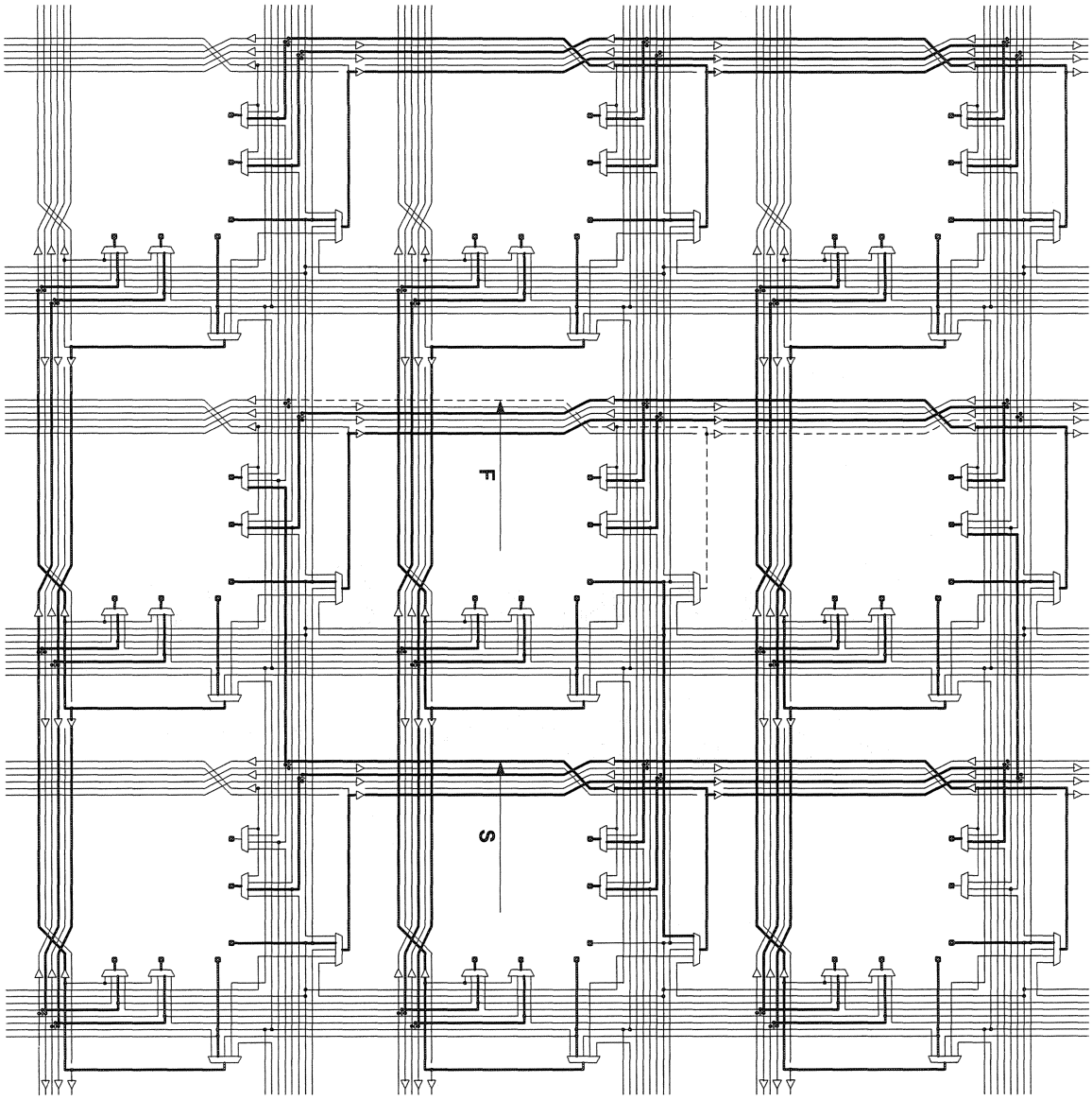


Figure A.13 Horizontal Inter-bus Bundle Replacement (up shift) R_B_H_U[(1,1),1]

Bundle S replaces bundle F. Good bundle is shifted up.

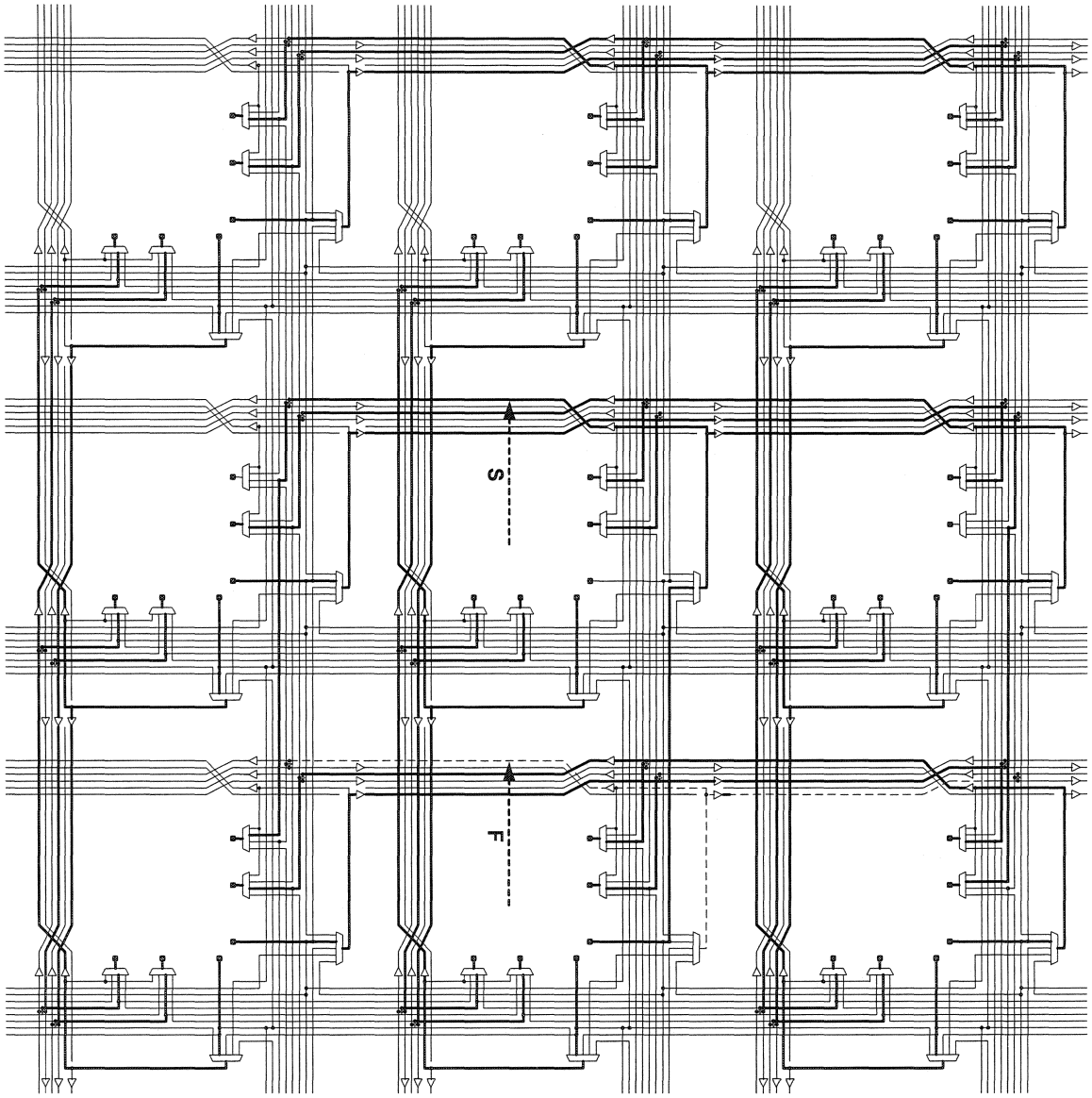


Figure A.14 Horizontal Inter-bus Bundle Replacement (down shift)

R_B_H_D[(1,2),1]

Bundle S replaces bundle F. Good bundle is shifted down.

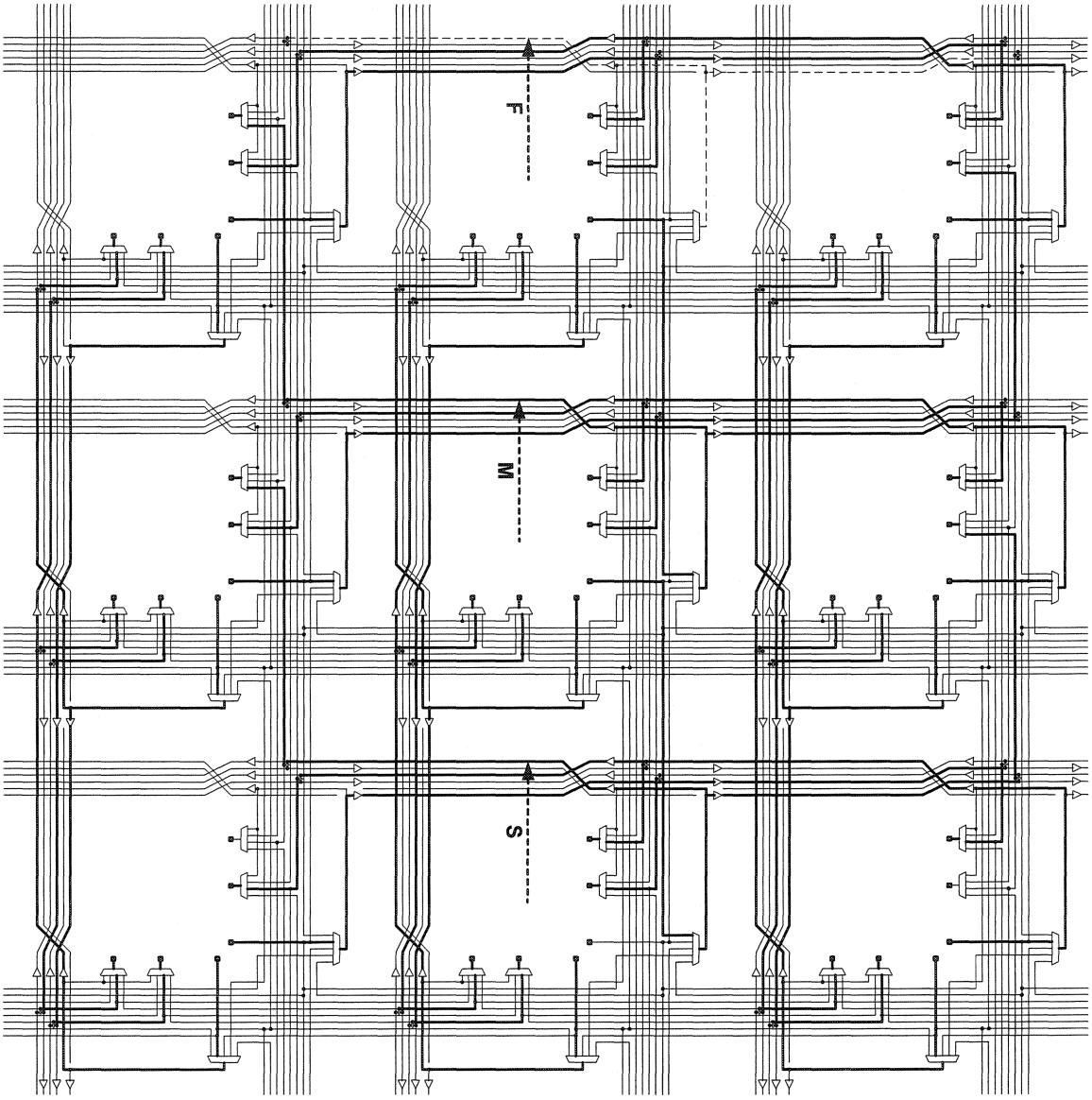


Figure A.15 Horizontal Inter-bus Bundle Replacement (left shift)
R_B_H_U[(1,0),2]

Bundle S replaces bundle F. The replacement is achieved by replacement ripple. Bundle S replaces bundle M; bundle M replaces bundle F Good bundle is shifted up.

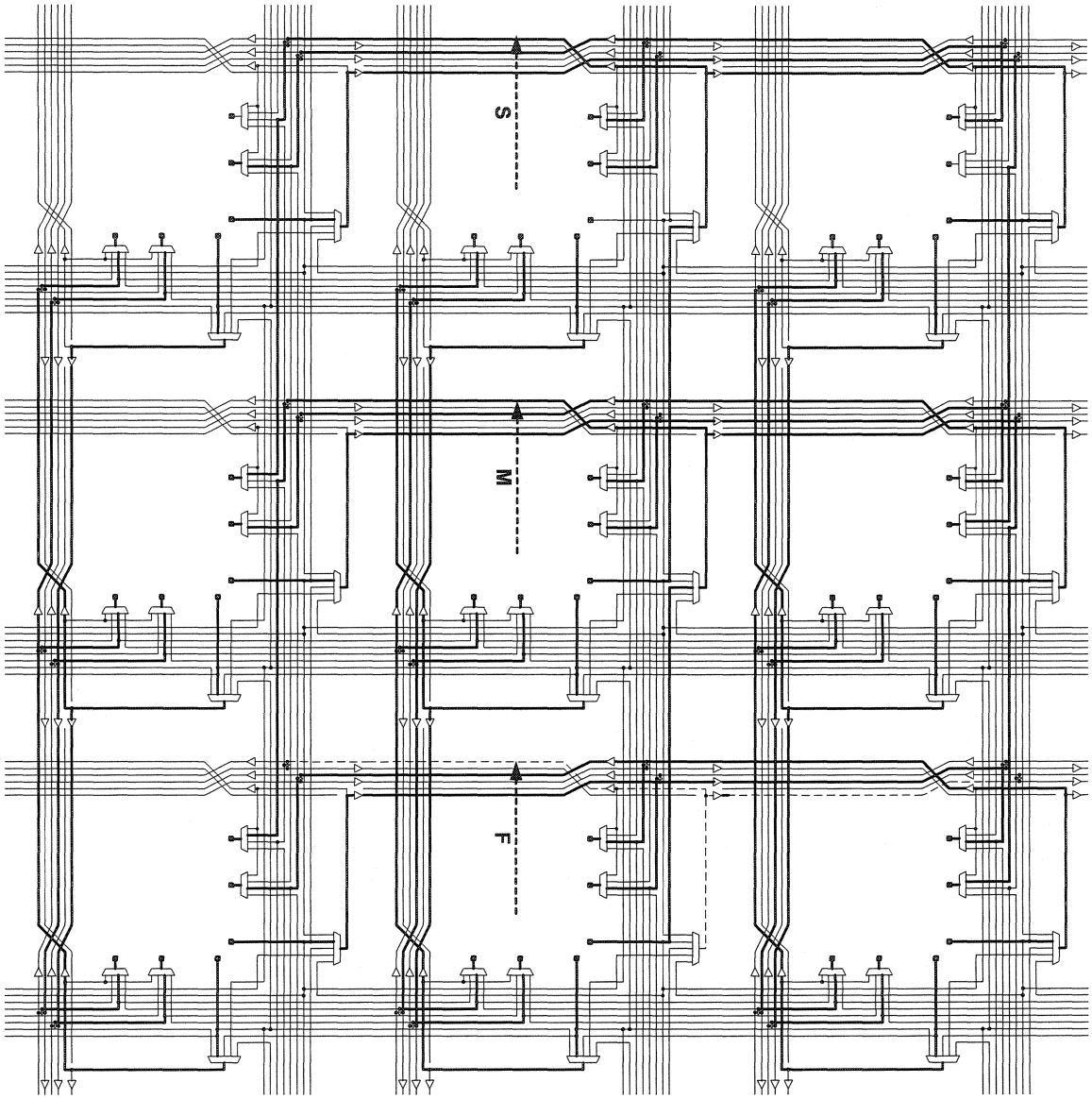


Figure A.16 Horizontal Inter-bus Bundle Replacement (down shift)

R_B_H_D[(1,2),2]

Bundle S replaces bundle F. The replacement is achieved by replacement ripple. Bundle S replaces bundle M; bundle M replaces bundle F Good bundle is shifted down.

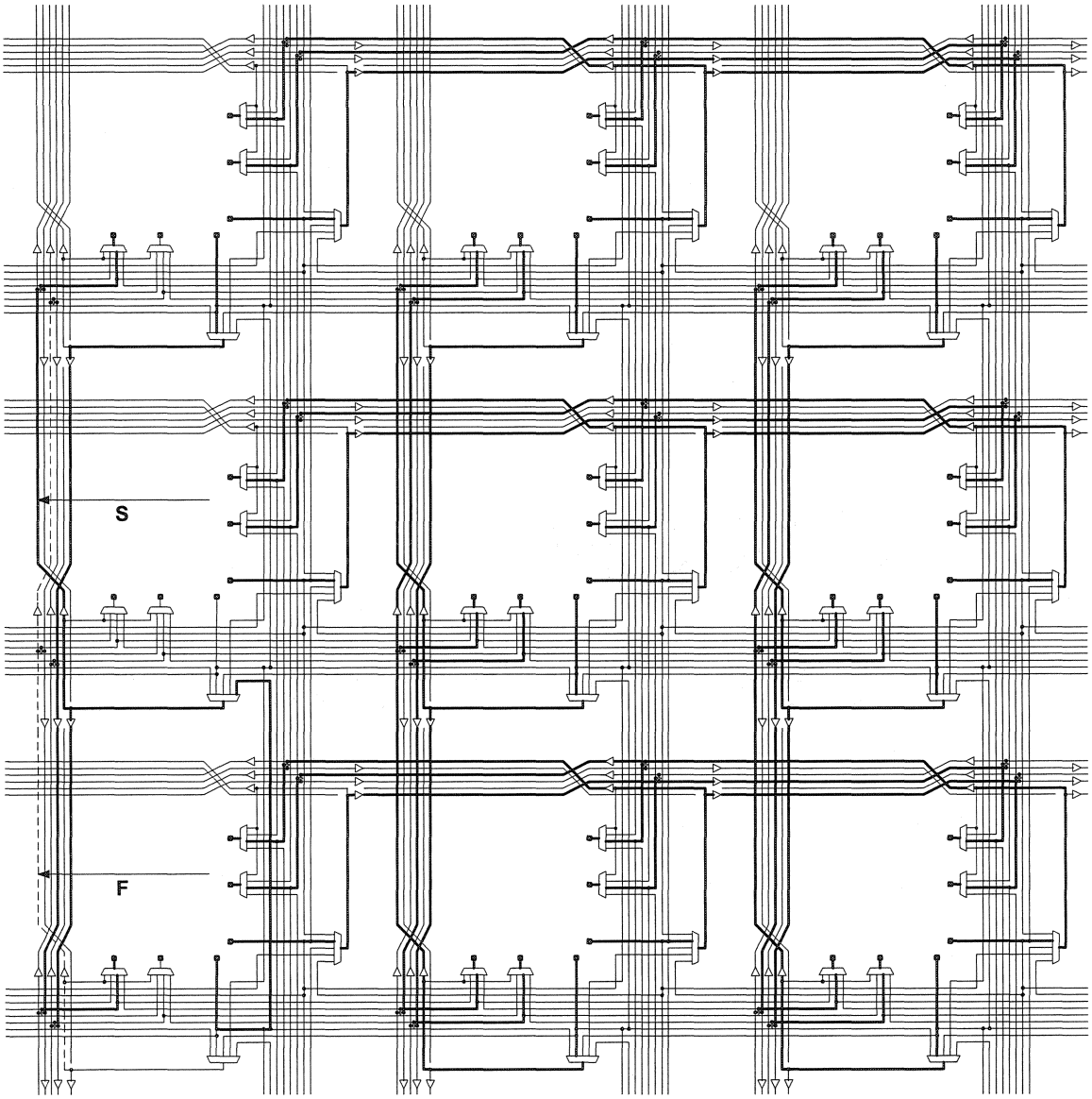


Figure A.17 Vertical Intra-bus Bundle Replacement (down shift)

R_B_V_D[(0,2),1]

Bundle S replaces bundle F. Good bundle is shifted down.

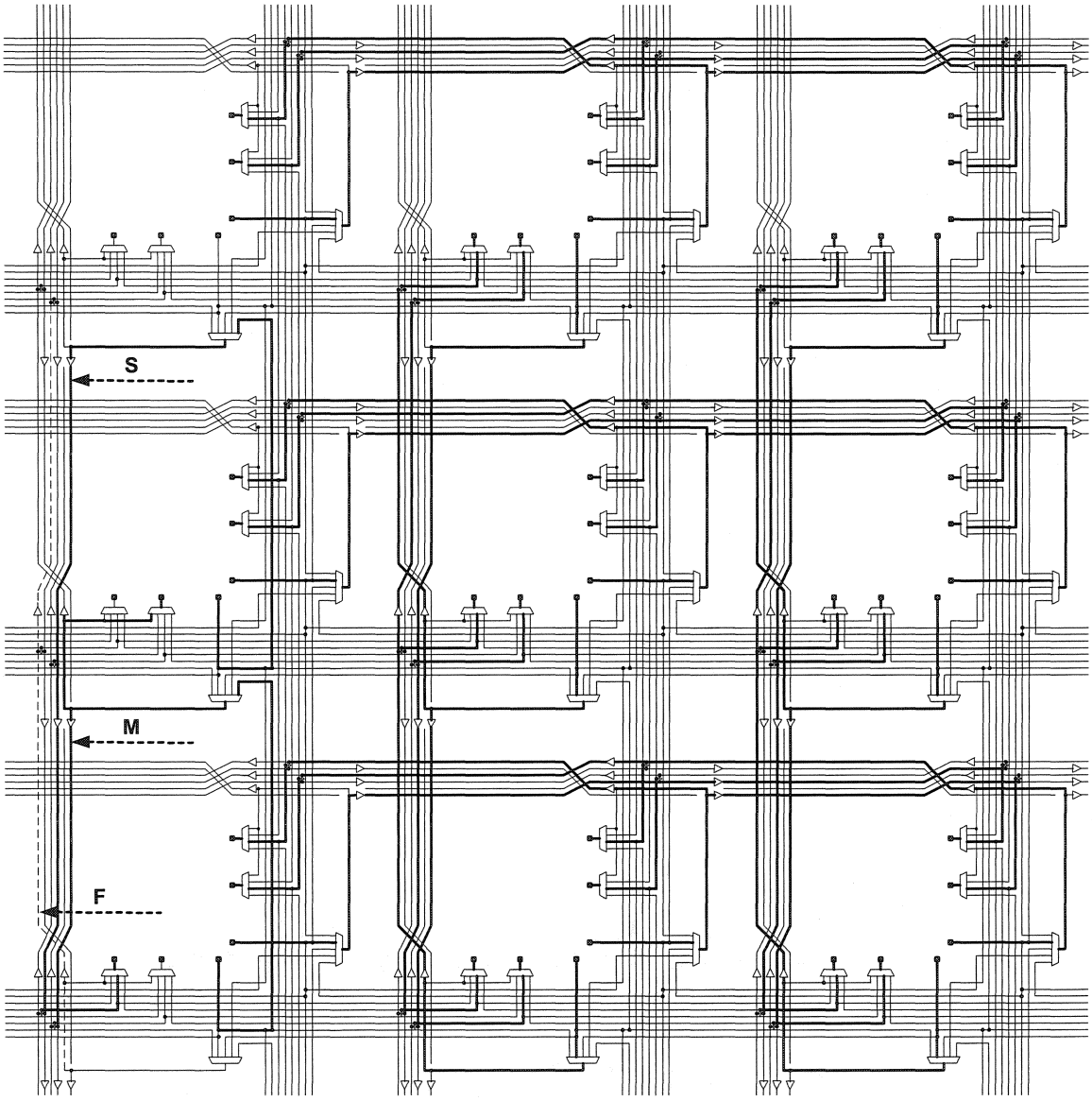


Figure A.18 Vertical Intra-bus Bundle Replacement (down shift)

R_B_V_D[(0,2),2]

Bundle S replaces bundle F. The replacement is achieved by replacement ripple. Bundle S replaces bundle M; bundle M replaces bundle F Good bundle is shifted down.

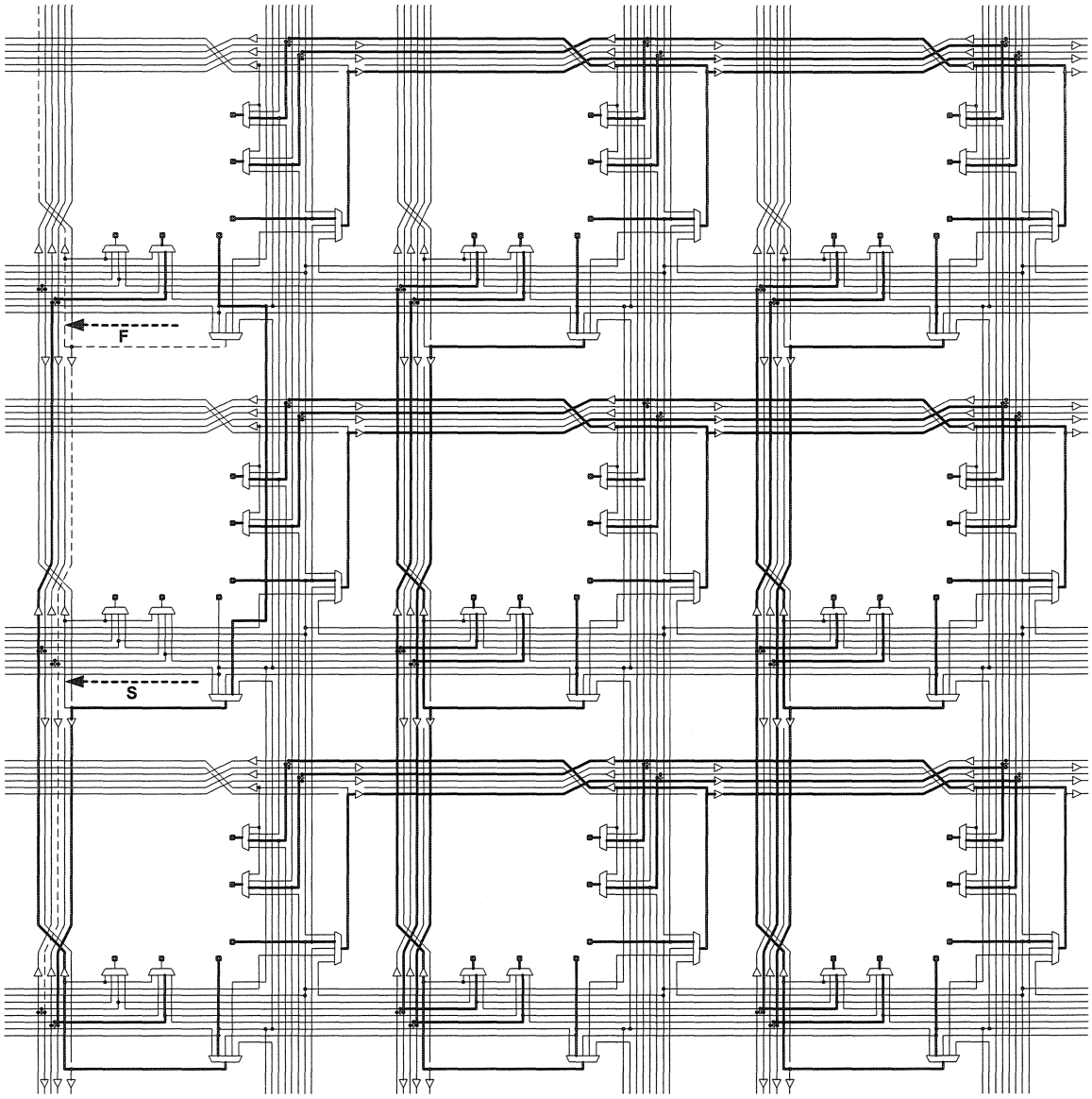


Figure A.19 Vertical Intra-bus Bundle Replacement (up shift) R_B_V_U[(0,0),1]

Bundle S replaces bundle F. Good bundle is shifted up.

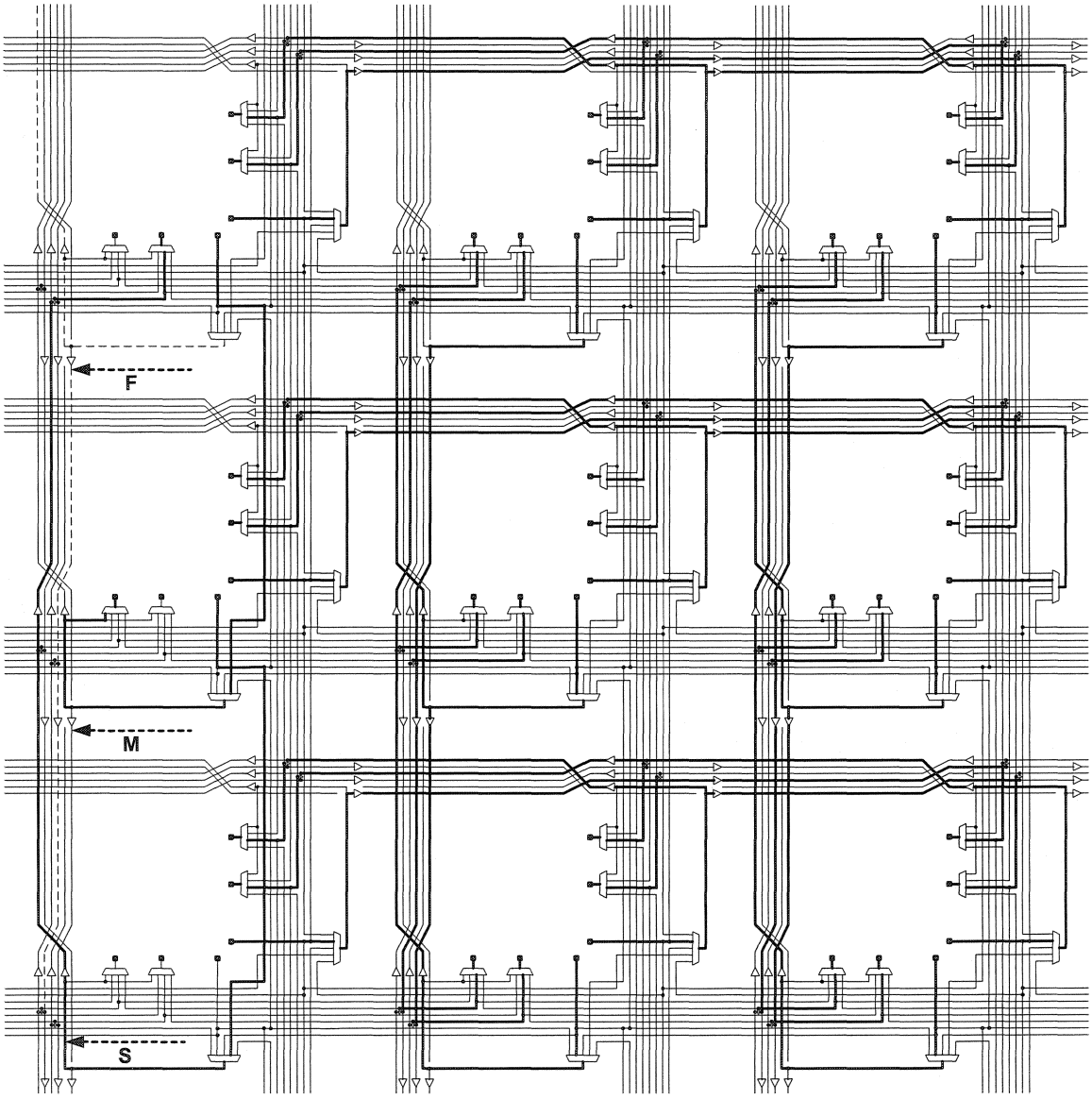


Figure A.20 Vertical Intra-bus Bundle Replacement (up shift) R_B_V_U[(0,0),2]

Bundle S replaces bundle F. The replacement is achieved by replacement ripple. Bundle S replaces bundle M; bundle M replaces bundle F Good bundle is shifted up.

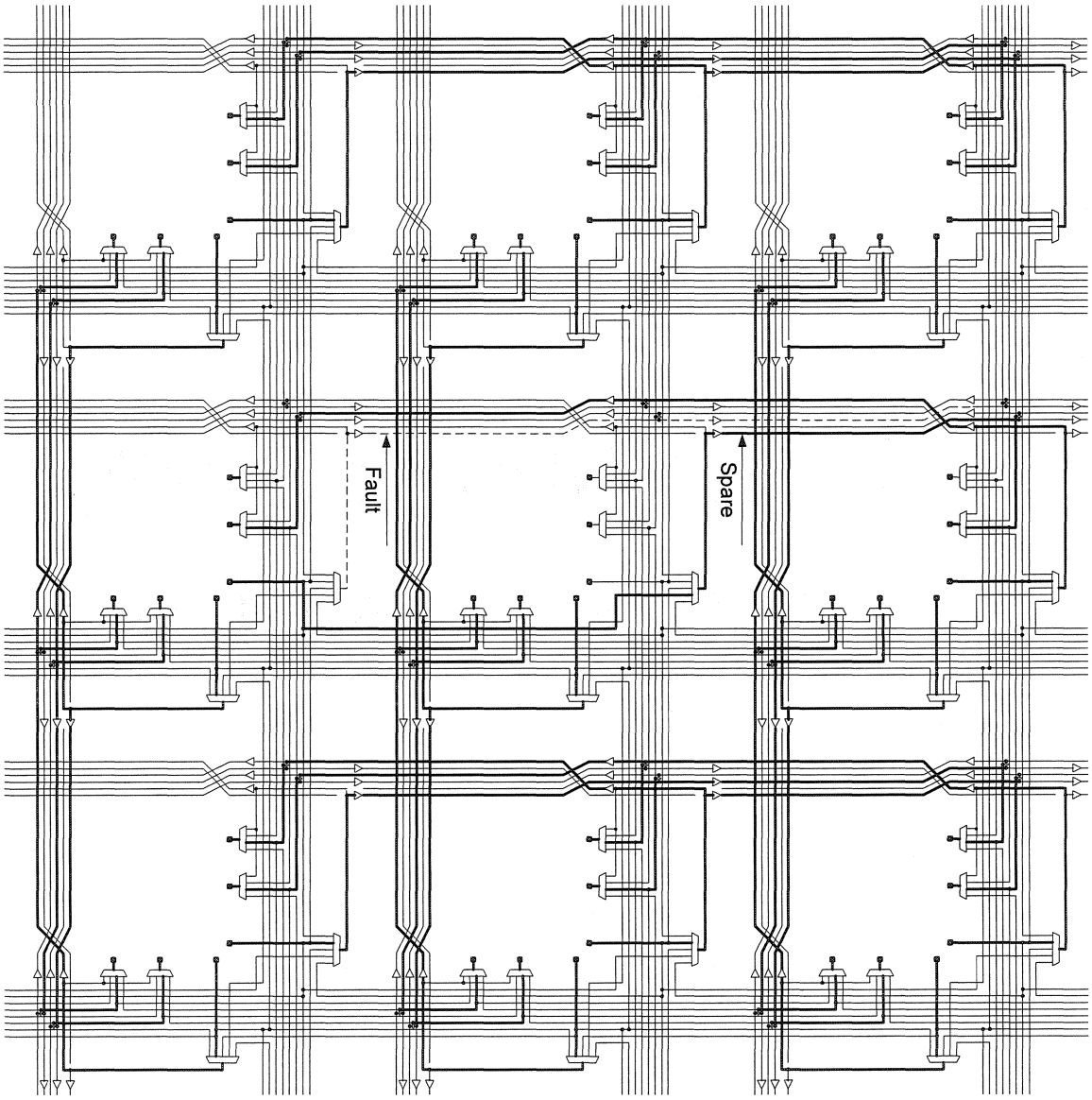


Figure A.21 Horizontal Intra-bus Bundle Replacement (left shift)

R_B_H_L[(0,1),1]

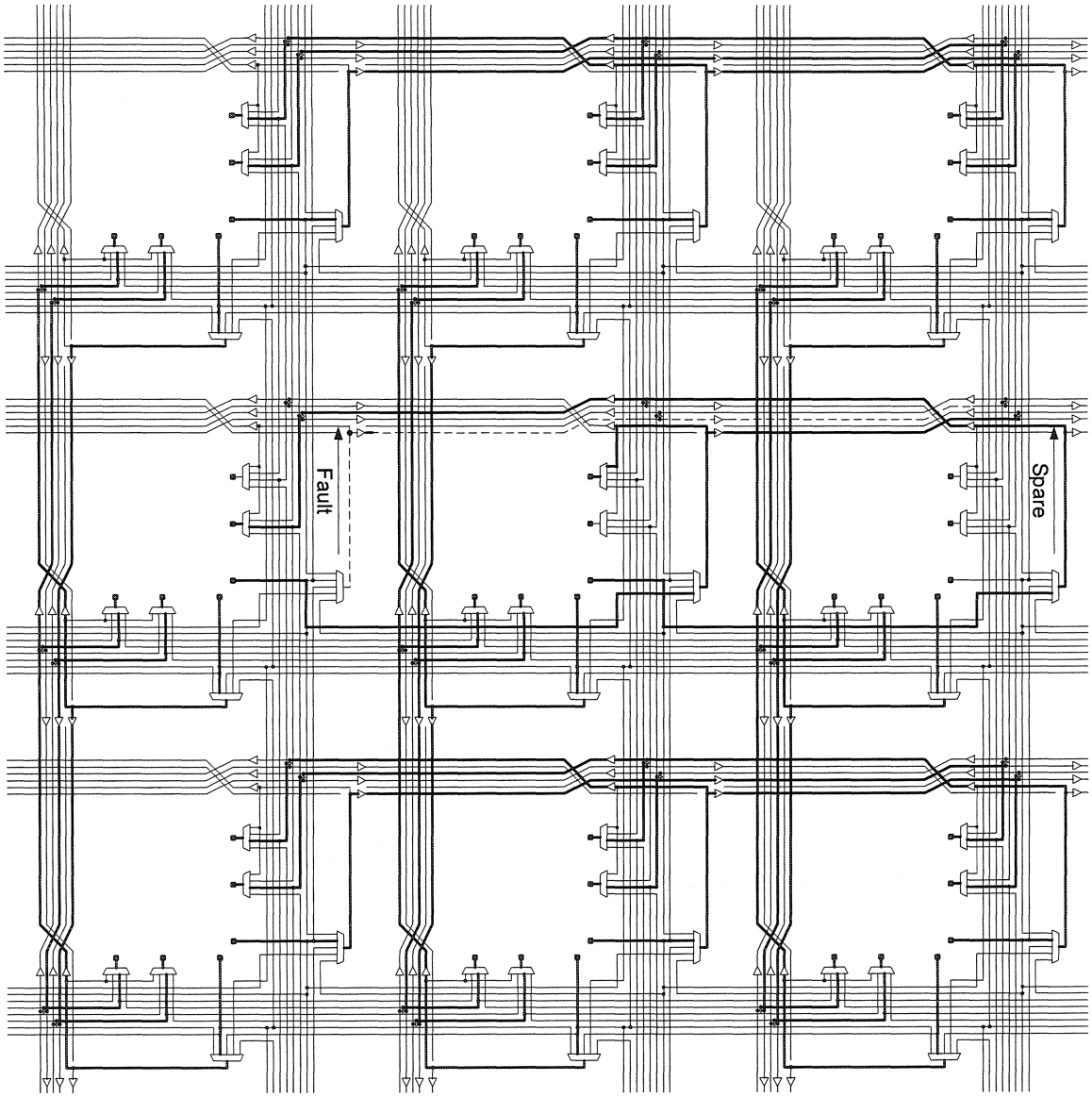


Figure A.22 Horizontal Intra-bus Bundle Replacement (left shift)

R_B_H_L[(0,1),2]

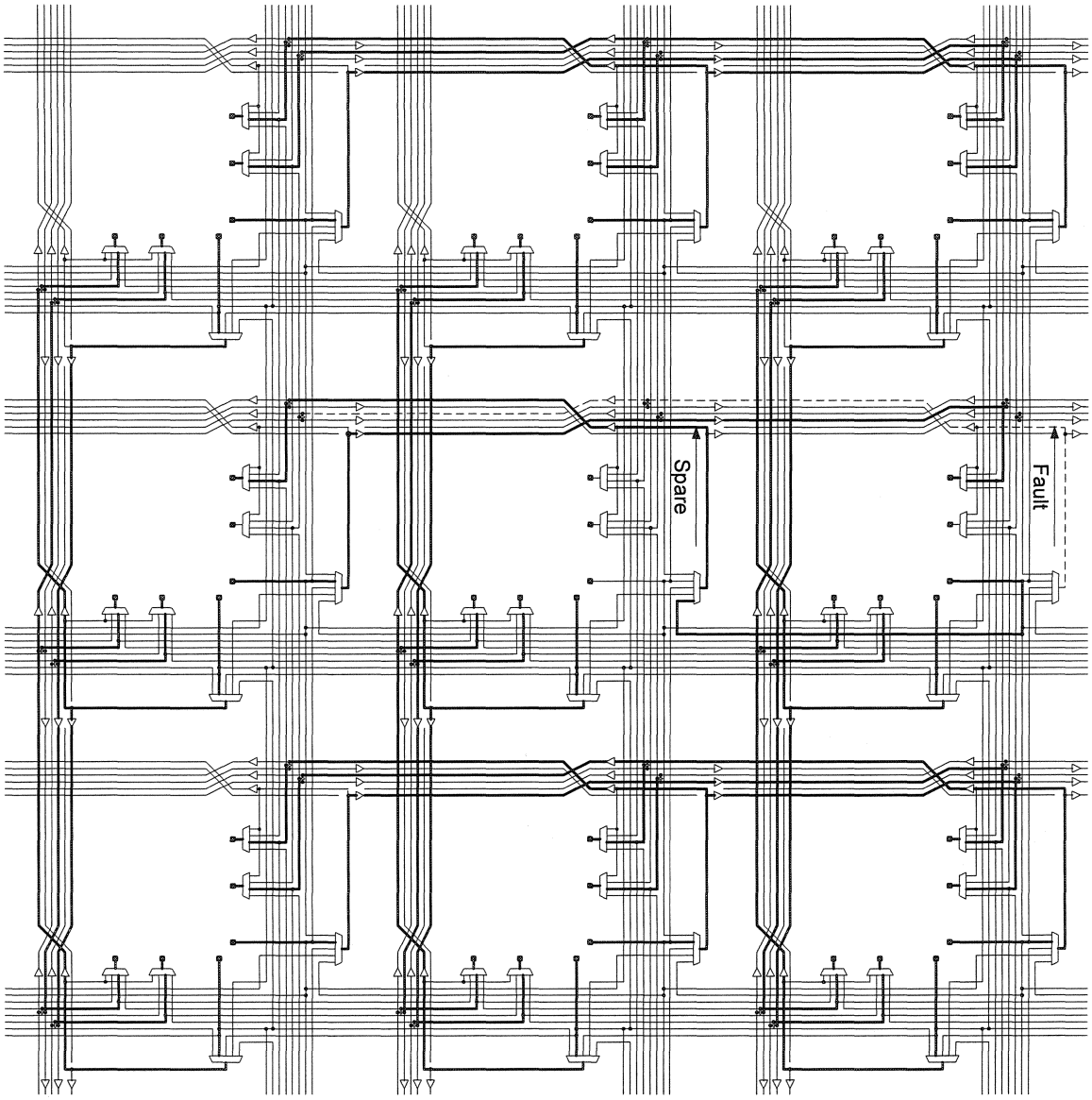


Figure A.23 Horizontal Intra-bus Bundle Replacement (right shift)

R_B_H_R[(2,1),1]

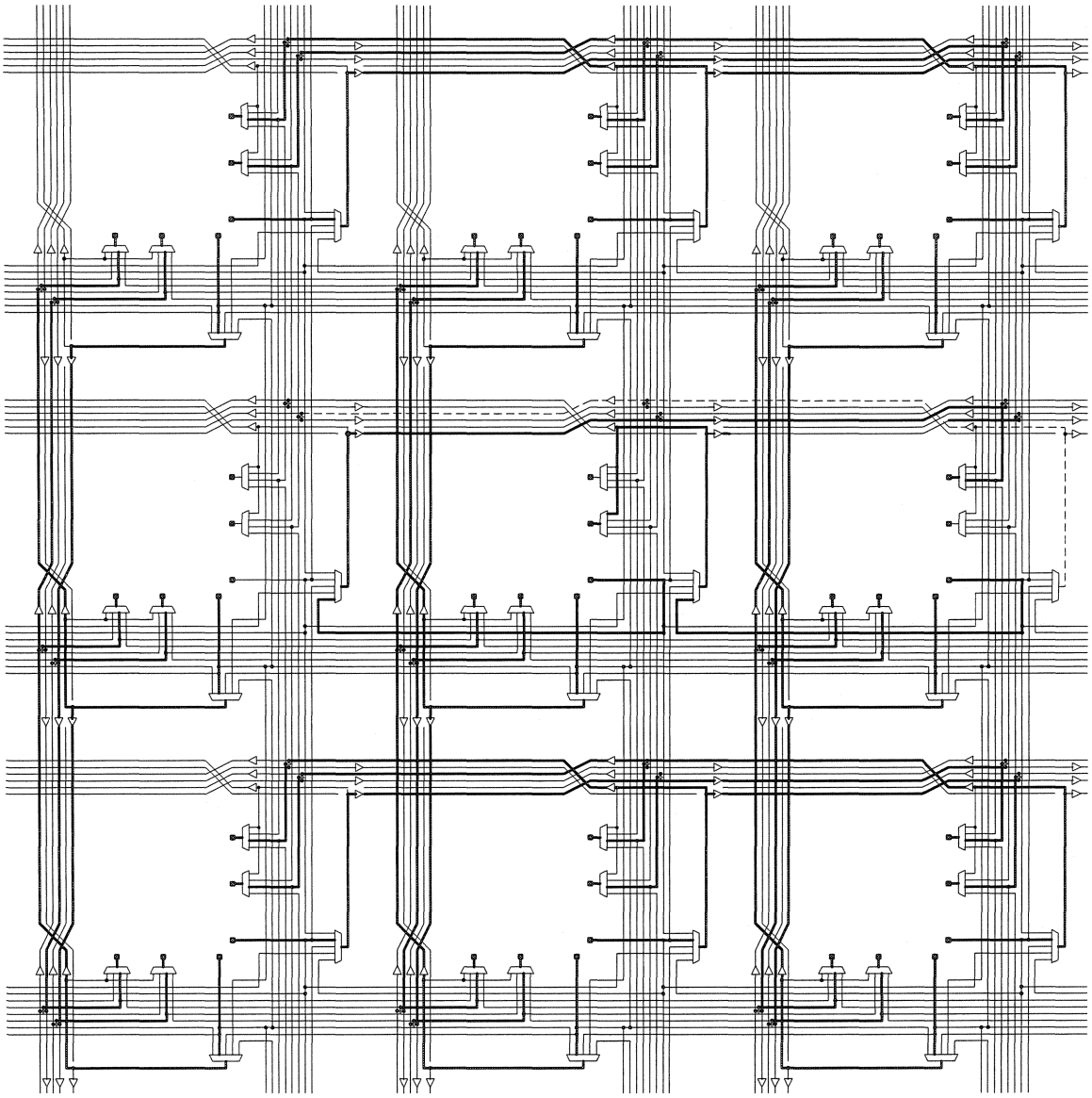


Figure A.24 Horizontal Intra-bus Bundle Replacement (right shift)

R_B_H_R[(2,1),2]

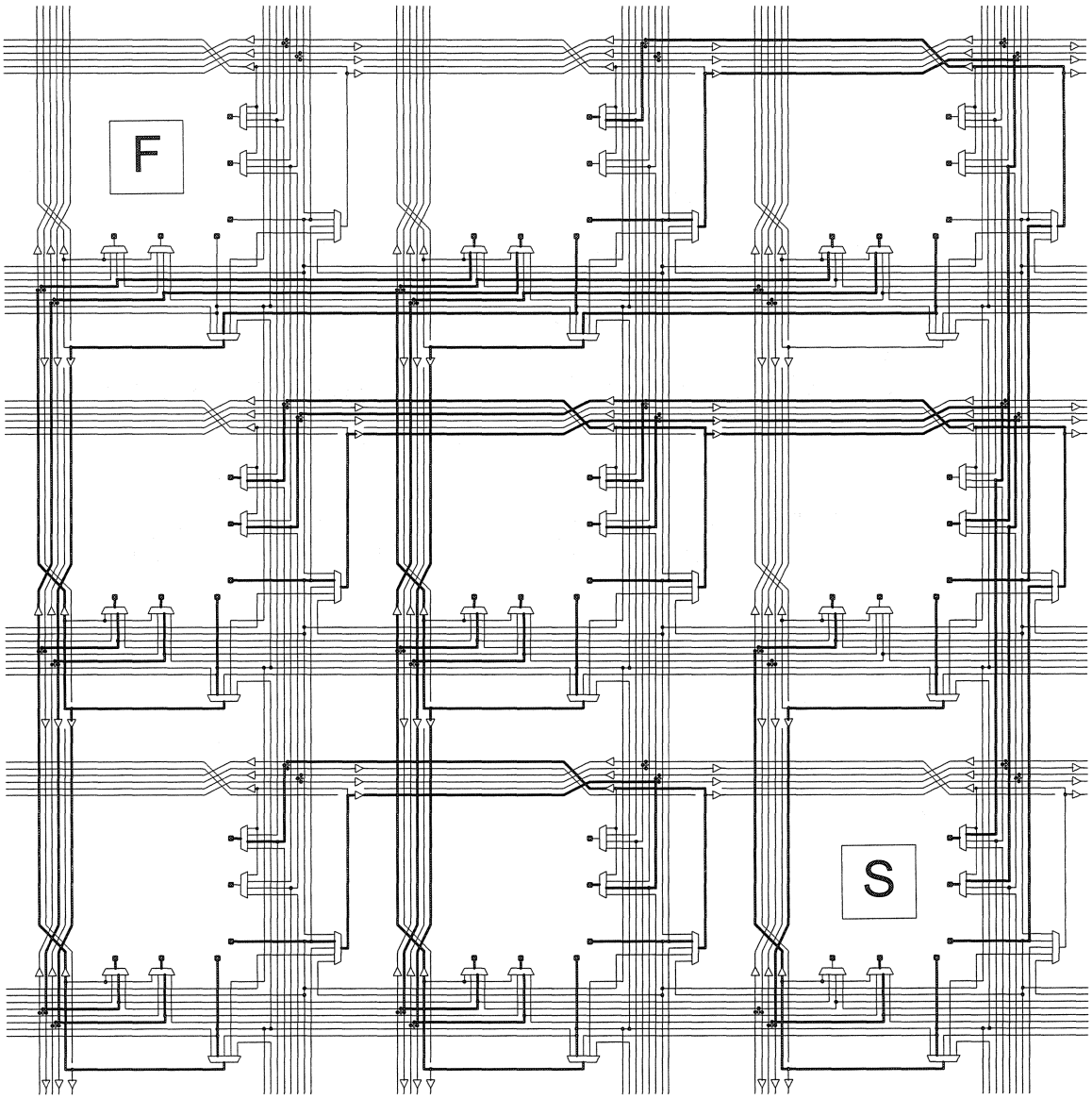


Figure A.25 Failed cell replacements with L-turned path

R_C_L[(0,0),2] + R_C_U[(2,0),2]

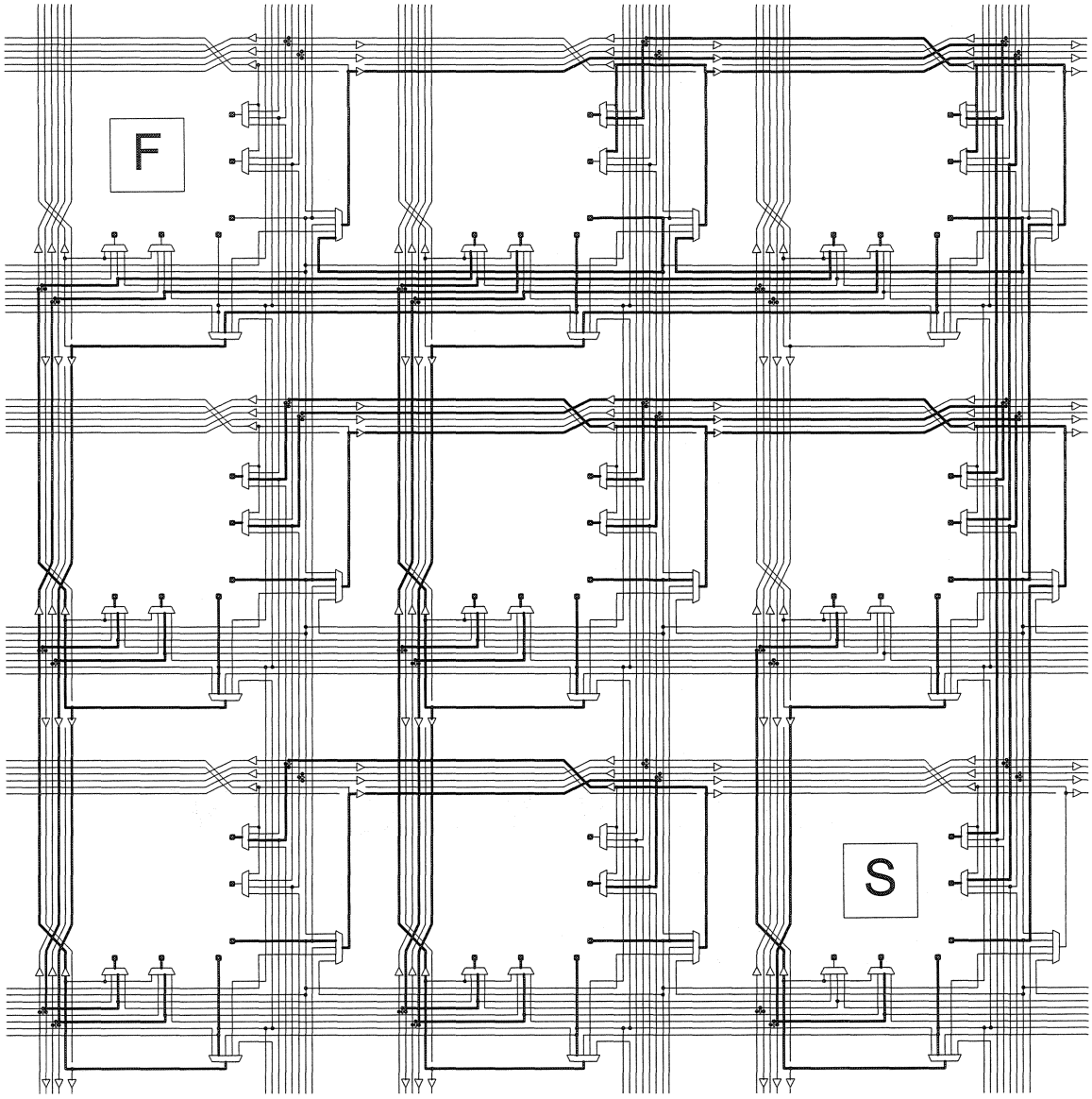


Figure A.26 Cell replacements with L-turned path

R_C_L[(0,0),2] + R_C_U[(2,0),2] + R_B_H_R[(2,0),2]

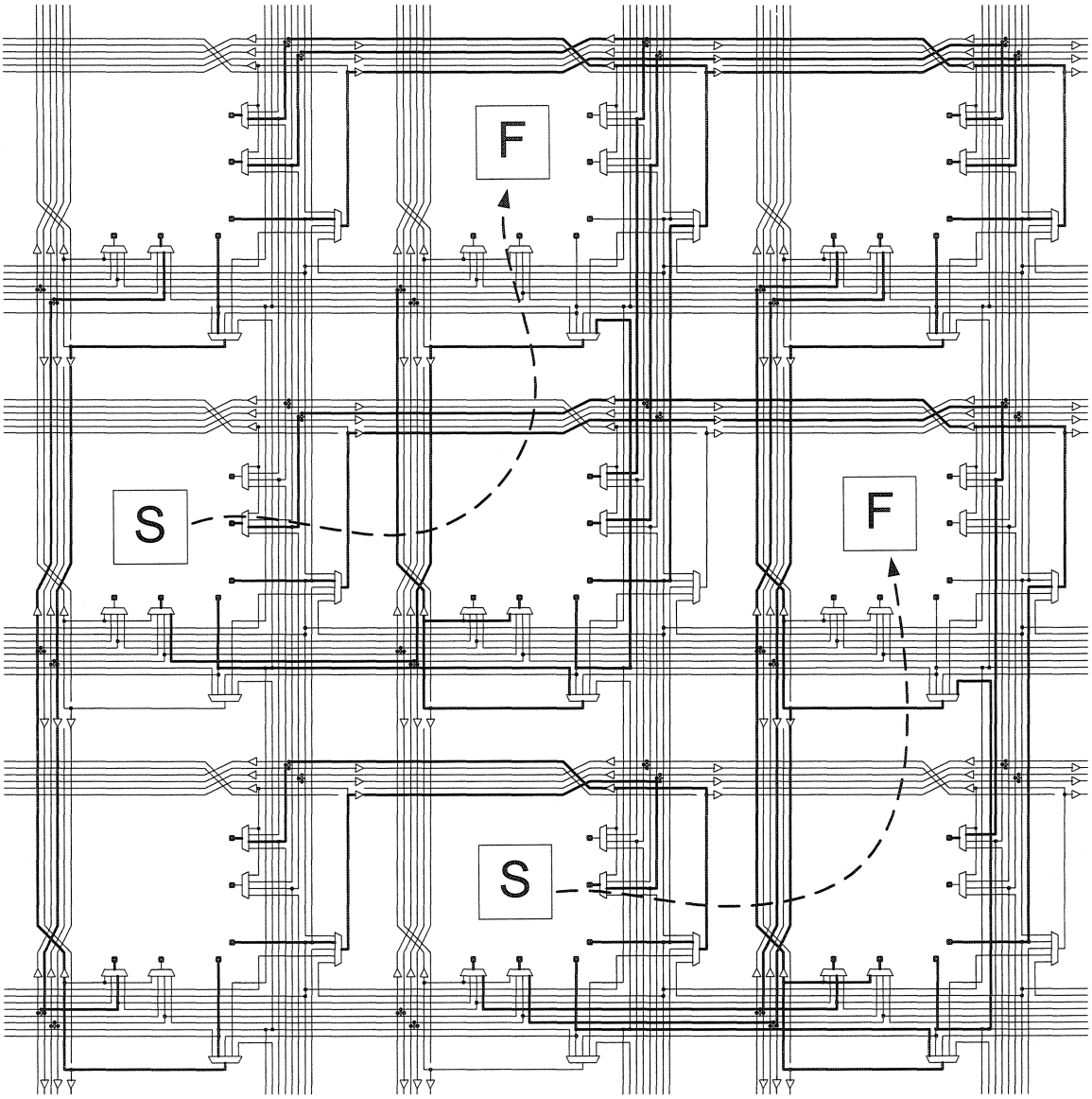


Figure A.27 Cell replacements with two parallel L_turned paths

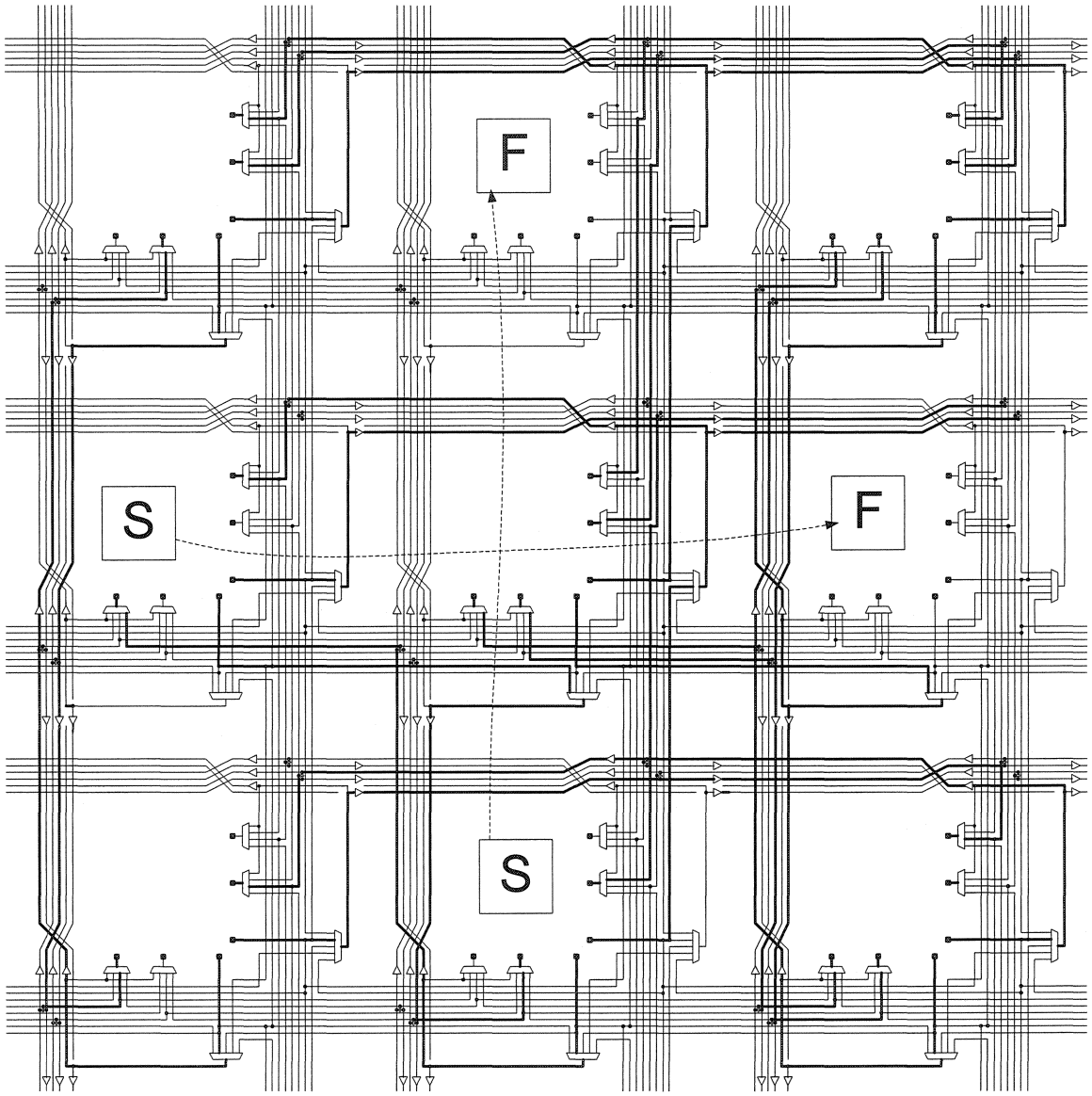


Figure A.28 Two cell replacements with crossed paths

Hazard (confused logic index)

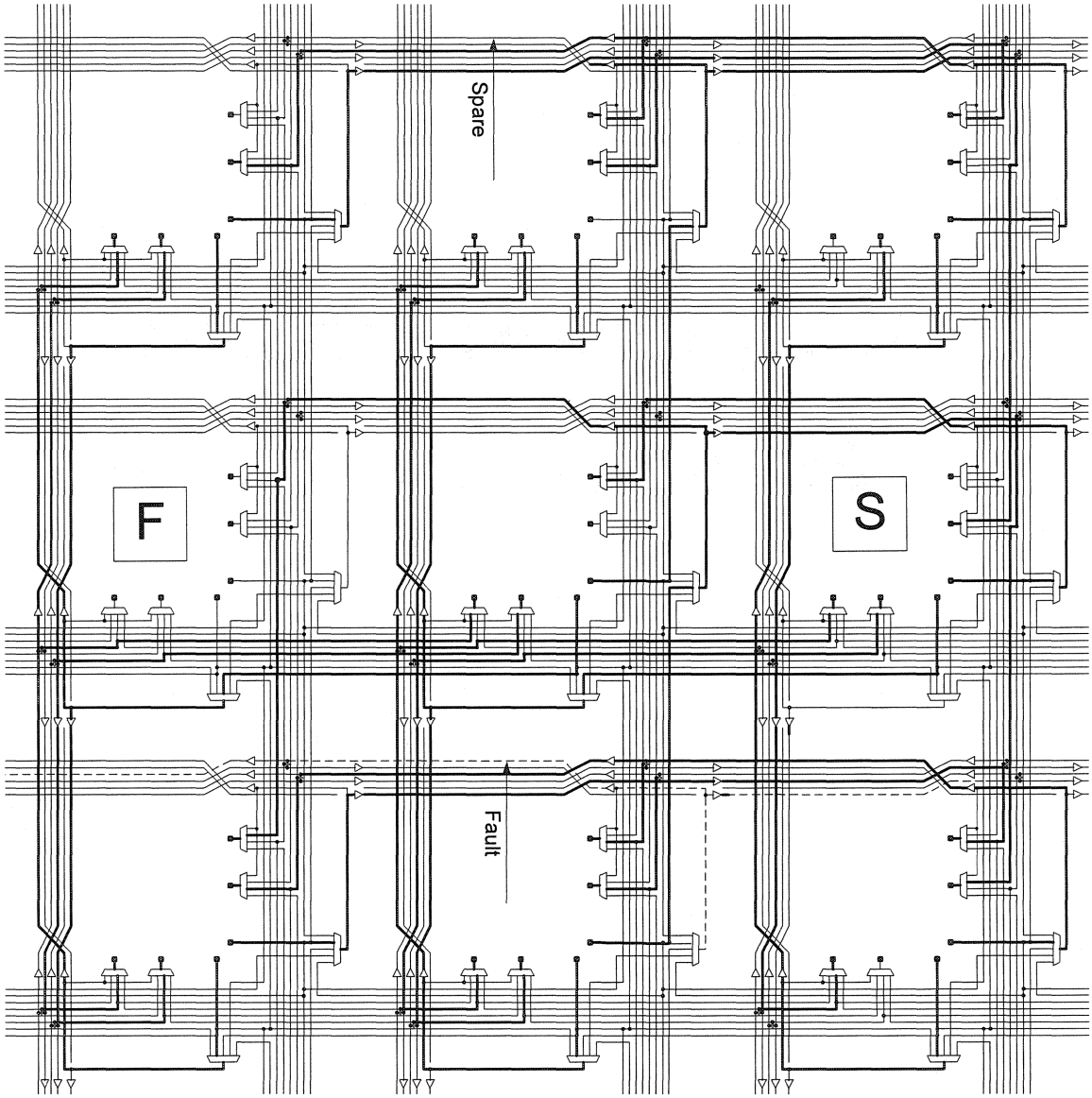


Figure A.29 Cell replacement perpendicular with Inter-bus Bundle Replacement

R_C_L[(0,1),2] R_B_V_D[(1,2),2]

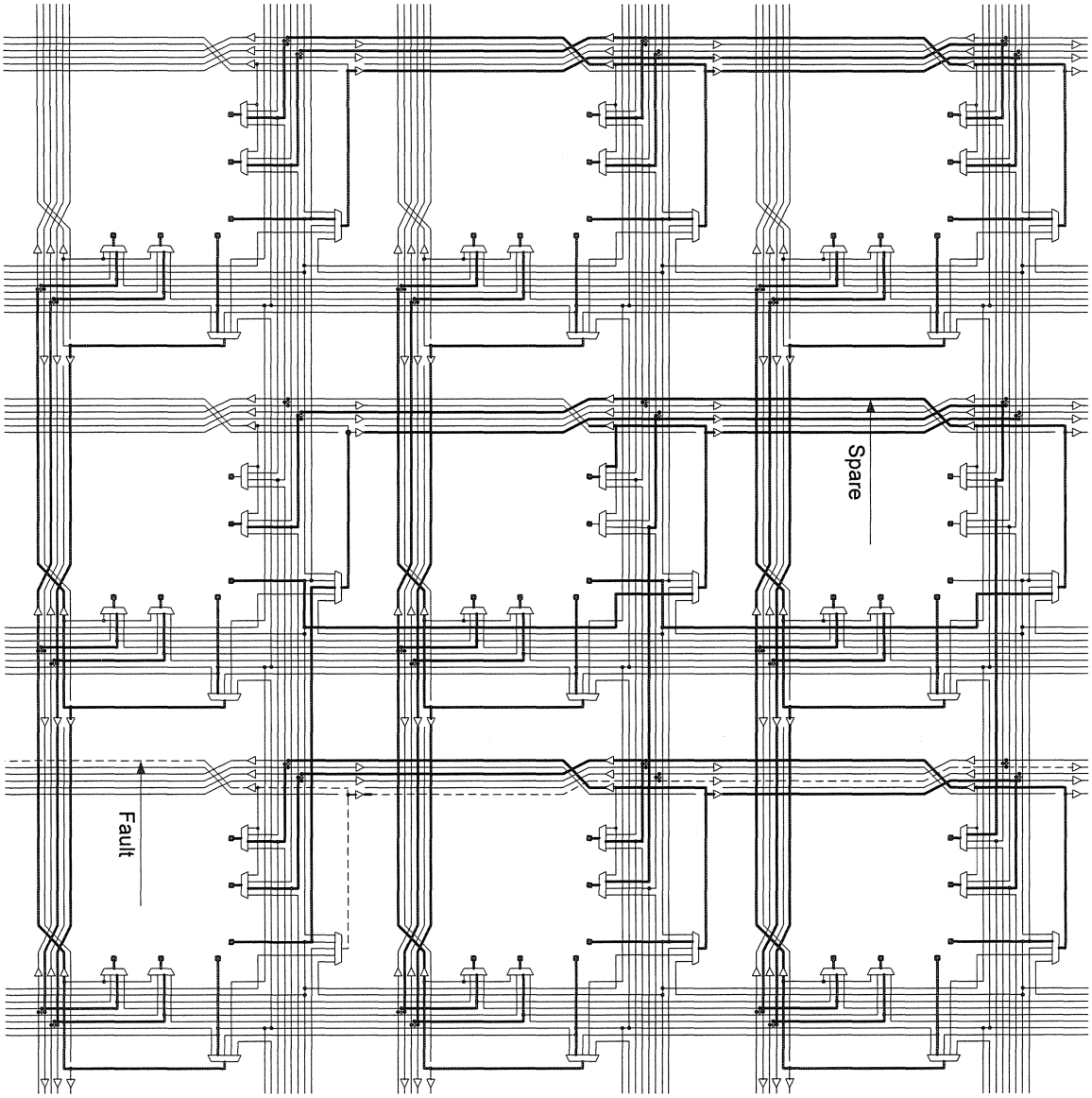


Figure A.30 Inter-bus Bundle Replacement continue with Intra-bus Bundle Replacement

R_B_H_D[(0,2),1] R_B_H_L[(0,1),2]

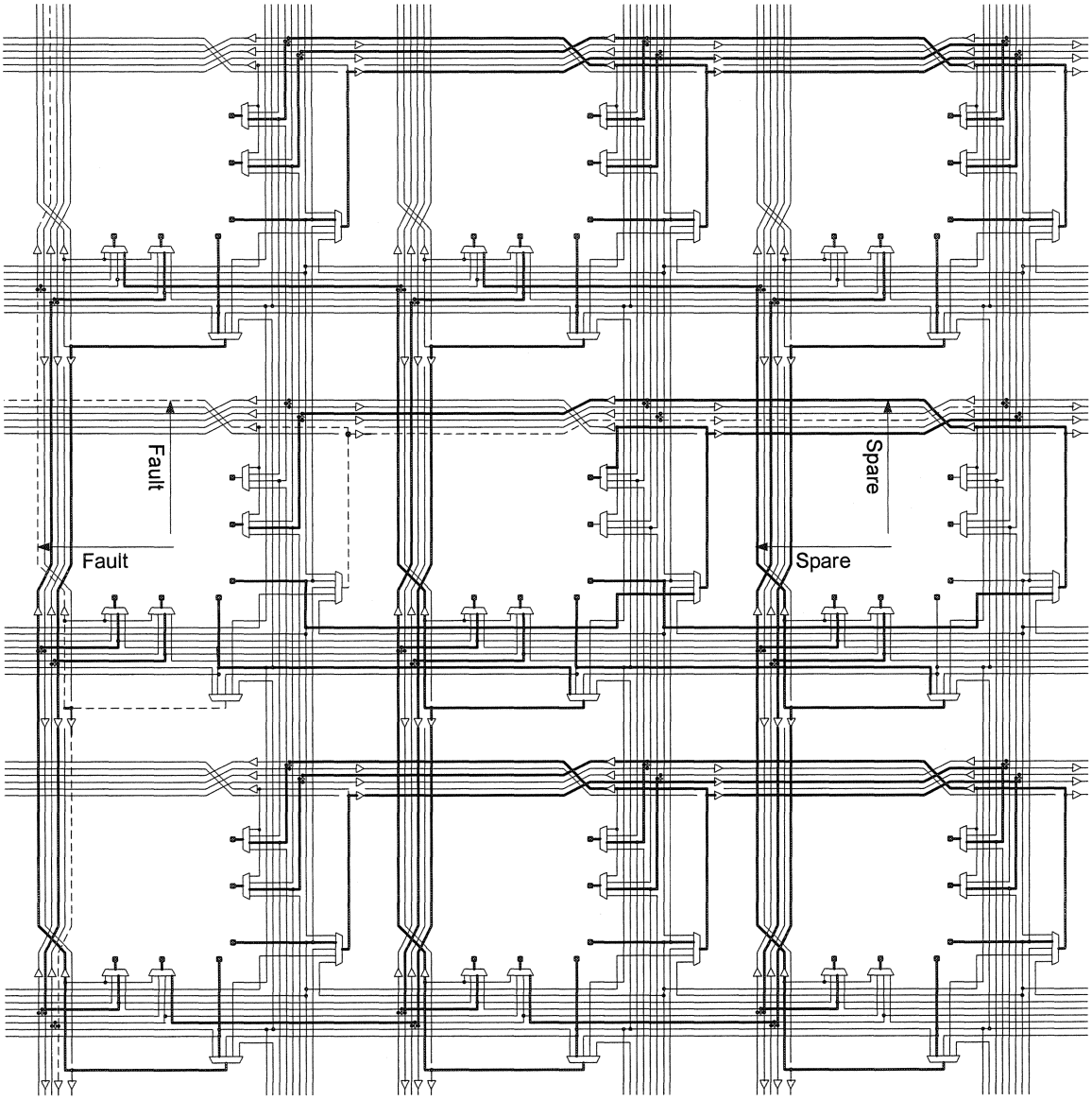


Figure A.31 Intra-bus Bundle Replacement parallel with Inter-bus Bundle Replacement

R_B_H_L[(0,1),2] R_B_V_L[(0,1),2]

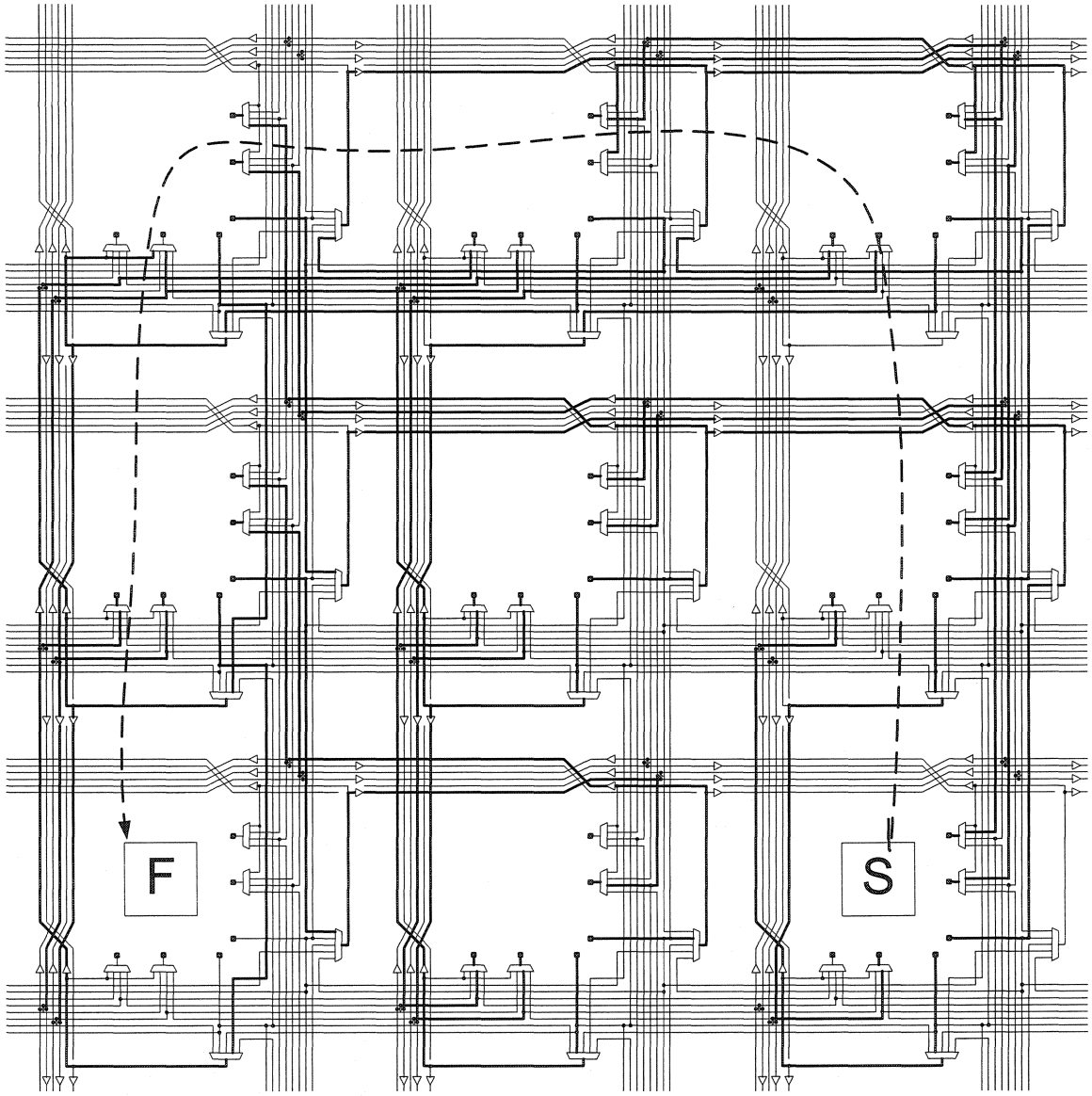


Figure A.32 Cell replacements with U-turned path

R_C_L[(0,0),2] + R_C_U[(2,0),2] + R_B_H_R[(2,0),2] + R_C_D[(0,2),2] + R_B_V_U[(0,0),2]

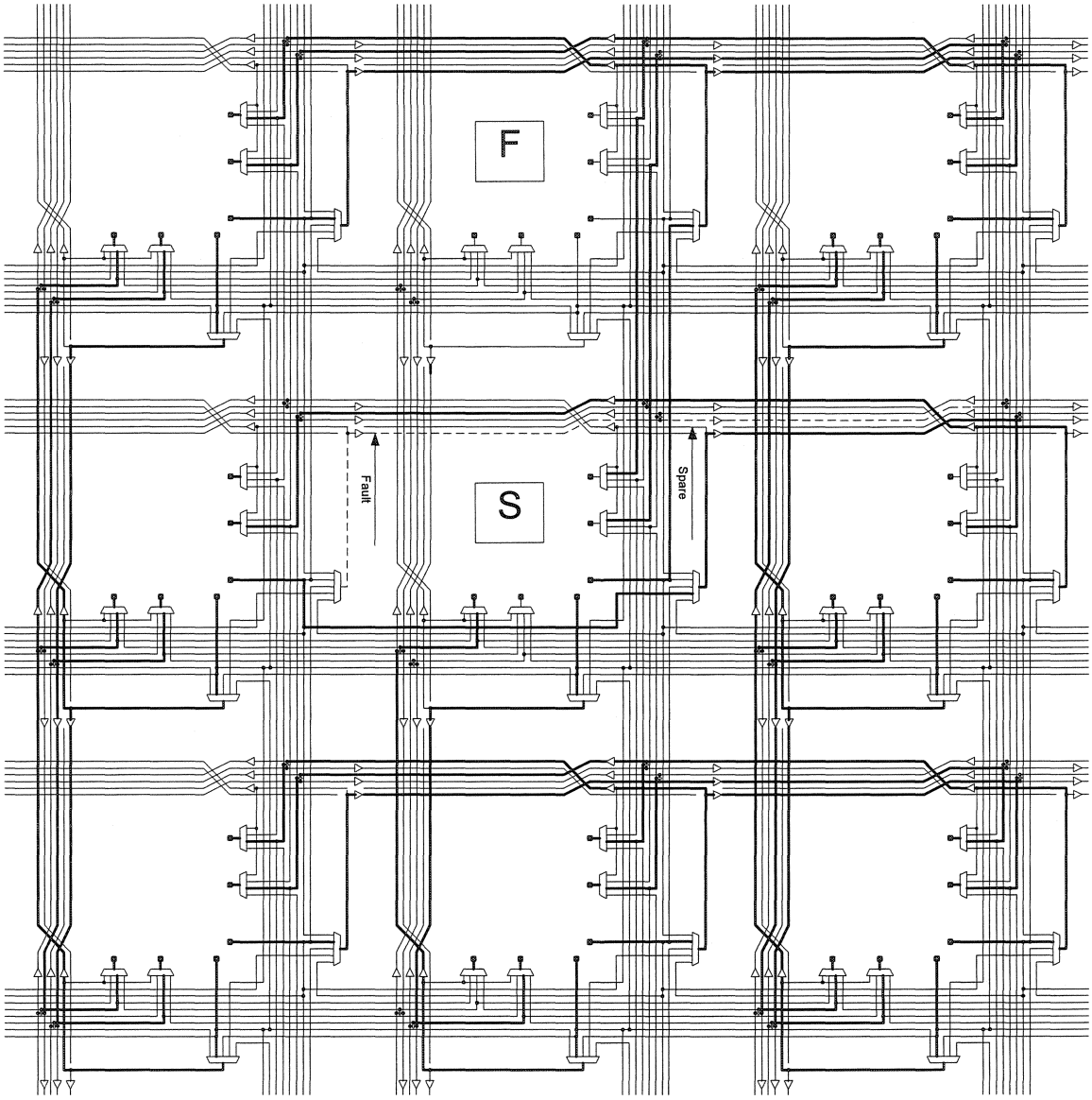


Figure A.33 Spare cell is the contributor of the spare bundle

R_B_H_L[(0,1),1] R_C_U[(1,0),1]

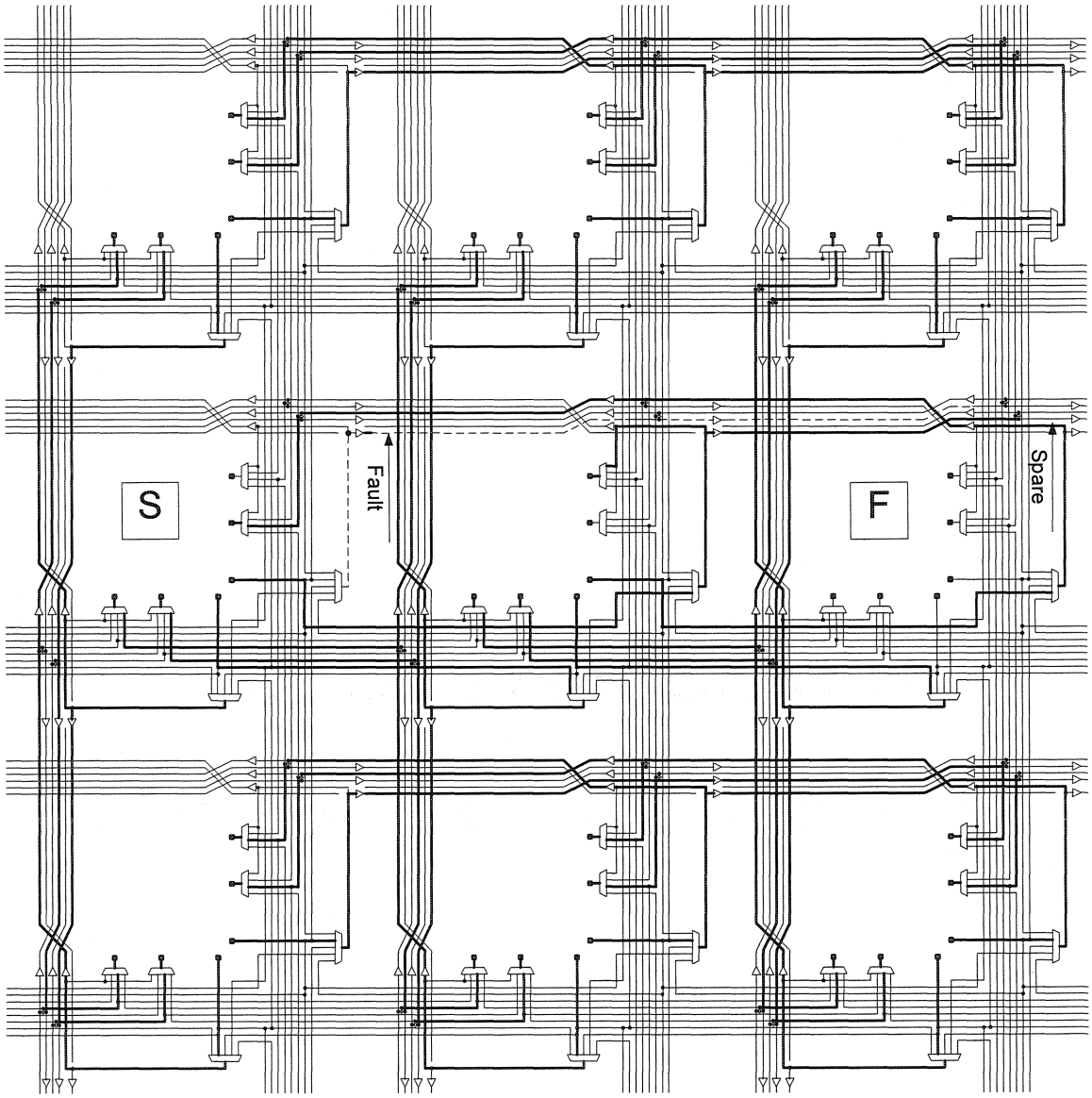


Figure A.34 Cell replacement parallel with Intra-bus Bundel Replacement

R_B_H_L[(0,1),2] R_C_R[(2,1),2]

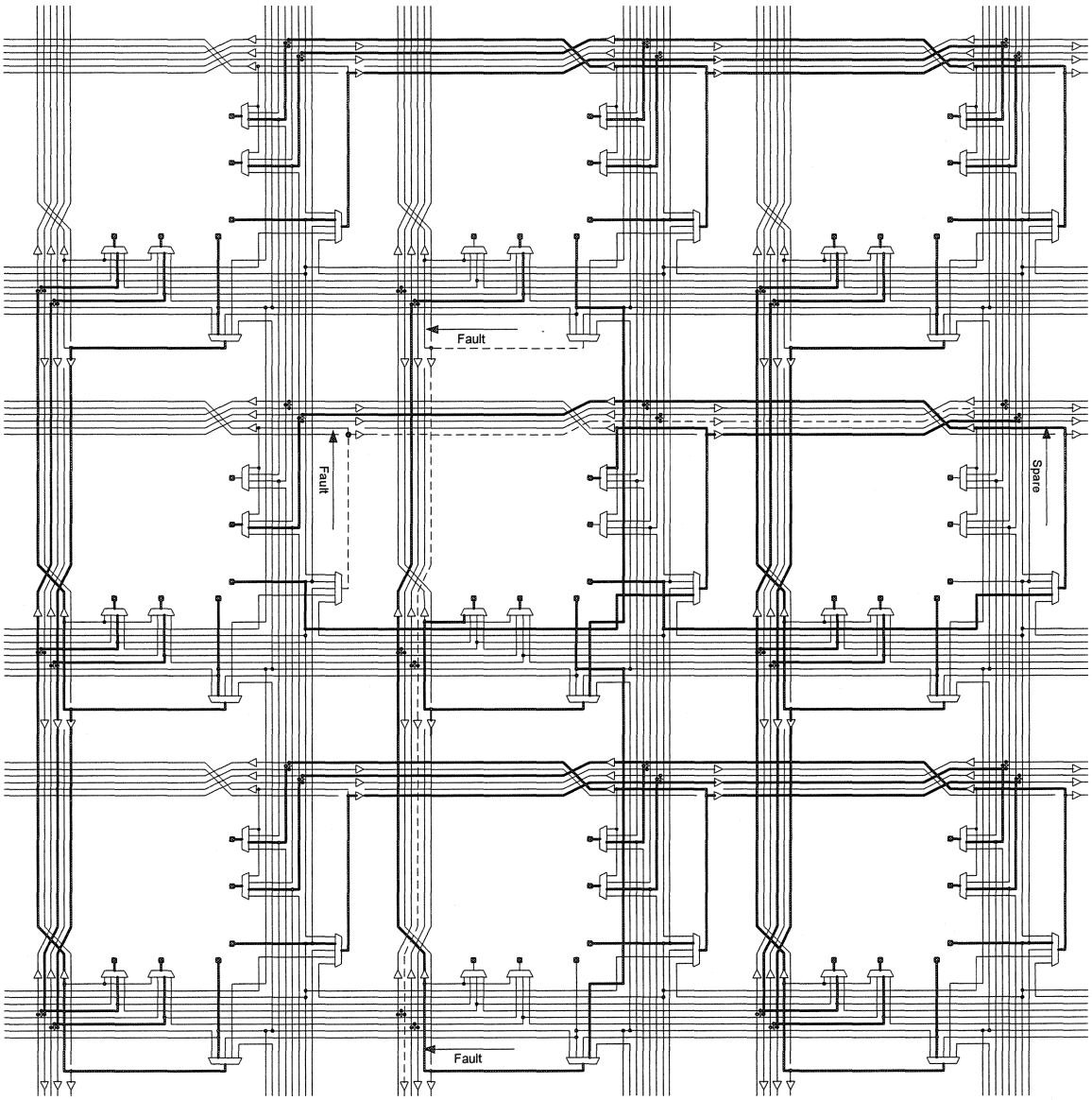


Figure A.35 Horizontal Intra-bus Bundle Replacement cross vertical Intra-bus Bundle replacement

R_B_H_L[(0,1),2] R_B_V_U[(1,0),2]

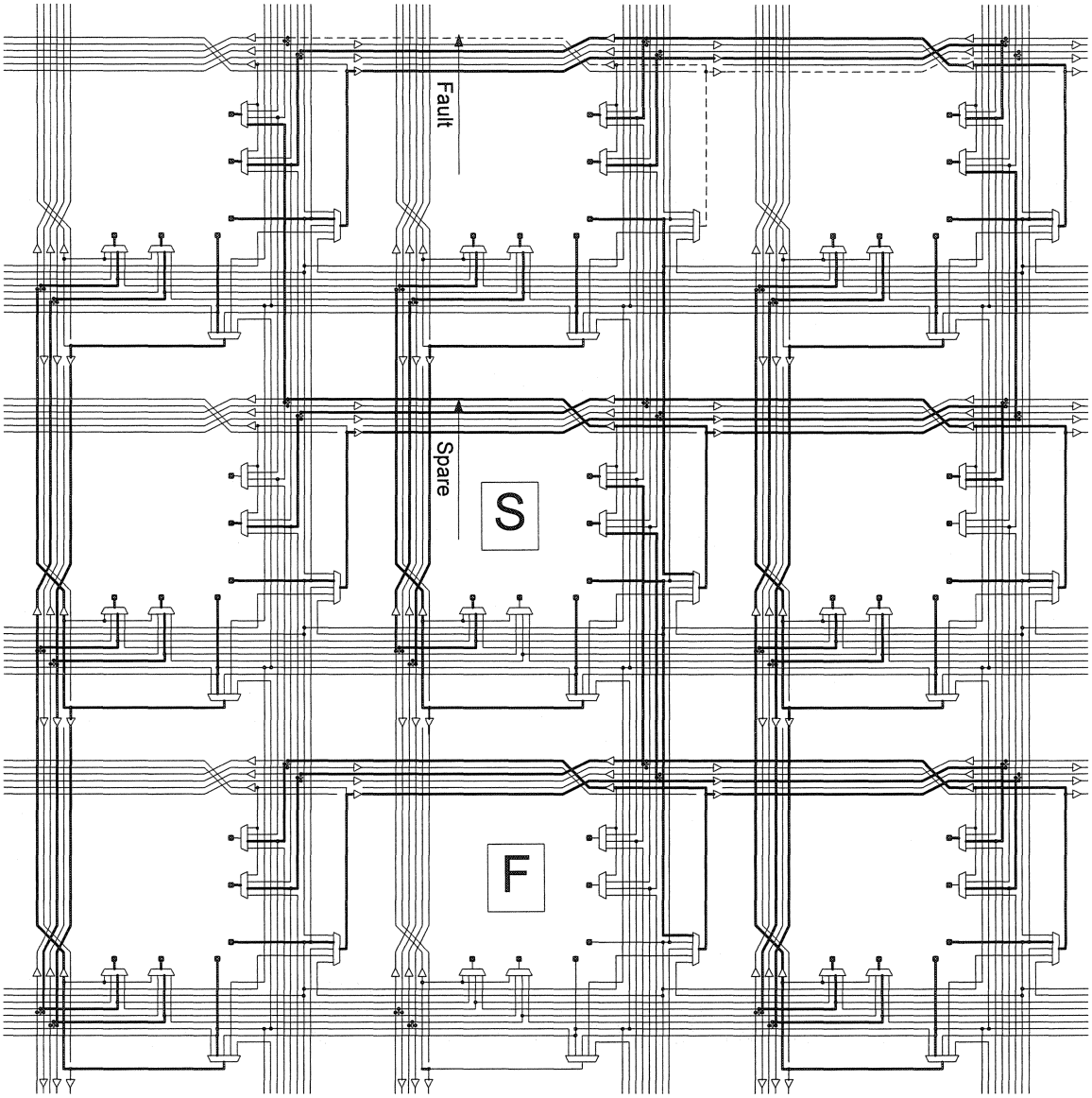


Figure A.36 Inter-bus Bundle Replacement continue with cell replacement

R_B_H_U[(1,0),1] R_C_D[(1,2),1]

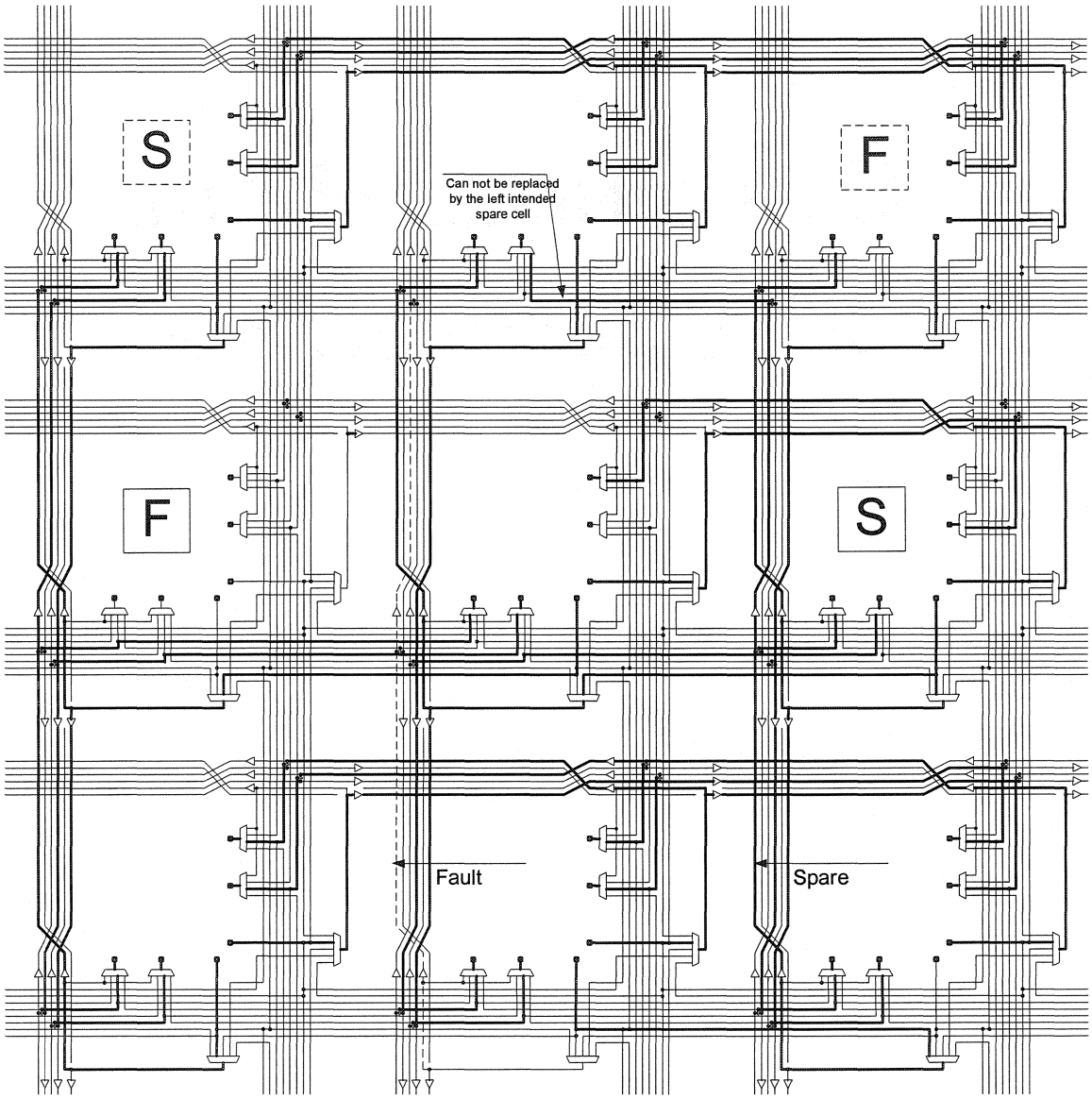


Figure A.37 Inter-bus Bundle Replacement parallel with cell replacement

R_B_V_L[(1,2),1] R_C_L[(0,1),2]

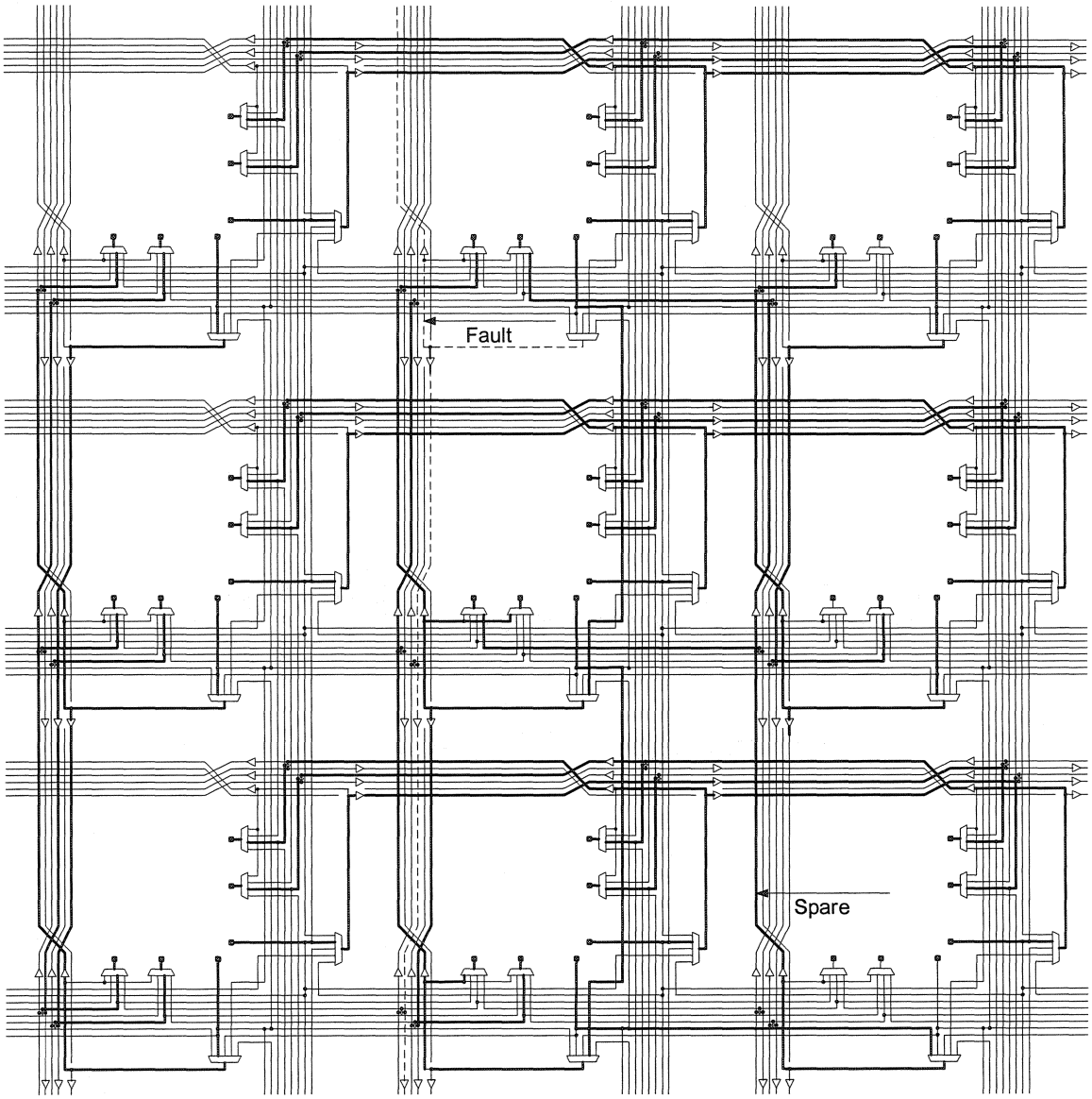


Figure A.38 Bundle replcement with with L-turned path

R_B_V_L[(1,2),1] R_B_V_U[(1,0),2]

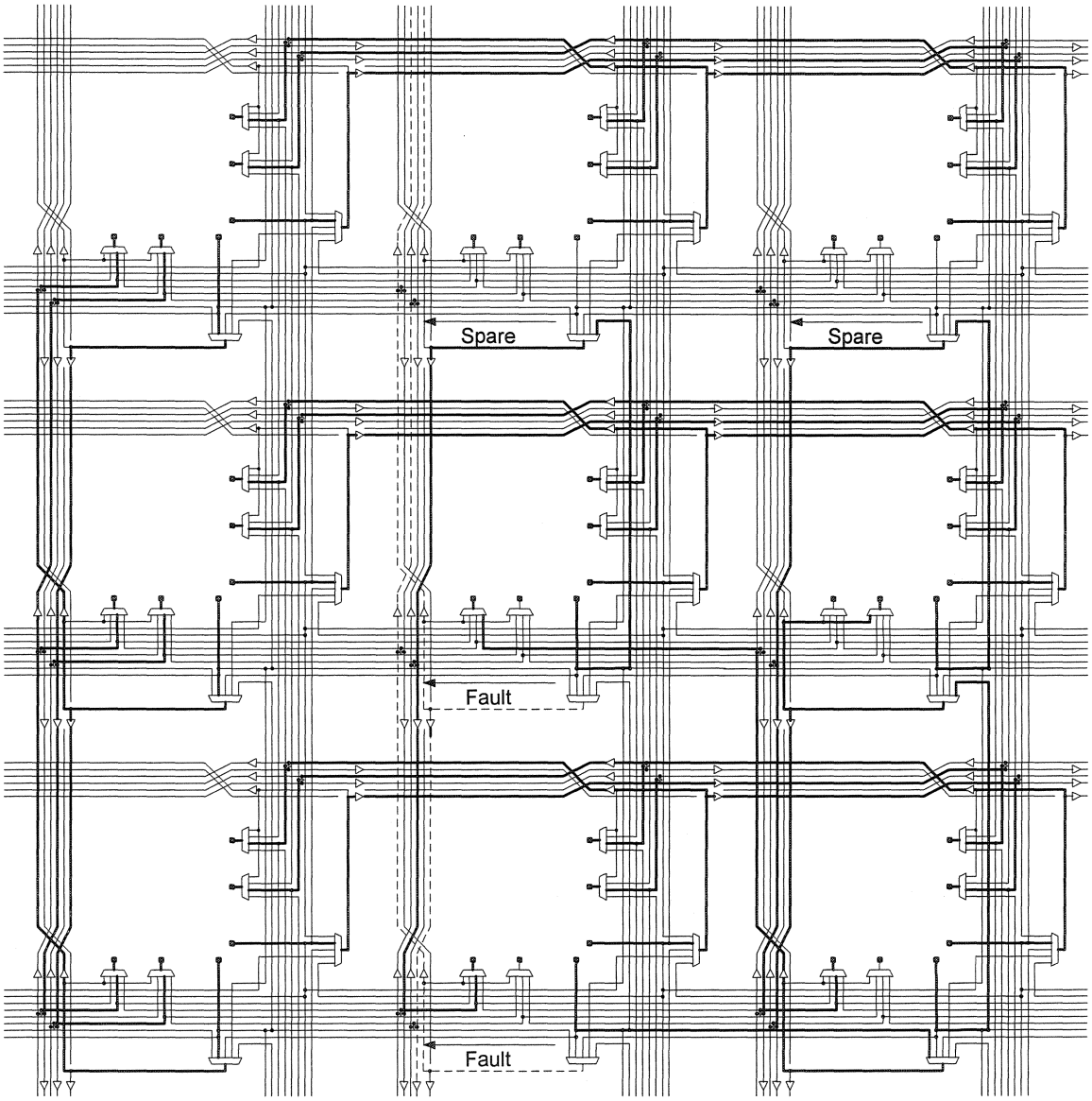


Figure A.39 Inter-bus Bundle Replacement combine with Intra-bus Bundle Replacement

R_B_V_L[(1,2),1] R_B_V_D[(1,1),1] R_B_V_D[(2,2),2]

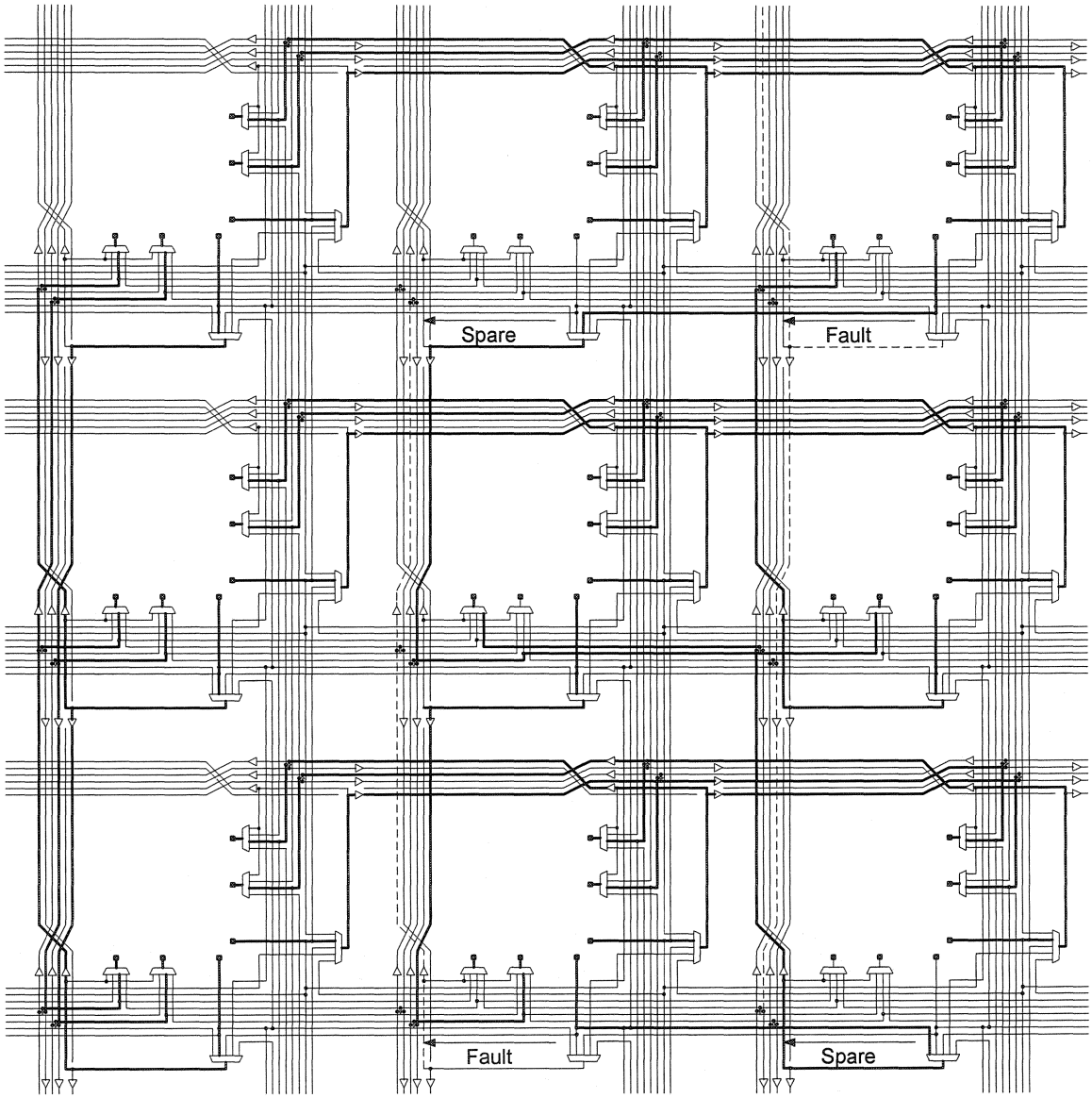


Figure A.40 Two Inter-bus Bundle Replacements in parallel

R_B_V_L[(1,2),1] R_B_V_R[(2,0),1]

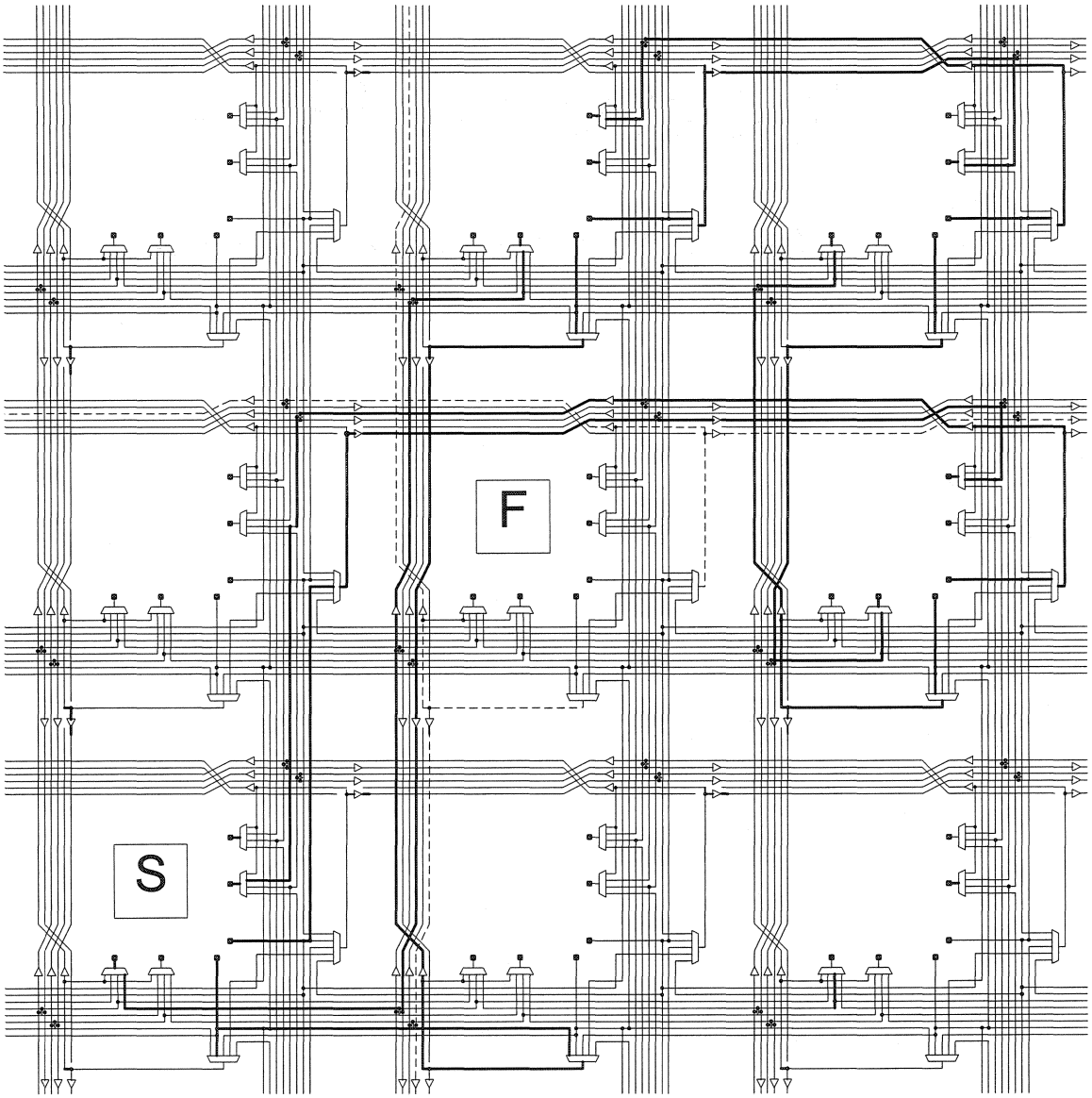


Figure A.41 Diagonal cell replacement (simple)

R_C_U[(0,1),1] R_C_R[(1,2),1]

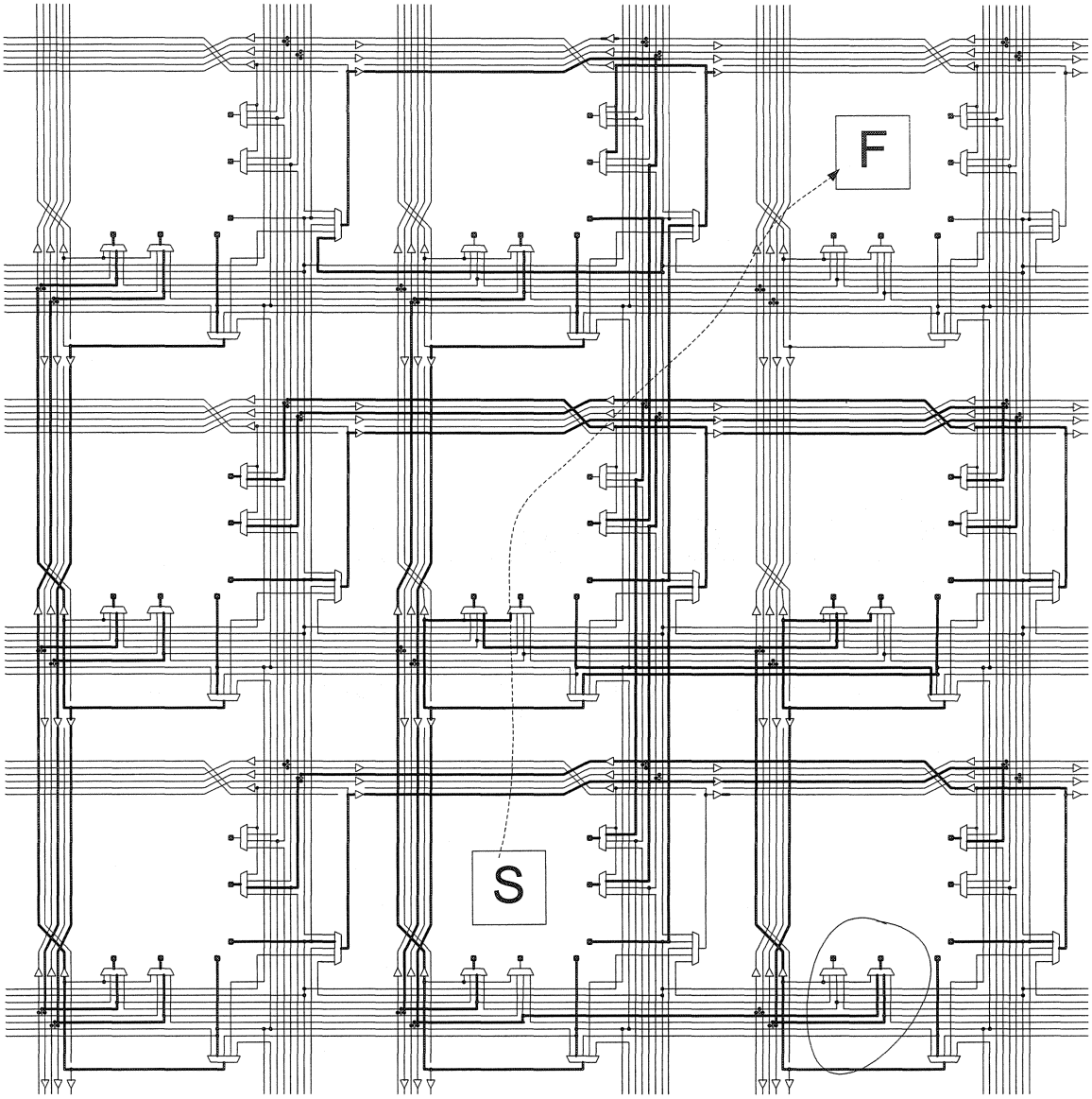


Figure A.42 Diagonal cell replacement (Receiver Source Exlusive Rule violation)

R_C_U[(1,0),1] R_C_R[(2,1),1] R_C_U[(1,1),1] R_B_H_R[(1,0),1] R_B_V_L[(1,1),1]

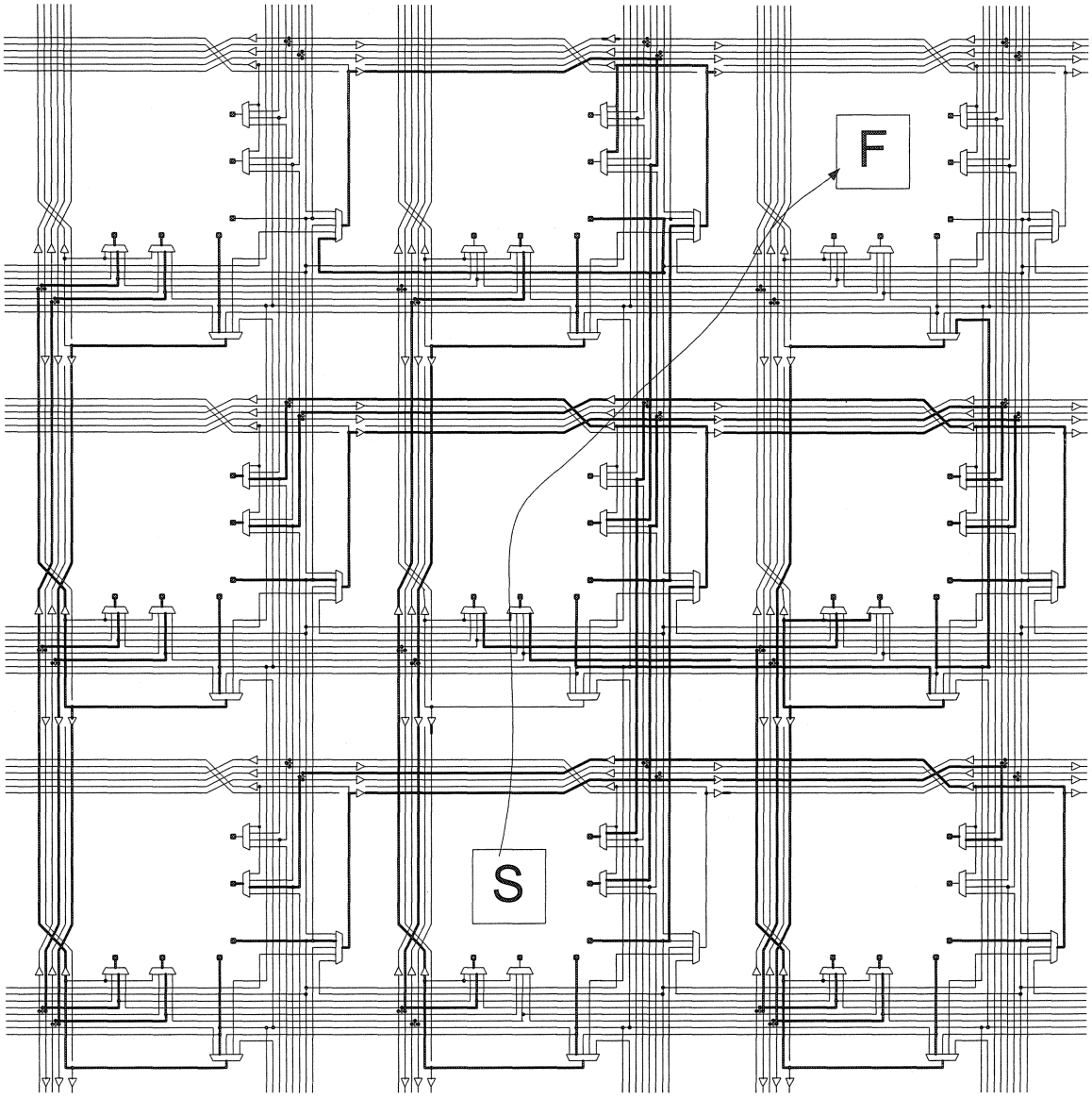


Figure A.43 Diagonal cell replacement (no violation)

R_C_U[(1,0),1] R_C_R[(2,1),1] R_C_U[(1,1),1] R_B_H_R[(1,0),1]
R_B_V_D[(2,1),1]

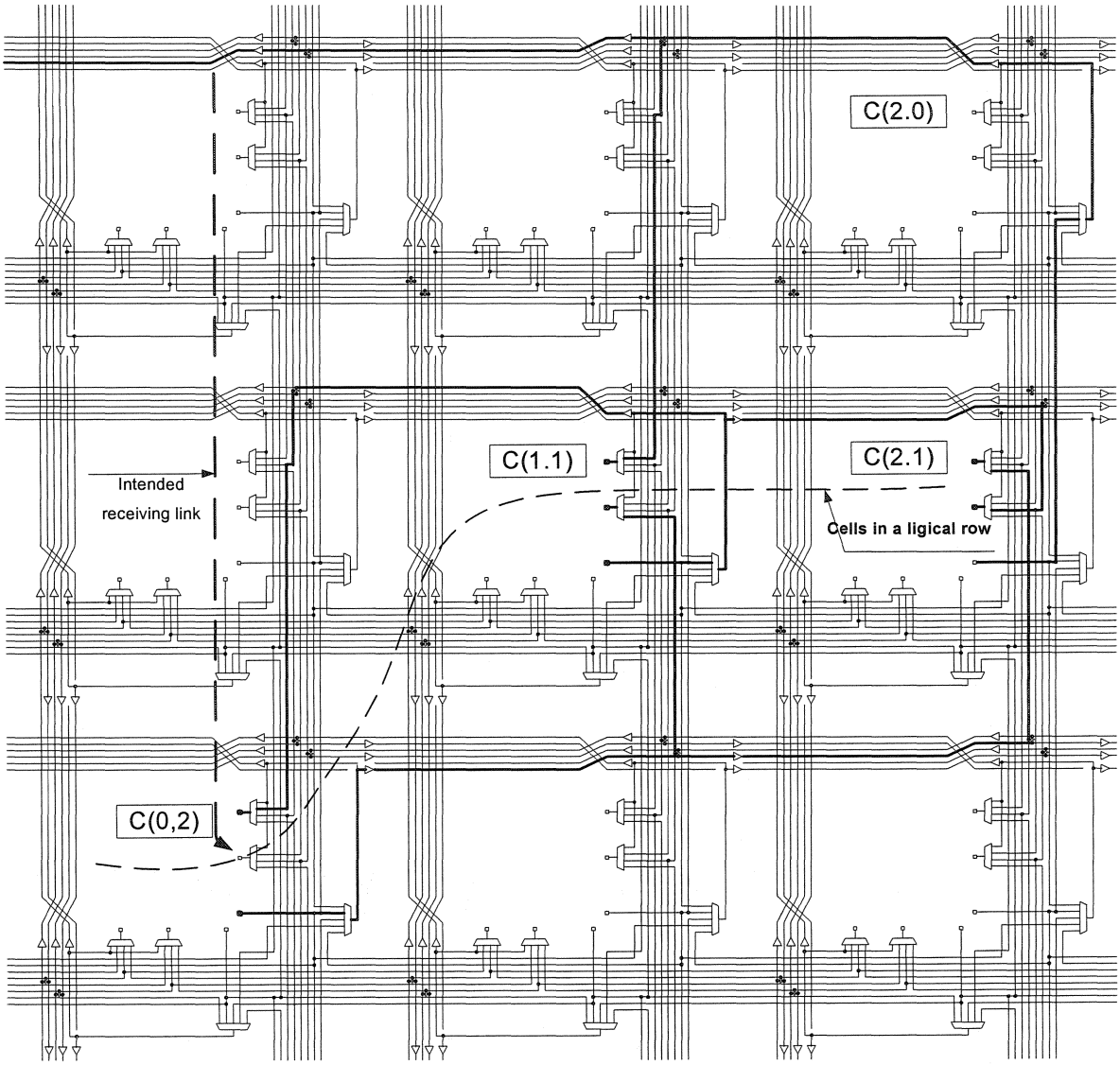


Figure A.44 Violation of Receiver Connectivity Rule (Rule.3)

Cells (C(0,2), C(1,1), C(2,1)) from two physical rows intend to form a logical row. According to configuration Rule 3, the available vertical bundles should be within the two rows (in bus V1, V2). This figure shows that no connectivity problem is caused if cell C(0,2) and cell C(1,1) connect to their natural bundles, that connectivity problem is caused if cell C(2,1) connects to cell C(2,0)'s bundle. The intended link shown by dashed bold line is not possible to establish due to connectivity constraints.

ANNEXE B

INTER-CONFIGURATION OPERATION CONSTRAINTS

An analysis is conducted to characterize the conflicts between two configuration operations. Four situations of two configuration operations are considered. The symbols and the corresponding real situations are shown in Figure X.1. The analysis results are shown in Table X.1 As shown in Figure A.28, the cell (1,1) can replace neither cell (2,1) nor cell (1,0) after the two crossing cell replacement. It is obvious that the two cell replacement operations crossing is not supported since it cannot form expected connections. As shown in Figure A.26, the two cell replacements can touch to form a turned path cell replacement. It can be summarized that the two cell replacements crossing or touching are partially supported. As shown in Figure A.27, the two parallel turned path cell replacements don't interfere each other. The situation of two cell replacement operations in parallel is fully supported. The rest in the table can be interpreted accordingly.

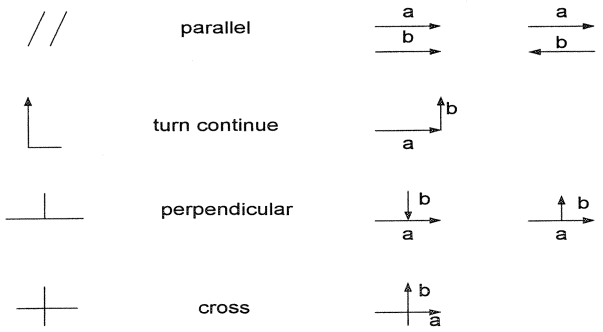


Figure B.1 The situations to be considered and the corresponding symbols

Table B.1 Inter configuration operation constraints

	Cross situation	Cell replacement	Intra-bus bundle replacement	Inter-bus bundle replacement
Cell replacement	⊥ + ↾	⌐ [+ O (App.28) ⊥ and ↾ ℓ (App 26)]	⌐ [Concern spare cell ℓ (App. 33) Concer other shifting cell O]	ℓ (App.29)
	//	ℓ(A.27)	ℓ	⌐
Intra-bus bundle replacement	⊥ + ↾	N/A	ℓ	⌐ [↾ ℓ⊥ and + O]
	//	N/A	ℓ	ℓ (App.31)
Inter-bus bundle replacement	⊥ + ↾	N/A	N/A	ℓ (App.40)
	//	N/A	N/A	ℓ

Where,

ℓ Fully supported

⌐ Partially supported

O Do not supported